ISSN No:-2456-2165

Pneumonia Classification for Chest X-rays Using Fine Tuning and Transfer Learning

Arif Bagde
Department of Computer Engineering
K.J. Somaiya College of Engineering
Mumbai, India

Prasanna Shete
Department of Computer Engineering
K.J. Somaiya College of Engineering
Mumbai, India

Abstract:- Pneumonia is a lung infection mainly caused by microbes where lungs become inflamed and tiny air sacs (alveoli) get filled with fluids causing difficulty in breathing. As stated by the World Health Organization (WHO), pneumonia is the single largest infectious cause of death in children worldwide accounting for 15% of all deaths of children under five years old. While young and healthy adults have low risk, older people have a greater chance of having pneumonia and are much more likely to die from it. The most convenient way to diagnose pneumonia is through chest x-rays. Deep Learning has shown some tremendous results in medical image analysis in recent times. Convolution Neural Networks (CNNs) are widely used in various classification problems starting from handwritten digit recognition to self-driving cars. However, training a CNN model from scratch could be a tedious task as it requires a huge labeled training data, extensive computational resources for training the model, and it often leads to overfitting and convergence issues. Hence, a convenient alternative for traditional CNN is to fine-tune a pre-trained CNN that has been trained using a large dataset. In this paper, we present the performance analysis of transfer learning and fine-tuning CNN for classifying pneumonia among the chest x-ray samples. Our proposed Fine-Tuned CNN model classifies pneumonia infected chest x-rays into 3 categories bacterial, normal, and viral achieves an accuracy of 83.33% which is comparable to the performance of human radiologists.

Keywords:- *Convolution Neural Network, Fine-Tuning, Transfer Learning, Chest X-rays, Medical Imaging.*

I. INTRODUCTION

Pneumonia is one of the leading causes of death among children and elderly people worldwide; is a lung infection caused by viruses, bacteria, and fungi that results in swelling of the lungs, which can be a life-threatening situation if not diagnosed in time. Chest x-rays are an important method for diagnosing pneumonia. Much research work has been carried out for identifying chest and lung disease using Artificial Intelligence and Machine Learning (ML). Deep learning is a sub-field of ML that has emerged with some excellent results in the field of health informatics. Among several deep learning algorithms, CNN has had a major impact in this domain [10]. CNNs have been used not only in the field of computer vision but across various applications ranging from natural language processing to hyperspectral image processing and medical image analysis [1]. CNN has proven very successful in solving image classification problems. One of the greatest advantages of neural networks is the generalization quality to solve different kinds of problems using similar architecture [9]. However, there are certain limitations of deep learning. The deep learning models cannot be interpreted easily because it is considered as a black box by the researchers without explaining how it provides good results. A common problem that can arise during model training is overfitting when the number of parameters equals the total number of samples in the training set. The network memories the training samples, but cannot generalize new samples. Another aspect is, raw data cannot be used directly as input for a neural network. Thus, preprocessing, normalization or input change is often required before training [6]. One of the major challenges in using deep learning in the medical domain is the scarcity of training data due to obstruction in collecting and labeling that requires expert knowledge. To overcome these problems, fine-tuning and transfer learning is introduced [6, 12, 14].

Fine-tunned CNN in the context of medical imaging is the best alternative to training a CNN from scratch. The performance of pre-trained CNN with shallow fine-tuning results in lower performance, while deeper fine-tuning results in superior performance to that of CNN trained from scratch [1]. For CAD and other medical imaging tasks, CNN training from scratch is often not possible. However, generic features can be adopted from CNN that have already been trained [13]. Ilyas Sirazitdinov et al. [2] present their work on identifying and localizing pneumonia on chest x-ray images by combining two CNNs, namely ResNet and Mask R-CNN. In [4] A similar type of study on brain tumor classification has been done using a block-wise transfer learning and fine-tuning strategy. In our work, we aim to classify pneumonia infected chest x-rays by utilizing fine-tuning and transfer learning techniques. Identifying pneumonia could require a significant amount of time and proper expertise to make a diagnosis. In our work, we present a solution that allows the medical officials to identify the type of pneumonia on the chest x-ray images with high precision and greater accuracy.

II. RESEARCH METHOD

A. *Dataset*

In this study, we used a publicly available dataset from 'Guangzhou Women and Children's Medical Center' which contains chest x-rays of patients aged one to five years. All chest x-rays imaging was performed as part of routine clinical patient care. The dataset was labeled by two specialists and re-evaluated by the third doctor to avoid scoring errors [11, 23]. The dataset was initially divided into 2 categories 'pneumonia' and 'normal', later we subdivided the dataset into 3 categories 'bacterial', 'viral', 'normal' according to their labels, see Fig.1. Physicians named the images in the dataset by specifically identifying the type of pneumonia. For example, the 'normal' category image was named as 'NORMAL2-IM-0376-0001'. Likewise, the 'bacterial' category image was named as 'person25_bacteria_120', and the 'viral' category image was named as 'person267_virus_552', this allowed us to divide the dataset into 3 categories. After division, the images labeled 'bacterial' were 2774, the images labeled 'viral' were 1490, and images labeled 'normal' were 1575. Considering the inconsistency in the number of images, we decided to train the model on 1000 randomly selected images from each category i.e. a total of 3000 chest x-ray images.

B. *Pre-trained CNN*

CNN architecture is mainly composed of 3 types of layers: convolutional layer, pooling layer, and fully connected layer. The convolution and the pooling layer perform feature extraction while the fully connected layer maps the features to a final output [7, 14, 15]. A pre-trained CNN model used in this experiment is a 14-layer model with more than 8 million training parameters. This model pre-trained model contains a total of 4 blocks with 9 layers of folding, 3 layers with maximum grouping, and 2 fully connected layers. 3 folding layers and 1 maximum grouping layer each, while the last block contains fully connected layers. A Convolution layer is a primary component of CNN architecture that performs feature extraction. It has a small matrix of numbers called kernels/filters and it is applied to the input image to extract the features from it [3]. The output of a convolution layer is called a 'feature map'. Features are extracted using 3 x 3 size filters. It extracts features, the ReLu activation function incorporates non-linearity into the system. The max-pooling layer helps to reduce the dimensionality of the feature maps. After processing through the first 3 blocks the feature map of the last max-pooling layer is flattened i.e., converted into a 1D array or a vector, and mapped onto the fully connected layer, also known as the dense layer. It maps the features extracted by the previous layers to each node present in the current layer. The output of dense layers is then connected to the final classification layer with the activation function 'softmax', which normalizes the output values of the last fully connected layer to the probabilities of the target class, with each value ranges between 0 and 1 [7]. The output of this function is a probability vector.

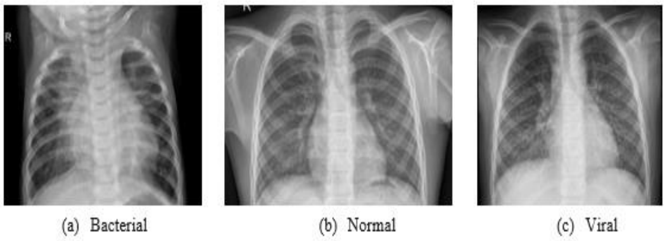

(a) Bacterial (b) Normal (c) Viral

Fig. 1. Randomly selected chest x-ray images from the dataset of each category. These are from 'Guangzhou Women and Children's Medical Center' The diagnoses for the images were then graded by two expert physicians before being cleared for training the AI system.

As discussed earlier CNN model is prone to overfitting. Overfitting refers to a situation where a model learns certain statistical regularities from the training set, i.e. instead of learning the signal, it stores the irrelevant noise and therefore performs less well on a new dataset [7]. To overcome this problem, 'dropout' layers are introduced. Dropout layers randomly turn off a certain number of neurons from the previous layer. This is a commonly used regularization technique to avoid overfitting of the layers.

C. *Fine-tuning CNN*

Training a CNN model with pre-trained weights is called fine-tuning. It begins with transferring the weights from a pre-trained network to a new network, also known as transfer learning. The exception is the last fully connected layer, the number of nodes of which depends upon the number of classes in the dataset. A common practice is to replace the last fully connected layer of a previously trained CNN with a new fully connected layer that has as many neurons as the application contains classes [1]. In our case, a pre-trained model has 2 classes, while a new model requires 3 classes; Hence the new fully connected last layer has 3 neurons. We tried to discover the difference between the classification performance by adding more dense layers to the model, we fine-tuned the model by replacing the entire final block i.e. Block 4, see Fig. 2.

Generally, the earlier layers of CNN learn the low-level features such as curves and edges, whereas the later layers learn more specific or high-level features, therefore, the learning of earlier layers is frozen [4]. To learn the domain-specific features of pneumonia from chest x-rays, we train the model by applying the strategy of layer-wise fine-tuning. The entire proposed model architecture can be seen in Fig. 2.

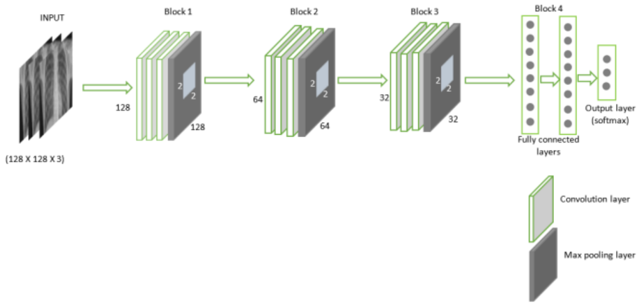


Fig. 2. The proposed fine-tuned CNN model. The final block was replaced with 3 fully connected layers and layer with 3 classes according to classes in the dataset. The white blocks refer to convolution layers; grey blocks refer to the Max-pooling layer. The blue box on the Max-pooling layer refers to a 2x2 kernel.

III. RESULT AND ANALYSIS

The performance of a proposed system has been evaluated by dividing the chest x-ray data into 3 subsets of equal size and separating it into 2 different configurations as shown in Table I. Experiments are carried out by using 3 different optimizers Adam, SGD, and RMSprop. Total 48 test runs are performed with 3 optimizers based on two different learning rates, refer to Table II.

TABLE I. DATA SEPARATION USED FOR PERFORMING EXPERIMENTS WITH THE MODEL.

Sr.no.	Training data	Validation data
1.	80% - 2400 images	20% - 600 images
2.	60% - 1800 images	40% - 1200 images

The proposed method was implemented on the Open-Source platform Google Colab. It is a cloud-based platform that gives an online jupyter notebook to execute AI projects to the users. Google Colab provides with 2 core Intel Xeon processor @ 2.3 GHz, up to 12GB GDDR5 GPU (scales depending upon requirement).

TABLE II. DIFFERENT OPTIMIZERS USED FOR PERFORMANCE ANALYSIS OF THE MODEL.

Sr.no.	Optimizers	LR – 1	LR – 2
1.	SGD	0.001	0.0001
2.	Adam	0.001	0.0001
3.	RMSprop	0.001	0.0001

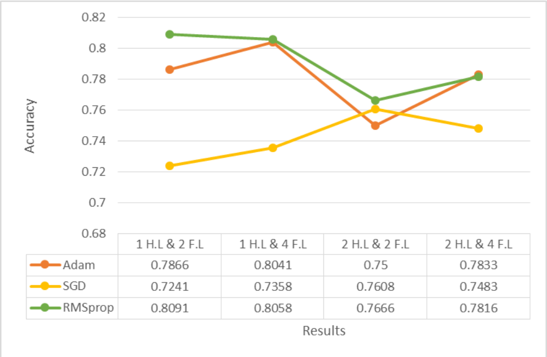
A. Training

The training and fine-tuning of a proposed network take around 25 – 30 minutes depending upon the parameters. To obtain the optimal result we monitored the training – validation accuracy and loss. The training of a network is limited to 100 epochs as there was no improvement seen for validation accuracy and loss after several epochs. The classification results obtained by performing various experiments are present in Table III, Table IV, Table V, and Table VI.

All the experiments are performed considering the combination of one hidden layer with freezing two or four pre-trained layers and similarly, two hidden layers with freezing two or four pre-trained layers. We used a trial and error based approach to determine Learning Rate (L.R.) values to 1e-3 (0.001) & 1e-4 (0.0001) and performed experiments with different parameters, refer to Table I & II. It has been seen in every set of experiments; Adam and RMSprop have presented superior results whereas SGD struggle in some cases because the optimal L.R value for SGD is 0.01. If we set the L.R. value as small or very large, then the model fails to converge and result in overfitting.

TABLE III. SHOWS THE RESULTS OBTAINED BY USING L.R. 1E-3 WITH 60% - 40% DATA SPLIT.

Config.	Optimizers	H.L	F.L	ACC.	Loss
60-40	Adam	1	2	0.7866	0.4954
60-40	Adam	1	4	0.8041	0.4953
60-40	Adam	2	2	0.75	0.5796
60-40	Adam	2	4	0.7833	0.5011
60-40	SGD	1	2	0.7241	0.6638
60-40	SGD	1	4	0.7358	0.6569
60-40	SGD	2	2	0.7608	0.5832
60-40	SGD	2	4	0.7483	0.5950
60-40	RMSprop	1	2	0.8091	0.4654
60-40	RMSprop	1	4	0.8058	0.5076
60-40	RMSprop	2	2	0.7666	0.5288
60-40	RMSprop	2	4	0.7816	0.5203



	1 H.L & 2 F.L	1 H.L & 4 F.L	2 H.L & 2 F.L	2 H.L & 4 F.L
Adam	0.7866	0.8041	0.75	0.7833
SGD	0.7241	0.7358	0.7608	0.7483
RMSprop	0.8091	0.8058	0.7666	0.7816

Fig. 3. Model training results for 60% - 40% data split & learning rate – 1e-3. It shows the graphical representation of the data in Table III. It clearly shows that the RMSprop outperformed the other two optimizers by achieving some great results. A total of 4 operations using each of the optimizers are performed and it can be observed that the SGD in this study showed some very inferior results as compared to the other two, whereas the Adam optimizer performed identically to RMSprop in most of the cases.

TABLE IV. SHOWS THE RESULTS OBTAINED BY USING L.R. 1E-3 WITH 80% - 20% DATA SPLIT.

Config.	Optimizers	H.L	F.L	ACC.	Loss
80-20	Adam	1	2	0.7966	0.5062
80-20	Adam	1	4	0.7799	0.5334
80-20	Adam	2	2	0.7816	0.5051
80-20	Adam	2	4	0.7783	0.5281
80-20	SGD	1	2	0.7300	0.6770
80-20	SGD	1	4	0.75	0.6464
80-20	SGD	2	2	0.7383	0.5725
80-20	SGD	2	4	0.7549	0.5692
80-20	RMSprop	1	2	0.7850	0.5174
80-20	RMSprop	1	4	0.8333	0.4745
80-20	RMSprop	2	2	0.8033	0.4854
80-20	RMSprop	2	4	0.8100	0.4438

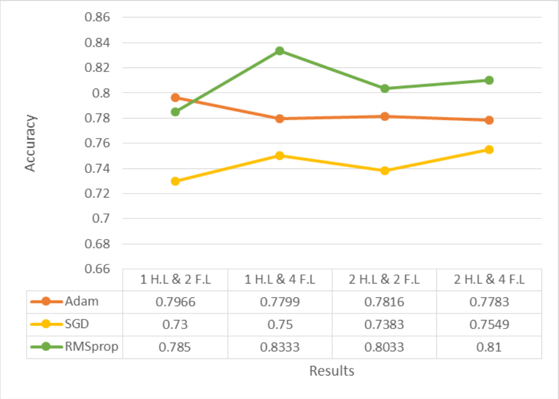


Fig. 4. Model training results for 80% - 20% data split & learning rate – 1e-3. It shows the graphical representation of the data in Table IV. The RMSprop optimizer showed a better result in 80% - 20% data split in most cases. It can be observed that the RMSprop once again outperformed the other two optimizers. Adam achieved some comparable results to RMSprop, whereas SGD achieved minimal results among the three.

Our choice of L.R. showed optimal results. We decrease the L.R. after every two epochs if no increase in accuracy is seen to prevent the network from overfitting.

TABLE V. SHOWS THE RESULTS OBTAINED BY USING L.R. 1E-4 WITH 60% - 40% DATA SPLIT.

Config.	Optimizers	H.L	F.L	ACC.	Loss
60-40	Adam	1	2	0.7866	0.5374
60-40	Adam	1	4	0.7925	0.5587
60-40	Adam	2	2	0.8075	0.4615
60-40	Adam	2	4	0.7858	0.4835
60-40	SGD	1	2	0.5291	0.9929
60-40	SGD	1	4	0.4541	1.0428
60-40	SGD	2	2	0.5233	1.0249
60-40	SGD	2	4	0.4875	1.0383
60-40	RMSprop	1	2	0.7741	0.5229
60-40	RMSprop	1	4	0.7925	0.5058
60-40	RMSprop	2	2	0.7741	0.5883
60-40	RMSprop	2	4	0.7933	0.5218

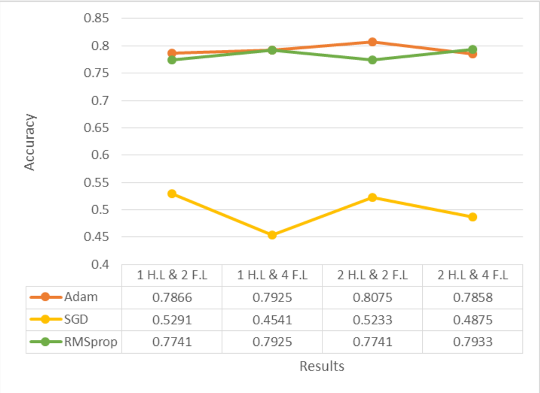


Fig. 5. Model training results for 60% - 40% data split & learning rate – 1e-4. It shows the graphical representation of the data in Table V. It can be observed that, as the L.R. was decreased, there was a huge drop in the performance of SGD. On the other hand, Adam achieved even more comparable results to RMSprop. Moreover, in this particular experiment, Adam optimizer ever achieved a better result than RMSprop with 80% accuracy.

TABLE VI. SHOWS THE RESULTS OBTAINED BY USING L.R. 1E-4 WITH 80% - 20% DATA SPLIT.

Config.	Optimizers	H.L	F.L	ACC.	Loss
80-20	Adam	1	2	0.7966	0.5170
80-20	Adam	1	4	0.8000	0.5023
80-20	Adam	2	2	0.8149	0.4538
80-20	Adam	2	4	0.8249	0.4473
80-20	SGD	1	2	0.5216	1.0076
80-20	SGD	1	4	0.4783	1.0179
80-20	SGD	2	2	0.5916	0.9035
80-20	SGD	2	4	0.6949	0.7916
80-20	RMSprop	1	2	0.8000	0.4910
80-20	RMSprop	1	4	0.8333	0.4599
80-20	RMSprop	2	2	0.7850	0.5196
80-20	RMSprop	2	4	0.8283	0.4614

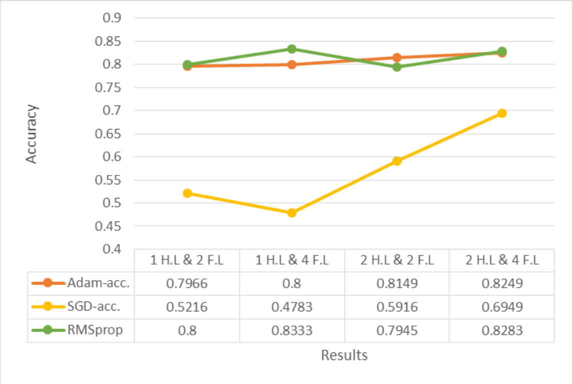


Fig. 6. Model training results for 80% - 20% data split & learning rate – 1e-4. It shows the graphical representation of the data in Table VI. It can be observed that, as the L.R. was decreased, there was a huge drop in the performance of SGD. On the other hand, Adam achieved even more comparable results to RMSprop. In this experiment, RMSprop achieved the highest accuracy of 83.33%.

B. *Performance Analysis*

We evaluated the performance of our model based on precision, sensitivity/recall, f1-score, specificity, and accuracy.

$$\text{Precision} = \frac{\text{TP}}{\text{TP + FP}} \quad (1)$$

$$\text{Sensitivity / Recall} = \frac{\text{TP}}{\text{TP +F N}} \quad (2)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN + FP}} \quad (3)$$

$$\text{F1-score} = 2* \frac{\text{Precison} * \text{Recall}}{\text{Precision + Recall}} \quad (4)$$

$$\text{Accuracy} = \frac{\text{TP + TN}}{\text{TP + TN + FP + FN}} \quad (5)$$

Sensitivity/Recall is the true positive ratio (TP) correctly classified by the model that describes how accurately the classifier classifies the correct category of a chest x-ray. Specificity is the true negative ratio (TNR) shows how accurately the classifier predicts the negative condition. Precision is a positive predictive rate (PPR). F1-score measures classification performance in terms of recall and precision. Accuracy is the overall classification accuracy in terms of TP and TN of the proposed method [4].

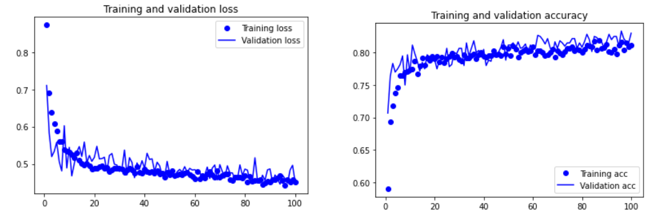


Fig. 7. Training progress: The accuracy and loss of the best performing model. Fig.7(a) represents training and validation loss. Fig.7(b) represents training and validation accuracy.

Fig. 8.

TABLE VII. AVERAGED VALIDATION MATRIX OF BACTERIAL, NORMAL, AND VIRUS CHEST X-RAYS.

	Precision	**Recall**	**Specificity**	**F1-score**
Bacterial	0.72	0.90	0.82	0.80
Normal	0.94	0.94	0.97	0.94
Viral	0.88	0.66	0.95	0.75

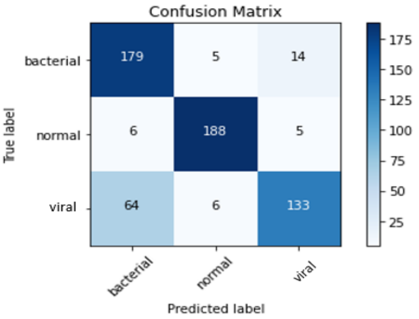
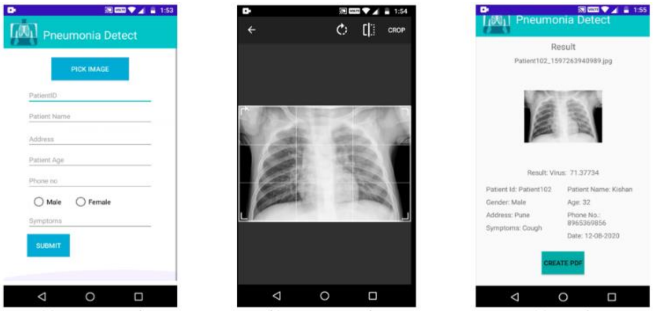


Fig. 9. Represents the confusion matrix of the best performing model. The confusion matrix generally represents the performance of the model on a set of test data for which true values are known. Accuracy, Precision, Recall, Specificity, and f1-score can be calculated from the confusion matrix using the equations (1, 2, 3 & 4).

Table VII. shows the individual class precision, recall, specificity, and the f1 score, which were calculated using the validation data set. The results are calculated using equations (1, 2, 3, and 4) showed the highest score for all (precision, recall, specificity & f1-score) among the three classes. The total accuracy achieved by the model is 83.33%. However, to improve the performance of the trained model, more data on pneumonia infected chest x-rays with 3 classes (‘bacterial’, ‘normal’, & ‘viral’) is needed, which was a major challenge in this study. A perfectly labeled high quality dataset would help improve performance. Increasing the training data would eventually increase the accuracy of the model.

We assessed the classification performance of the model using different configurations and parameters and presented the results in tables and figures. Fig.7 represents the results of the model with the best performance. The good thing about the transfer learning and fine-tuning method is, it reduces the overfitting and speeds up the convergence. The early layers in CNN learn the low-level features and the later layers learn the high-level or domain-specific features. The dataset we used is chest x-ray images similar to those in the pre-trained model. We tried fine-tuning the model in two ways, firstly by replacing only the final classification layer and secondly by replacing the entire final block i.e. Block 4, see Fig. 2. with the addition of hidden layers. Our model struggled to perform with SGD optimizer in some cases as it shows a decrease in classification performance especially when the L.R. is set to 1e-4. The results obtained with SGD were not comparable with the other two optimizers. The results achieved by the Adam optimizer and RMSprop optimizer were identical in almost every experiment. Our fine-tuned CNN model achieved the highest classification accuracy of 83.33% using the RMSprop optimizer shown in Table VI.



Result

Patient102_1597263940989.jpg

Result: Virus: 71.37734

Patient Id: Patient102

Patient Name: Kishan

Gender: Male

Age: 32

Address: Pune

Phone No.: 8965369856

Symptoms: Cough

Date: 12-08-2020

(a) New Record (b) Image cropping (c) Result

Fig. 10. Application Screenshots. Fig. 9(a). Represents the form to add a new patient record. Fig. 9(b). Represents image cropping activity to avoid unwanted backgrounds. Fig. 9(c). Represents the results activity.

The best trained model is integrated into an android application that will help medicals officials and doctors to predict the disease much faster. The users can add new patient details, select the chest x-ray image of the patient by internal storage, or by clicking the image from the camera as shown in Fig. 9(a). The application allows the users to select the image from the device, cloud storage, etc., and perform image cropping operation to avoid unwanted backgrounds that will affect model prediction as shown in Fig. 9(b). The users can add details like Patient id, Patient name, City, Age, Phone no, Gender, and the symptoms that the patient is having. The further step is to submit the data. After submitting the data, it will store on the Firebase Firestore database and the chest x-ray image will be given as an input to the model which will return the prediction result. The next activity displays the result containing the chest x-ray image, predicted result, and patient details as shown in Fig. 9(c). the application also allows the user to generate a pdf of the result that can be shared with doctors and physicians.

IV. CONCLUSION

In this paper, we proposed a method for pneumonia chest x-ray classification using transfer learning and fine-tuning. In this proposed strategy, the performance analysis has been done by applying different configurations to the model such as different learning rates, various optimizers, and different amounts of separations of training and validation data. We have seen that the RMSprop optimizer outperformed the other two by achieving the highest 83.33% accuracy. In comparison with Adam, SGD, and RMSprop optimizers the RMSprop had very little overfitting as well as an underfitting state during the model training. Adam optimizer was the second-best performing optimizer after RMSprop both the optimizer achieved identical results in most of the experiments. The performance of the SGD optimizer in our study was inferior among the other two. SGD ends up in an underfitting state in almost every experiment performed along with achieving the lowest accuracy.

The proposed approach may also be used for implementing the classification system of other medical imaging domains such as CT scans, MRIs, etc. The performance achieved in this project shows the proposed strategy is comparable to the performance of a human radiologist. The study can be extended to predict other diseases such as lung cancer, pneumothorax, etc. using chest radiology images. This study can also be effective in predicting the latest COVID-19 infected chest x-ray images.

REFERENCES

[1]. Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway, and Jianming Liang. “Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?,” IEEE Transactions on medical imaging, Vol. 35, no.5, pp. 1299-1312, May 2016.

[2]. Ilyas Sirazitdinov, Maksym Kholiavchenko, Tamerlan Mustafaev, Yuan Yixuan, Ramil Kuleev, Bulat Ibragimov, “Deep neural network ensemble for pneumonia localization from a large-scale chest x-ray database,” Computers and Electrical Engineering. Vol.78, pp. 388-399. September 2019.

[3]. Neha Sharma, Vibhor Jain, Anju Mishra, “An Analysis of Convolutional Neural Networks for Image Classification,” International Conferences on Computational Intelligence and Data Science, Vol. 132, 2018, pp. 377-384.

[4]. Swati, Zar & Zhao, Qinghua & Kabir, Muhammad & Ali, Farman & Ali, Zakir & Ahmad, Saeed & Lu, Jianfeng. “Brain tumor classification for MR images using transfer learning and fine-tuning,” Computerized Medical Imaging and Graphics, Vol. 75, pp. 34-46, July 2019.

[5]. Devnath, Liton, Suhuai Luo, Peter Summons, and Dadong Wang. “Tuberculosis (TB) classification in chest radiographs using deep convolutional neural networks.” International Journal of Advances in Science Engineering and Technology. Vol. 6, no.3, pp. 68-74, August 2018.

[6]. Ravì, Daniele & Wong, Charence & Deligianni, Fani & Berthelot, Melissa & Andreu, Javier & Lo, Benny & Yang, Guang-Zhong. “Deep Learning for Health Informatics,” IEEE Journal of biomedical and health informatics, Vol. 21, no.1, pp. 4-21, January 2017.

[7]. Yamashita, R., Nishio, M., Do, R.K.G. et al. “Convolutional neural networks: an overview and application in radiology,” Insights into Imaging, Vol. 9, no. 4, pp. 611–629, June 2018.

[8]. Marleen de Bruijne. “Machine learning approaches in medical image analysis: from detection to diagnosis,” Medical Image Analysis, Vol. 33, pp. 94-97, October 2016.

[9]. Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng and M. Chen, "Medical image classification with convolutional neural network," 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), pp. 844-848, December 2014.

[10]. Jaiswal, Amit & Tiwari, Prayag & Agnihotri, Sachin & Gupta, Deepak & Khanna, Ashish & Rodrigues, Joel, “Identifying Pneumonia in Chest X-Rays: A Deep Learning Approach,” Measurement. Vol. 145, pp. 511-518, June 2019.

[11]. Prayogo, K.A., Suryadibrata, A. and Young, J.C. “Classification of pneumonia from X-ray images using siamese convolutional network,” Telkomnika, Vol. 18, no. 3, pp.1302-1309, June 2020.

[12]. Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., … Summers, R. M. “Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning,” IEEE Transactions on Medical Imaging, Vol. 35, no. 5, pp. 1285–1298, February 2016.

[13]. Giger, M. L. “Machine Learning in Medical Imaging,” Journal of the American College of Radiology, Vol. 15, no. 3, pp. 512–520, March 2018.

[14]. Alzubaidi, L., Fadhel, M. A., Al-Shamma, O., Zhang, J., Santamaría, J., Duan, Y., & R. Oleiwi, S. “Towards a Better Understanding of Transfer Learning for Medical Imaging: A Case Study,” Applied Sciences, Vol. 10, no. 13, pp. 4523. June 2020.

[15]. Hussain M., Bird J.J., Faria D.R. “A Study on CNN Transfer Learning for Image Classification,” Advances in Computational Intelligence Systems, Vol 840, pp. 191-202, August 2018.

[16]. World Health Organization, “Pneumonia”, Internet: https://www.who.int/news-room/fact-sheets/detail/pneumonia, August 2, 2019.

[17]. Chhavi Sachdev, “Why Does India Lead the World in Deaths From TB?” Internet: https://www.npr.org/sections/goatsandsoda/2017/11/09/561834263/why-does-india-lead-the-world-in-deaths-from-tb, November 9, 2017.

[18]. Judith Marcin, “What you should know about pneumonia”, Internet: https://www.medicalnewstoday.com/articles/151632, January 24, 2020.

[19]. Felix, “A Comprehensive guide to Fine-tuning Deep Learning Models in Keras (Part I)”. Internet: https://flyyufelix.github.io/2016/10/03/fine-tuning-in-keras-part1.html, October 3, 2016.

[20]. Raghu Prabhu, "Understanding of Convolutional Neural Network (CNN) — Deep Learning", Internet: https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148, March 4, 2018.

[21]. Pedro Marcelino, "Transfer learning from pre-trained models", Internet: https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751, October 23, 2018.

[22]. NIH, "Pneumonia", Internet: https://www.nhlbi.nih.gov/health-topics/pneumonia, April 26, 2018.

[23]. Kaggle, "Chest X-ray images (Pneumonia)", Internet: https://www.kaggle.com/paultimothymooney/chest-xray pneumonia, March 25, 2018.