Pneumonia Image Classification using EfficientNet with Focal Loss

Wenyu Li*, Shujin Wei†, Yunling Bai†, Zikang Song† and Teoh Teik Toe‡
*School of Aeronautics and Astronautics,University of Electronic Science and Technology of China, China
Email: 202022100534@std.uestc.edu.cn
†Glasgow College,University of Electronic Science and Technology of China, China
Email: 2614082w@student.gla.ac.uk, 2510792B@student.gla.ac.uk, 2020190908022@std.uestc.edu.cn
‡NTU Business AI Lab,Nanyang Technological University, Singapore
Email:ttteoh@ntu.edu.sg

***Abstract*—Transfer learning, an optimization, can realize higher processing speed or better performance in learning a new task in deep learning (DL). In this paper, we discuss how to distinguish pneumonia patients based on binary classification by utilizing transfer learning. The applied model for transfer learning, EfficientNet, is a systematic scaling way of CNN (convolution neural networks) in which three dimensions of networks are expanded and unified. However, parameters were found to be frozen when some errors occurred with the original CNN model. Therefore, optimization to prevent overfitting and realize rapid operation is detailed in this paper. On the other hand, a method of loss control for sample imbalance is proposed and demonstrated which somehow has an edge over the traditional method of cross entropy. Specifically, it functions by involving additional variables. Through practical experiments, the final accuracy was around 90%.**

***Keywords*-pneumonia; image classification; convolutional neural networks; transfer learning;**

I. Introduction

Coronavirus disease 2019 (COVID-19), which originates from severe acute respiratory syndrome coronavirus 2, has influenced numerous countries and people. According to network statistics [1], there have been more than 600 million Coronavirus cases since its outbreak. Many patients suffered from acute lung injury (ALI), cytosine storm, and other diseases and complications, and died in severe cases. Consequently, medical treatments, including vaccines and drugs, are developed to prevent and suppress the virus respectively. While medical experts are making effort to alleviate human suffering, engineers are also contributing to the fight against COVID-19.

Remarkable innovations in deep learning and the growing number of large annotated medical image datasets are stimulating dramatic advances in the automated understanding of medical images. In the past several decades, computer-aided diagnosis (CAD) has bloomed in the field of medical imaging. Granted, computer diagnosis cannot completely replace manual diagnosis, CAD with the purpose of reducing error and enabling more efficient measurement is able to lessen the workload and stress for medical staff nevertheless.

In the case of COVID-19, CNN technology can be used in the diagnosis of chest X-ray images for pneumonia patients by establishing a CAD scheme. In view of an imbalanced dataset, pretreatment for the dataset is in need, which is the aim of data augmentation here. Besides, transfer learning can be applied in this CAD in order to expedite the processing, making the model fit the data better. And focal loss function can be also effectively utilized in controlling the loss when dealing with sample imbalance.

The remainder of this paper is structured as follows. Section II lists the related works in this field. Methods used in the research are detailed in Section III. Section IV is dedicated to showing the whole process of the experiment, analyzing the results, and making further discussion.

II. Related Work

A lot of work has been done on deep learning in the medical field, one important application of image recognition is pneumonia detection with chest X-ray images. Vikash et al introduced the concept of transfer learning in the field of deep learning using the pre-trained ImageNet models and their ensembles for the detection of pneumonia [2]. Vikash Chouhan proposed an integrated model composed of many individual models. Wang et al classified and localized thoracic diseases with a large dataset of a hospital scale [3].

M.E.H Chowdhury et al. compared four algorithms, DenseNet201, ResNet18, SqueezeNet, and AlexNet, for transfer learning [4]. Y.Q.Feng et al proposed the Deep Supervised Domain Adaptation (DSDA) to detect pneumonia from chest X-ray images automatically [5]. G. Labhane constructed four models, VGG16, VGG19, InceptionV3, and a basic convolutional neural network using CNN and transfer learning methodologies [6].

III. Methodology

Over-fitting is a potential problem when training a large neural network on medical images with limited training samples. In this paper, we thus proposed a CNN network, which is specially designed for medical images like chest x-ray and try to avoid over-fitting on imbalanced data.

The task of this paper is to distinguish patients with pneumonia from those with normal by a binary classification using transfer learning from EfficientNet, which provides a more convenient way to diagnose and prevent the spread of COVID-19. Section III-A indicates the dataset of chest X-ray used in the model. And in Section III-B, we adopt some data augmentation methods to enhance the input images. The main framework of the model is presented in section III-C. Finally, in section III-D, we introduced the focal loss function to solve the class imbalance problem.

A. Dataset

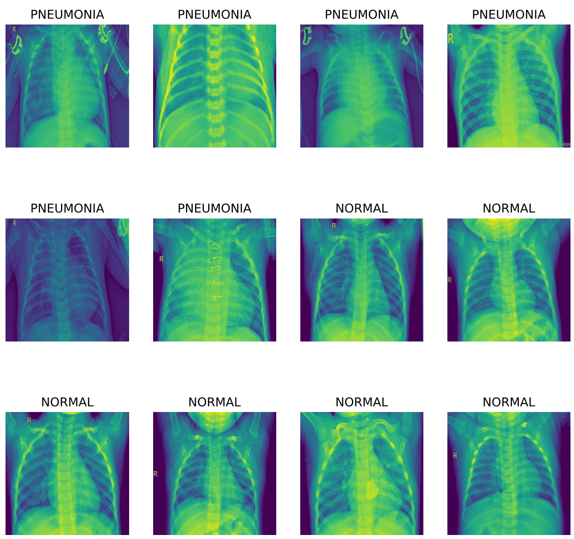


Figure 1. Examples of dataset.

The experiment was conducted on a public pneumonia chest x-ray dataset on Kaggle website [7]. It contains 5216 images of chest x-ray, including normal and pneumonia categories, which come from pediatric patients from Guangzhou Women and Children's Medical Center, Guangzhou.Sample images are shown in Fig.1.The distribution of images to each category in this data set is listed in Table I.

In order to train a convolutional network model, the data needs to be classified and divided into training, test, and validation sets. Since the original validation set in the dataset was too small, we re-divided the three sets to 70%, 10%, and 20% respectively.

Table I
DISTRIBUTION OF THE DATASET.

Dataset	Classes	Train	Test	Val
Raw Data	Normal	1341	234	8
	Pneumonia	3875	390	8
	Total	5216	624	16
After Split	Total	4173	624	1043

Noticing that the number of pneumonia and normal samples are extremely imbalanced for training and testing, which makes the results of the model tendentious.

To improve the generality of the model, we will try to eliminate the imbalance by using data augmentation, transfer learning, and focal loss function. We will describe these methods in the following sections.

B. Data Augmentation

When a neural network works with limited samples, it is easy to be affected by the distribution of training data. Data augmentation is a common solution to reduce overfitting and enhance the generalization ability by artificially enlarging the dataset. In this section, we input images after augmentation into the CNN model before training.

First, we resize all of the images into 456*456*3, and then we apply the following augmentation techniques to generate more samples for training the model, including translation, zoom, rotation, scaling, and flipping. We also randomly adjust the brightness and contrast of images. These steps are finished by keras image augmentation layers, and the setting parameters are listed in Table II. We also give examples of the corresponding relationship between the generated and the original,as shown in Fig 2

Table II
PARAMETERS FOR RANDOM IMAGE AUGMENTATION.

Augmentation	Parameter
Random Rotation	0.2
Random Contrast	0.2
Random Zoom	0.2
Random Brightness	0.2
Random Translation	0.1 0.1
Random Horizontal Flip	True
Random Vertical Flip	True

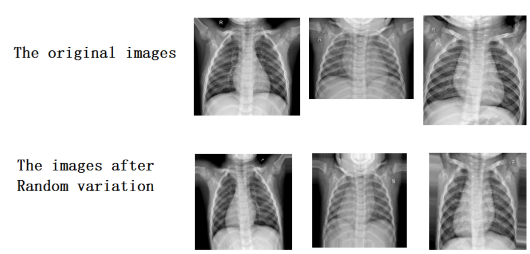


Figure 2. Examples of dataset.

C. Network Architecture

We used model EfficientNet [8] for transfer learning, which is presented by Google in 2019. EfficientNet compresses the standard convolution by depthwise separable convolution and increases the depth of the neural network by increasing the residual network, enabling deeper neural networks to extract features and reduce the number of parameters Between the whole series of the EfficientNet, from EfficientNetB0 to B7, the parameters of width, depth and resolution are different. And we found EfficientNetB5 is the most suitable.EfficientNetB5 contains 38 MBConvBlock.But compared to the model with similar accuracy, it is about eight times smaller and can be more efficient.

Table III
THE STRUCTURE OF THE PROPOSED NETWORK.

Layer (type)	Output Shape	Param
input_2 (InputLayer)	[(None, 150, 150, 3)]	0
sequential (Sequential)	(None, 150, 150, 3)	0
efficientnetb5 (Functional)	(None, 5, 5, 2048)	28513527
global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0
dropout (Dropout)	(None, 2048)	0
dense_1 (Dense)	(None, 512)	1049088
dropout_1 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 64)	32832
dropout_2 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 32)	2080
dense (Dense)	(None, 1)	33
Total params: 29,597,560 Trainable params: 1,084,033 Non-trainable params: 28,513,527		

In addition, the EfficientNetB5 is not a simple stack of different kinds of layers, on the contrary, it optimizes the model by balancing the relationship between network depth(the number of the layers), network width(number of channels at each layer) and resolution of the feature map. Although magnifying any dimension of these three can improve the accuracy, the benefit of it for the large model is reduced a lot. So it is the main point of this model that uses the method named compound scaling to unify the three dimensions together with a contrast ratio to find the balance.

In transfer learning, the parameters of the pre-trained model are frozen, which is untrainable and will not be updated. The basic idea is to make the model learn basic low-level features of the image from a large dataset, and learn high-level features from the specific training data.The earlier layers extract basic features and require no or less fine-tuning. So in the present work, we only need to fine-tune the last added layers of EfficientNet. Therefore, we use EfficientNetB5 for transfer learning. To make the model more suitable for our data, we cut the last layers from the original model and added dense layers at the end.

Firstly, we added a global average pooling layer. Then we used a series of dense layers as connecters. In addition, Relu is selected as activation function. Dropout layers are added at 20% rate to prevent the model from overfitting. The ouput dense layers using sigmoid function for classification task. Therefore, the parameters in these layers can be updated constantly to make a more optimal model. The whole framework can be found in Table III, which details the layers and parameters of the model.

D. *Focal loss function*

Since the number of normal samples is far less than that of pneumonia samples, we decided to solve the data imbalance problem by using focal loss(FL) rather than cross entropy(CE). Focal loss function was proposed [9] to address the imbalance between foreground and background classes in object detection. but it is also available for classification problem with imbalanced data set [10]. Focal loss adds different weights of categories and weights of difficult samples on the basis of cross entropy loss function to improve the model learning effect.

The binary cross-entropy loss function for classification is defined as

$$CE = \begin{cases} -\log(p), & \text{if} \quad y = 1 \\ -\log(1-p), & \text{if} \quad y = 0 \end{cases} \tag{1}$$

where y is ground-truth label and p is the estimated possibility of the positive sample.

The equation of focal loss function is defined as

$$FL = \begin{cases} -\alpha(1-p)^{\gamma}\log(p), & \text{if} \quad y = 1 \\ -(1-\alpha)p^{\gamma}\log(1-p), & \text{if} \quad y = 0 \end{cases} \tag{2}$$

α and γ indicate how sensitive it is to the easily classified samples. Weighting factor α is introduced to balance the importance of positive/negative samples. It is defined to be the rate of the percentage of hard samples in the total sample. so the value of α in this experiment is 1341/5216. Tunable focusing parameter γ is used to control the loss based on the difficulty of the model prediction. It smoothly adjusts the rate at which easy examples are down-weighted by measuring the contribution of simple samples and hard samples to the total loss.

IV. EXPERIMENT AND RESULTS

A. *Experimental Setup*

All experiments were run on two Nvidia GeForce RTX 2080Ti GPUs. We implement the proposed CNN model on Tensorflow and Keras. It is essential to avoid overfitting in the model training step. It means that all models tend to be sufficiently trained to get as close as impossible to the results. So if the accuracy of the training database keeps increasing when it is higher than a balanced point, the loss will also rise instead of decline and the accuracy of evaluation will drop quickly. Besides the methods we mentioned in Section III, we control two hyperparameters in the experiment step.

The initial epoch is set as 100 to make sure the model is convergent entirely, but we used EarlyStopping to stop training at an appropriate time by the declining speed of the validation loss. As shown in Fig.4, the training was stopped due to the absence of changes in both accuracy and loss after about 20 epochs.

The initial learning rate is set as 0.01, with the help of LR finder [11]. Then learning rate decay controls the learning speed dynamically. The standard of its judgment is the convergence rate of validation loss. It means we can set a relatively fast speed initially to make the model convergent rapidly. The learning rate will decay to 30 percent of the last status each time in the next epoch when it detects no changes. After setting learning rate, we use Adam as optimizer for experiments.

B. Metrics and Results

We first evaluate our CNN model based on EfficientNetB5. Confusion matrix shown in Fig 3 gives a picture of classification results, indicating the comparison results between the predicted and label. 4 indicates both loss and accuracy on train and validation dataset.

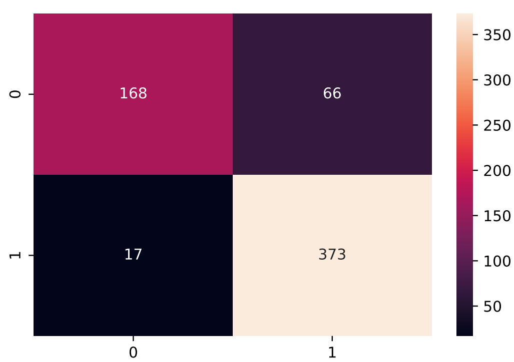
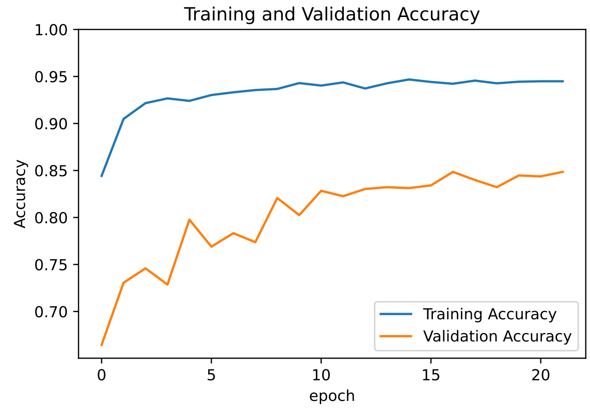


Figure 3. Confusion matrix of the model.

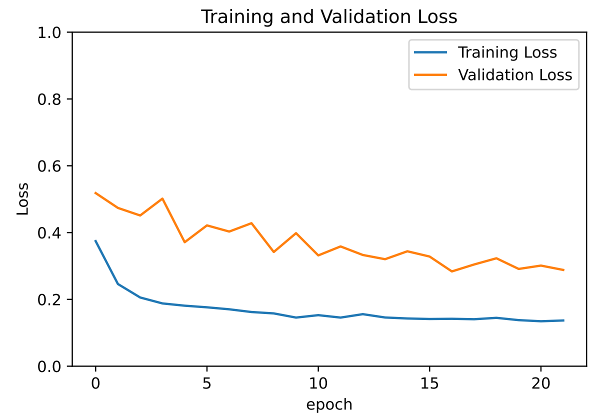
Besides the EfficientNetB5, we also tried some other models. We make a comparison between different pre-trained networks, VGG19, ResNet50, and DenseNet121. All models are with the same architectures except the transfer learning model.

After training, their accuracy on validation sets are 92.04%, 95.01%, 86.58% and 94.14% for VGG19, ResNet50, DenseNet121 and EfficientNetB5 respectively. It seems ResNet50 and EfficientNetB5 are both suitable base models for this task. Then we evaluate these models on test data set, which has not been used in any training stage.

To better evaluate the performance of classification models on an imbalanced dataset, more performance metrics in the field are introduced in the model test step, including precision, recall, and f1 Score, which can be more important in practical applications. They are mathematically defined in Equation 3.



(a) accuracy



(b) loss

Figure 4. Accuracy and loss of the proposed model on train and validation data.

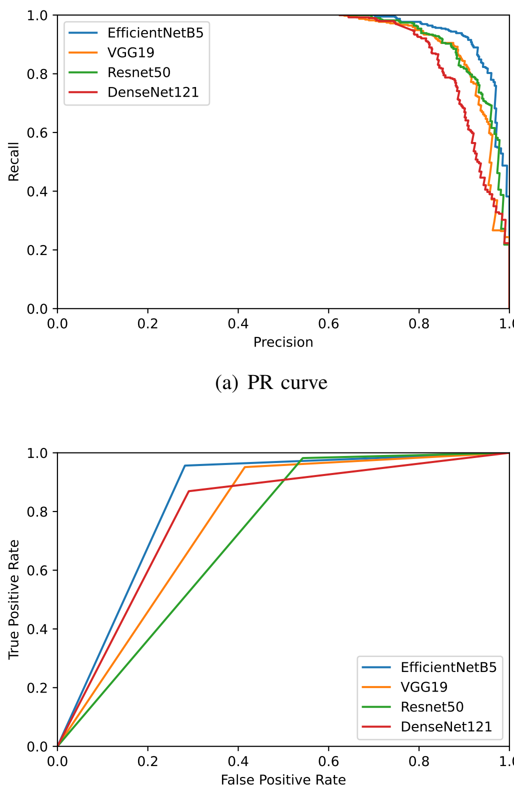
$$
\begin{aligned}
Accuracy &= \frac{TP+TN}{TP+TN+FP+FN}, \\
Precision &= \frac{TP}{TP+FP}, \\
Recall &= \frac{TP}{TP+FN}, \\
F1\text{-}score &= \frac{2 * Precision * Recall}{Precision + Recall}
\end{aligned} \tag{3}
$$

where TP is True positive, FN is False Negative, TN is True Negative, and FP is False Positive. Specially, ROC curve draws TP and FP at different thresholds. AUC calculates the area under the ROC Curve, providing an aggregate measure of performance across all possible classification thresholds.

Then we can calculate these metrics,and the results of different models are presented in Table IV. Based on previous results, we draw the PR curves and ROC curves in Fig 5. EfficientNet based CNN model achieve best performance with the highest percentage of accuracy, precision, auc and f1-score. It is obviously that the EfficientNetB5 is the most suitable for our task.

Table IV
COMPARISON RESULTS OF DIFFERENT PRE-TRAINED MODELS ON TEST DATA.

	Accuracy	Precision	Recall	AUC	F-1 Score
VGG19	0.814	0.793	0.951	0.910	0.865
ResNet50	0.785	0.751	**0.982**	0.921	0.851
DenseNet121	0.809	0.833	0.869	0.879	0.851
EfficientNetB5	**0.899**	**0.896**	0.949	**0.957**	**0.900**



(a) PR curve

(b) ROC curve

Figure 5. Result curve of models on test data.

V. CONCLUSION

This paper showed that by using EfficientNet5, the accuracy of detecting pneumonia could reach around 90%, which is remarkably higher than other transfer learning models in the same situation. Future research can continue to explore different types of data imbalance and apply more methods to avoid overfitting and improve judgment accuracy.

ACKNOWLEDGMENT

Li Wenyu, Wei Shujin, Bai Yunling, and Song Zikang thank Prof. Teoh Teik Toe, the director of the AI lab at Nanyang Technological University. We express our deepest gratitude to him for his inculcation and instruction. Without his careful guidance, this paper would not have reached its present form.

REFERENCES

[1] "COVID Live - Coronavirus Statistics - Worldometer," https://www.worldometers.info/coronavirus/.

[2] V. Chouhan, S. K. Singh, A. Khamparia, D. Gupta, P. Tiwari, C. Moreira, R. Damaševičius, and V. H. C. De Albuquerque, "A novel transfer learning based approach for pneumonia detection in chest X-ray images," *Applied Sciences*, vol. 10, no. 2, p. 559, 2020.

[3] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2097–2106.

[4] M. E. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, and N. Al Emadi, "Can AI help in screening viral and COVID-19 pneumonia?" *IEEE Access*, vol. 8, pp. 132 665–132 676, 2020.

[5] Y. Feng, X. Xu, Y. Wang, X. Lei, S. K. Teo, J. Z. T. Sim, Y. Ting, L. Zhen, J. T. Zhou, and Y. Liu, "Deep Supervised Domain Adaptation for Pneumonia Diagnosis From Chest X-Ray Images," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 3, pp. 1080–1090, 2021.

[6] G. Labhane, R. Pansare, S. Maheshwari, R. Tiwari, and A. Shukla, "Detection of pediatric pneumonia from chest X-ray images using CNN and transfer learning," in *2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE)*. IEEE, 2020, pp. 85–92.

[7] "Chest X-Ray Images (Pneumonia)," https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia.

[8] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6105–6114.

[9] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[10] G. S. Tran, T. P. Nghiem, V. T. Nguyen, C. M. Luong, and J.-C. Burie, "Improving accuracy of lung nodule classification using deep learning with focal loss," *Journal of healthcare engineering*, vol. 2019, 2019.

[11] L. N. Smith, "Cyclical learning rates for training neural networks," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2017, pp. 464–472.