



Deep Learning for Prediction of Business Outcomes

Final Project

Image Diagnosis of Pneumonia

Project Group 5, Section 21

Group Members:

Fengqiao Yang 490489 (MSBA)

Shijie Liu 491859 (MSBA)

Xingchen Li 489489 (MSBA)

May 5, 2021

Presentation recording:

https://wustl.zoom.us/rec/share/zdw06QXSe4qN50AsKPREVjczMWENStun_dsZZtqSF5Lxulz3rfKPY3IcRSWZeXRq.N6M7WKWDbul2PYPs?startTime=1620531219000

Contents

1 Introduction.....	1
2 Literature Review.....	1
3 Problem Description.....	2
4 Model Description.....	3
5 Dataset Description and Data Processing.....	4
6 Model Adjustment.....	6
7 Result.....	7
8 Conclusion, Discussion and Recommendation.....	10
9 Reference.....	11

Abstract

Pneumonia is an infection in the lungs and generally more common in children younger than 5 years old. It is the single largest cause of death in children worldwide. Report from WHO (World Health Organization) claims that "Every year, it kills an estimated 1.4 million children under the age of five years, accounting for 18% of all deaths of children under five years old worldwide". To find illness early and resolve the infection as soon as possible and to assist the doctor in the diagnosis for reducing subjective misdiagnosis, we use 2 traditional Convolutional Neural Network models with max-pooling and normalized confusion matrix to evaluate our results. In our pneumonia binary model, the accuracy of correctly predicting pneumonia chest X-Ray images and the normal chest X-Ray respectively are 91% and 97%. For the pneumonia categorical target model, the accuracy rate of 90% of healthy cases is relatively low compared to only recognizing whether it is infected or not, and 85% and 66% accuracy rate of bacteria cases and viral cases have a big gap compared with simply discriminating normal chest X-Ray images. The main reason is that the normal images are easier to distinguish and the number of normal and bacteria images is more than that of viral images.

1. Introduction

In China, there are about 57 million misdiagnosed cases in clinical medicine each year and the total misdiagnosis rate is 27.8% according to a report from the Chinese Medical Association. There are many factors composed of misdiagnosis. The main reasons are the lack of knowledge and experience of doctors or lack of imaging technology. Experience cumulation for new doctors would consume lots of resources of the hospital. In this paper, we will focus on how to use our model to classify images and to save costs for the hospital.

The model can help the medical business in many ways. During the spread of new coronary pneumonia, many countries have fallen into a serious shortage of medical supplies and personnel. In order to cope with related problems faster and help doctors reduce their workload in this emergency situation, using models to assist in judgment is a more labor-cost-saving way and can effectively reduce the possibility of encountering ethical problems. Since the number of pneumonia cases is large enough, the unit fixed cost of the purchase can be diluted, and the work efficiency of doctors has been significantly improved due to the auxiliary tools to make judgments. In addition, the level of hospital inspections also affects the hospital's competitiveness in the market. Overall, this model can improve the hospital's operational efficiency and contribute to the sustainable development of the hospital.

2. Literature Review

In the past several years, with the continuous improvement of computing power and the increase in the amount of available data, Image recognition and classification is widely used in the broad field of imaging sciences and engineering. The principle is to use a large amount of data to train the model by constructing a multi-layer model, learn the internal laws of the data, and then combine relevant information to further optimize the model.

In 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), AlexNet won the competition, after that, deep learning attracted intense attention in this field and CNN is more used in image analysis. Some Convolutional Neural Network information can be found in Qing Li et al.(2014), Tianmei Guo et al.(2017), and Yue Li et al.(2019). So, we have applied CNN to classify these images with 3 types. In addition, we only find that there are some reports using binary models to predict the result of getting sick

or not(Journal of Healthcare Engineering, 2019). For more comprehensive analysis, we use both binary¹ model and category model.

In the process of piling up layers, “ReLU’s have the desirable property that they do not require input normalization to prevent them from saturating. If at least some training examples produce a positive input to a ReLU, learning will happen in that neuron. (Alex Krizhevsky et al.,2012).” But they still use local normalization schemes to aid generalization. So, we apply normalization in our categorical model and the accuracy indeed improved about 3%.

What’s more, we learn some experience from Tolga Ergen et al.(2020).kernel_regularizer indicates that the regularizer term is applied to the weight of neurons, and each weight will add the changed weights to the loss function. In this case, the weight of each node of the model will decrease with the number of layers and L1 regularization can make most of the weights close to zero, leaving only individual weights with larger values. bias_regularizer applies a penalty on the layer’s bias and it will get similar inputs and outputs of weights. activity_regularizer will get the smaller output because it adds a penalty on the layer’s output.

Most of the models in the research papers are designed for binary output, which is to distinguish between normal chest X-Ray and pneumonia chest X-Ray. In the research paper of Okeke Stephen (2019), he constructed an efficient CNN model for the binary model of whether the patient is infected with pneumonia or not. The validation accuracy of his training is 93.73%, with a “remarkable” test accuracy as he claimed.

3. Problem Description:

Our goal is to diagnose Pneumonia based on the chest X-Ray for Asian Children. From our literature review, we could notice that most of the models are built with high accuracy to distinguish between normal and pneumonia. We would like to improve the model by having additional features to distinguish between viral and bacterial infection after we finished building the binary model. The general methods used in the literature reviews are CNN models to deal with the image input. Some of them used traditional Convolutional layers, and some used separable convolutional layers. However, drop out and regularization were barely used in their model.

We expect a similar rate of accuracy for the categorical model to predict normal chest X-Ray, but lower accuracy rate in general because of the error that may happen while distinguishing between the categories.

The first objective of our project is to distinguish between Pneumonia chest X-Ray images with the normal chest X-Ray. After getting an accuracy of 95% prediction on the testing dataset, we drew a confusion matrix to show the details of our prediction’s performance.

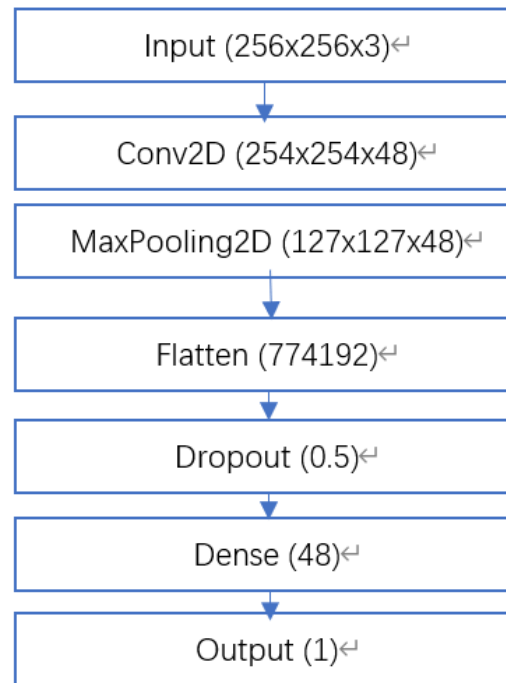
In the second part, we start to distinguish between two different types of Pneumonia: Bacterial Pneumonia and Viral Pneumonia. The output of classification would be 3 different types: normal, bacteria and virus. The most important improvement we used is data (image) modification and augmentation. After we process the images, it is much easier to distinguish the difference between Virus and Bacteria infection. Dropout and regularization were also implemented to solve the overfitting problem. And we still obtain about 90% accuracy of predicting the normal and less than 5% inaccuracy of predicting normal when it’s actually infected for either bacterial or viral.

4. Model Description:

The major method we used is CNN models (Convolutional Neural Network) combined with Max Pooling. It is the most common model researchers use to analyze the image input. Some classical models for CNN models include LeNet-5, AlexNet, and VGG-16. However, we used the traditional CNN model that comes with TensorFlow. The Max Pooling method we used after the Conv2D layers works as a filter to reduce the size of input. Max Pooling first divides the table into the desired number of sectors and chooses the max value in it. We set the Max Pooling to cut the layer size to half in order to reduce the neuron nodes. Measurement matrix of our models are all set to be accurate for the training set, validation set and testing set.

For the binary model to distinguish between normal chest X-Ray and Pneumonia chest X-Ray. We used CNN models for our image input training.

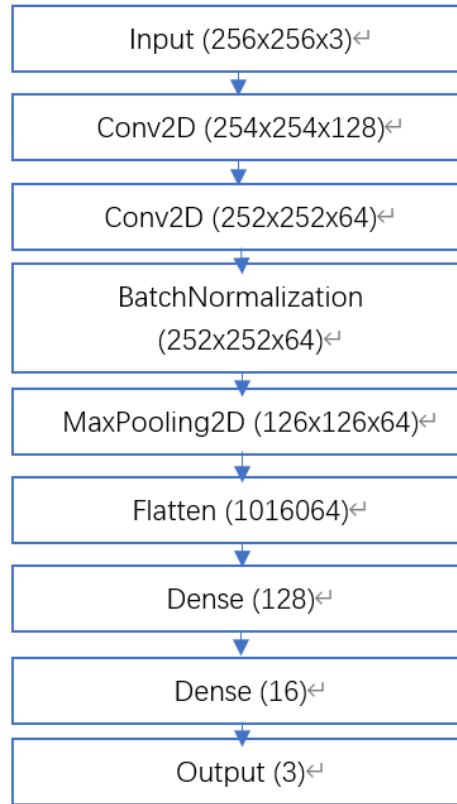
The input size is 256x256 of a single image with depth of 3 because the image mode is RGB. We have two Convolutional layers to handle the image input with two maxpooling2D followed by the convolutional layers as filters. In the next step, we add a flatten layer to transfer the 2D convolutional layers to be a fully connected layer. Drop out value is set to be 0.5 and then connected to a dense layer with 332 neuron nodes. The output size is 1 because we only expect a binary output of either normal or pneumonia.



The activation function we use for all layers other than output is “Relu”, and for the output we use “sigmoid”, because “sigmoid” works best for binary output. L2 Regularization is also applied for the first Dense layer to prevent the existence of overfitting. The loss function we use is “binary_crossentropy” and the optimizer is “adam” with a learning rate of (5e-5).

For the categorical model, we also used CNN model first to handle our image input. However, it is worth noting that we added the Batch Normalization layer after the convolutional layers. Batch Normalization is designed to stabilize the Neural Network by normalizing the layers’ input by re-scaling and re-centering.

After several numbers of trails, we found out that Batch Normalization increases the validation accuracy by about 5% generally.



While comparing the sample size of each category, we notice that viral pneumonia has significantly smaller sample size compared to bacterial pneumonia. Therefore, we applied data augmentation to our categorical model. The method we use is rotation, and we defined the rotation range to be (0,90) degrees, which means the images have several copies that are rotated between 0 to 90 degrees before entering the model. The input size of our categorical model is still 256x256 RGB images with depth of 3. The activation functions we use for all layers are “Relu” except for output. “Softmax” is a better choice as the activation function for the output after we proceed with multiple trials. The loss function is “binary_crossentropy”. We tried “categorical_crossentropy” as well, but the validation accuracy is about 3% lower compared to the loss function of “binary_crossentropy”.

5. Data Description and Data Processing

5.1 Data Description

The dataset we used in this project is a folder with 5856 grayscale chest X-ray images. Images are divided into two folders, “Normal” and “Pneumonia”, according to professional physicians. For images in “Pneumonia” folder, their names are marked with notation either “virus” or “bacteria” to identify the type of the pneumonia. Among all the images, 1583 are normal and 4273 are abnormal. Among 4273 abnormal images, 2780 observations belong to “bacteria” and 1493 observations belong to “virus”.

5.2 Data Frame Generation

To build the data frame used for training, we first read the file names from “Normal” and “Pneumonia” folders into the data frame and generate a variable ‘image’ to store the file path and a binary variable ‘y’ according to the name of folder files in. If an image belongs to normal, it is classified as 0 for attribute ‘y’, else, it is classified as 1. If we can well predict the binary target y, we also want to try to distinguish the type of pneumonia. Thus, we generate a categorical variable ‘y1’ for which images should be classified into three categories, “normal”, “virus”, “bacteria”, where the last two categories are the type of pneumonia. To fill y1, we detect type in the file name for pneumonia images and write the type into variable y1. Below is a sample slice of the data frame we generated.

	image	y	y1
0	NORMAL/IM-0548-0001.jpeg	0	normal
1	NORMAL/IM-0542-0001.jpeg	0	normal
2	NORMAL/IM-0539-0001-0001.jpeg	0	normal
3	NORMAL/IM-0547-0001.jpeg	0	normal
4	NORMAL/IM-0530-0001.jpeg	0	normal
...
5851	PNEUMONIA/person1678_virus_2895.jpeg	1	virus
5852	PNEUMONIA/person94_bacteria_457.jpeg	1	bacteria
5853	PNEUMONIA/person83_bacteria_407.jpeg	1	bacteria
5854	PNEUMONIA/person82_bacteria_404.jpeg	1	bacteria
5855	PNEUMONIA/person85_bacteria_422.jpeg	1	bacteria

After constructing the data frame, we split our dataset into 81% of train, 9% of validation, and 10% of test set randomly.

5.3 Image Processing

The original color mode of images is “L”, the gray scale. Images we obtained have different sizes and have issues of improper contrast, low sharpness, improper brightness. Thus, we process images with the following steps:

- Resize the images into 256 x 256
- Adjust contrast with *autocontrast* function
- Increase the sharpness by 20%
- Set black points of the graph to 50
- Adjust the brightness of non-black parts of images to 150.

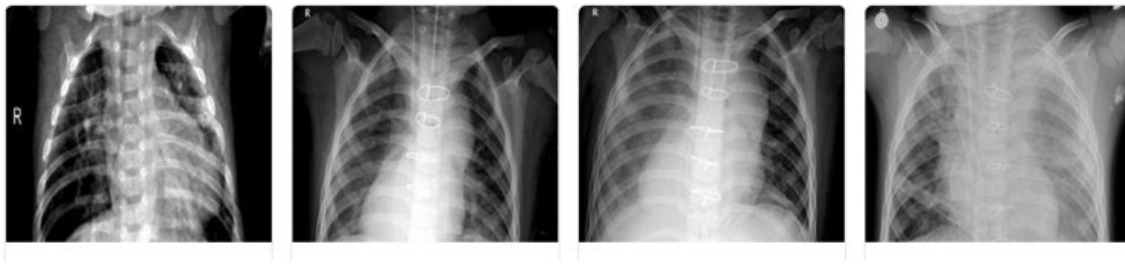
Remark: Non-black part here indicates the pixel with brightness lower than 50. We focus only on the non-black part to avoid bias of unequal distribution of background and object. The bound of non-black part of 150 is determined by trial and error.

After the preprocess, the mode of images turned to “RGB”. We found that models trained with processed images have false positive errors lower than models trained with original images set so we applied the processed dataset in all the models. Below are the sample of images before process and after process.

Images Before Process



Image After Process



6. Model Adjustment

6.1 Optimize Parameters

6.1.1 Binary Model

We start our binary model construction from a model with two Conv2D layers and three Dense layers. To improve our model, we change the layers of the model and step by step. We tried to change the number of the Conv2D layers, the number of dense layers, and the number of filters in layers. Below are parts of model structures we tried and the final structure without parameters tuned is highlighted with yellow.

Binary Model 0				Binary Model 1				Binary Model 2				Final Binary Model			
	Output Shape		Input		Output Shape		Input		Output Shape		Input		Output Shape		Input
Conv2D	(None,256,256,3)			Conv2D	(None,256,256,3)			Conv2D	(None,256,256,3)			Conv2D	(None,256,256,3)		
Max_Pooling2D	(None,127,127,48)			Max_Pooling2D	(None,127,127,48)			Max_Pooling2D	(None,127,127,48)			Max_Pooling2D	(None,127,127,48)		
Conv2D	(None,125,125,16)			Flatten	(None,774192)			Flatten	(None,774192)			Flatten	(None,774192)		
Max_Pooling2D	(None,62,62,16)			Dense	(None,48)			Dense	(None,32)			Dense	(None,32)		
Flatten	(None,61504)			Dense	(None,32)			Dense	(None,1)		Output	Dense	(None,48)		
Dense	(None,48)			Dense	(None,1)		Output					Dense	(None,1)		Output
Dense	(None,32)														
Dense	(None,1)		Output												
Test Loss: 0.4968				Test Loss: 0.4902				Test Loss: 0.3486				Test Loss: 0.3087			
Test Accuracy:0.8311				Test Accuracy:0.8501				Test Accuracy:0.8880				Test Accuracy:0.9051			

6.1.2 Categorical Model

We start our categorical model construction from a model one Conv2D layer and three Dense layers. In order to improve the performance of model, we tried to apply stratify sample split, increase number of Conv2D layers, and augment image dataset with rotated images. The final model we achieved is the model highlighted with yellow. Although the final categorical model has an accuracy similar to the categorical model 2, it has higher viral type accuracy, which is what we expected for categorical model.

Categorical Model 0			Categorical Model 1			Categorical Model 2			Final Categorical Model		
normal sample split			stratify sample split			normal sample split			normal sample split		
ImageDataGenerator	(rescale)		ImageDataGenerator	(rescale)		ImageDataGenerator	(rescale)		ImageDataGenerator	(rescale, rotation_range)	
Output Shape	(None,256,256,3)	Input	Output Shape	(None,256,256,3)	Input	Output Shape	(None,256,256,3)	Input	Output Shape	(None,256,256,3)	Input
Conv2D	(None,254,254,128)		Conv2D	(None,254,254,128)		Conv2D	(None,254,254,128)		Conv2D	(None,254,254,128)	
Max_Pooling2D	(None,127,127,128)		Conv2D	(None,252,252,64)		Conv2D	(None,252,252,64)		Conv2D	(None,252,252,64)	
Flatten	(None,774192)		Batch_Normalization	(None,252,252,64)		Batch_Normalization	(None,252,252,64)		Batch_Normalization	(None,252,252,64)	
Droupout	(None,774192)		Max_Pooling2D	(None,126,126,64)		Max_Pooling2D	(None,126,126,64)		Max_Pooling2D	(None,126,126,64)	
Dense	(None,128)		Flatten	(None,1016064)		Flatten	(None,1016064)		Flatten	(None,1016064)	
Dense	(None,16)		Droupout	(None,1016064)		Droupout	(None,1016064)		Droupout	(None,1016064)	
Dense	(None,3)	Output	Dense	(None,128)		Dense	(None,128)		Dense	(None,128)	
			Dense	(None,16)		Dense	(None,16)		Dense	(None,16)	
			Dense	(None,3)	Output	Dense	(None,3)	Output	Dense	(None,3)	Output
Test Accuracy: 0.713			Test Accuracy: 0.7576			Test Accuracy: 0.8055 Test Accuracy - Batcerial: 0.78 Test Accuracy - Virtual: 0.68			Test Accuracy: 0.8020 Test Accuracy - Batcerial: 0.77 Test Accuracy - Virtual: 0.75		

6.2 Tune Hyper-Parameters

6.2.1 Binary Model

For the binary model with the model structure we reach after adjustments in layers, we make further adjustments with parameters epoch, batch size, rate of dropout to tune the model. Parts of combinations we tried are listed in the table below. We finally reached the accuracy 95.39% with 20 epochs, batch size 16, and dropout rate 0.5 for the only dropout layer.

	Epoch	Batch Size	Dropout	Accuracy
Initial	10	64	0.2	0.905
1	20	64	0.2	0.933
2	20	16	0.2	0.945
Final	20	16	0.5	0.953

6.2.2 Categorical Model

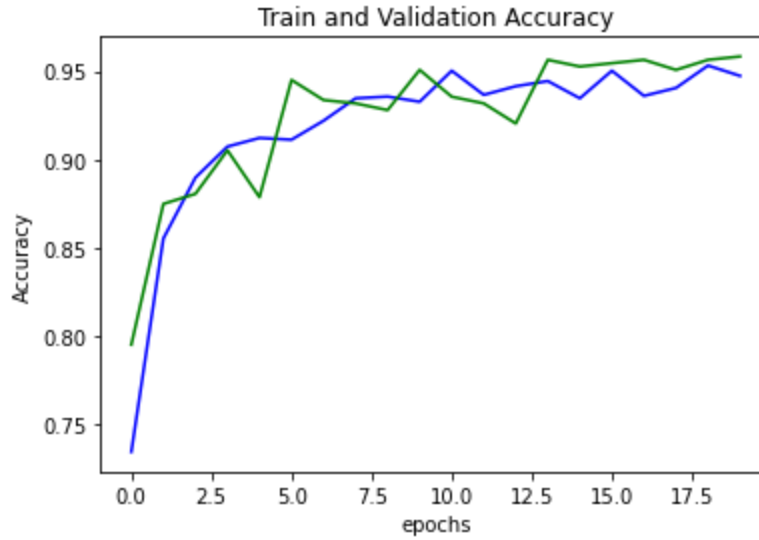
For the categorical model with the model structure we reach after adjustments in layers, we make further adjustments with parameters epoch, batch size, rate of dropout to tune the model. Parts of combinations we tried are listed in the table below. We finally reached the accuracy 80.20% with 25 epochs, batch size 64, and dropout rate 0.2 for the two dropout layers.

	Epoch	Batch Size	Dropout	Accuracy
Initial	10	16	(0.5,0.5)	0.7105
1	25	16	(0.5,0.5)	0.7438
2	25	64	(0.5, 0.5)	0.789
Final	25	64	(0.2,0.2)	0.802

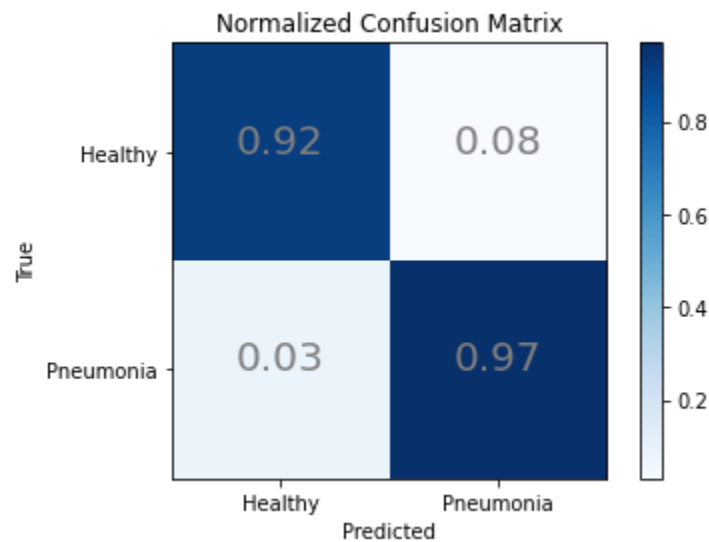
7. Results:

7.1 Binary Model:

According to the accuracy plot of the after-tuned binary model below, where blue line indicates train accuracy and green line indicates validation accuracy, we can learn that the model is neither overfitting nor underfitting for the test and validation are close and increasing.

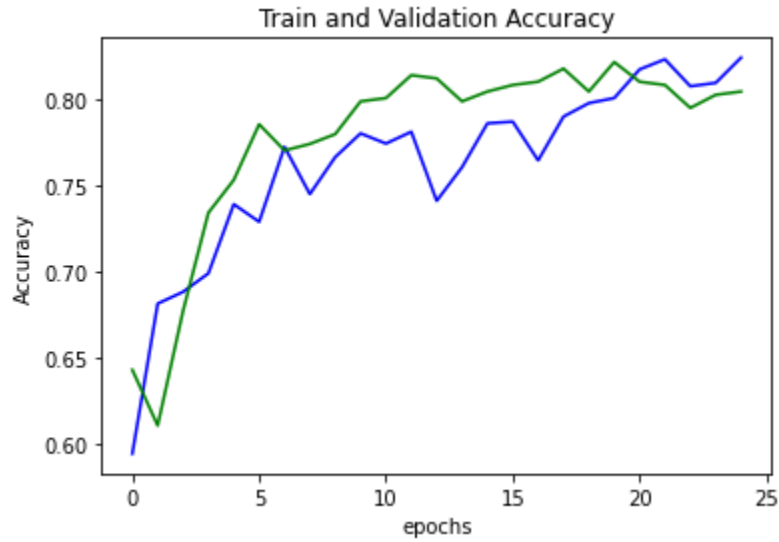


The after-tuned binary model has a test accuracy of 0.9539. According to the normalized confusion matrix below, the binary model has a type I error of 0.08 and a type II error of 0.03. It indicates that for normal samples, the model can have an accuracy of 92%, and for pneumonia samples, the model can predict with an accuracy of 97%.

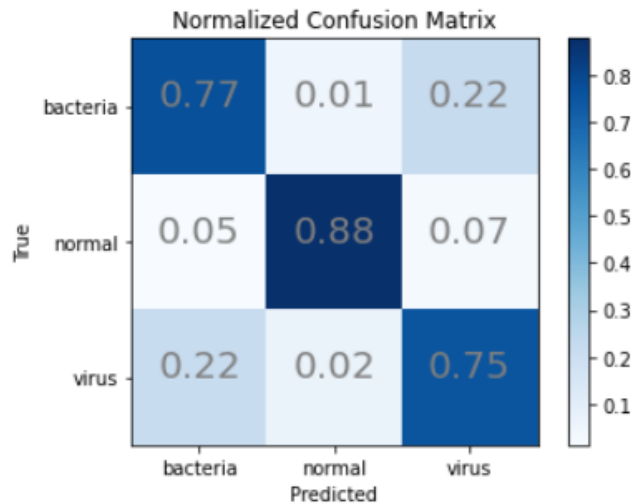


7.2 Categorical Model:

Given that the binary model predicts well, and the machine can easily distinguish normal X-ray image and pneumonia X-ray images, we constructed the categorical model, aiming to recognize the type of pneumonia. Our main goal of building this model is to classify types of pneumonia well. The train and validation accuracies of the categorical model we finally obtained are around 80%.



The final categorical model has a test accuracy of 0.802. According to the normalized confusion matrix below, the model predicts normal observations with an accuracy of 0.88. Meanwhile, the model can tell the difference between bacterial and viral samples with probability above 75%.



7.3 Model Save and Application

We save the binary model we built as `model_pneumonia` and the categorical model as `model_categorical_pneumonia` with command `save()` in order to apply the models in practice in future. To diagnose pneumonia patients with models, we can input the file path of chest X-ray images into the models. The `model_pneumonia` will output the probability that the patient is ill with pneumonia, if the probability is larger than 0.5, patient will be diagnosed as pneumonia. The `model_categorical_pneumonia` will output probability that the patient is ill with bacterial pneumonia, probability that the patient is healthy, and probability that the patient is ill with virus pneumonia. The patient will be diagnosed as the category with the largest percentage.

8 Conclusion, Discussion and Recommendation

We construct two convolution neural network (CNN) models, binary target model and categorical target model to solve our pneumonia image diagnosis problem.

8.1 Binary Pneumonia Diagnosis Model:

The model can accurately discriminate the normal and pneumonia images with an accuracy of 95.39%. The type II error, which is the false negative, is crucial in diagnosis and should be controlled at a low level. The model generates an ideal type error rate 3% and it is accurate enough for real-life diagnosis. We managed to solve our first problem.

8.2 Categorical Pneumonia Diagnosis Model:

The model performs 80.2% accuracy in discriminating against normal, bacteria pneumonia, and virus pneumonia. The model can still accurately recognize normal images and it can pick out viral pneumonia from bacterial pneumonia with accuracy more than 75%. It's a satisfactory result.

The results of the models show that both models can accurately tell the difference between a healthy and pneumonia image, and the categorical model identifies the exact type of pneumonia with normal accuracies. Evaluating medically, the performance of the categorical model is reasonable because bacteria and virus pneumonia can be easily distinguished in early stages but will become similar in image patterns as conditions worsen. Moreover, the whole dataset only includes thousands of observations, the model can be further improved with more training data before being put into practice.

Generally speaking, the Binary Pneumonia Diagnosis Model and Categorical Pneumonia managed to solve the problem of distinguishing pneumonia patients and distinguishing pneumonia type.

9 References

- Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, 2012,
<https://papers.nips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>
- Chinese medical association, <https://www.cma.org.cn>
- Okeke Stephen, Mangal Sain, Uchenna Joseph Maduh, Do-Un Jeong, "An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare", *Journal of Healthcare Engineering*, vol. 2019, Article ID 4180949, 7 pages, 2019.
<https://doi.org/10.1155/2019/4180949>
- Qing Li; Weidong Cai; Xiaogang Wang; Yun Zhou; David Dagan Feng; Mei Chen, Medical image classification with convolutional neural network, 2014,
<https://ieeexplore.ieee.org/abstract/document/7064414>
- Tianmei Guo; Jiwen Dong; Henjian Li; Yunxing Gao, Simple convolutional neural network on image classification, 2017, <https://ieeexplore.ieee.org/abstract/document/8078730>
- Tolga Ergen & Mert Pilanci, IMPLICIT CONVEX REGULARIZERS OF CNN ARCHITECTURES: CONVEX OPTIMIZATION OF TWO- AND THREE-LAYER NETWORKS IN POLYNOMIAL TIME, 2020,
<https://openreview.net/pdf/dba1d25e1354e478235ccc68af0dd34e0cf91c79.pdf>
- Yue Li; Dong Liu; Houqiang Li; Li Li; Zhu Li; Feng Wu, Learning a Convolutional Neural Network for Image Compact-Resolution, 2019,
<https://ieeexplore.ieee.org/abstract/document/8476610>