

Analysis And Prediction of Covid-19 Disease Using CNN-Resnet Algorithm

Vaishali S P^[1], Dr.K.Logesh^[2], Dr.D.Sumathi^[3]

Department of Computer Science and Engineering, Kuppam Engineering College, KES Nagar, Kuppam, Andhra Pradesh 517425 India

Abstract

In this paper presents the deep learning diagnostic functions for chest x-rays and a COVID-Net-based image classifier for classifying chest x-ray images. In this article, we use model integration and transfer learning to classify chest x-rays in two ways: covid and non-covid. CNN can be used to make our result, which is more sensitive than radiologists in the detection and diagnosis of lung modules. According to the precision and loss value, choose the CNN model with good effect for the fusion and dynamically improve your weight ratio during the training process. This algorithm, using the RESNET50 model, is more sensitive than radiologists in screening and diagnosing lung modules.

Keywords: Disease Detection, Artificial Intelligence, Digital Image Processing, Transfer Learning, Model Integration

1. INTRODUCTION

The spread of the COVID-19 disease has unraveled life today. The most critical problem in health and medicine is the rapid detection of COVID19. Therefore, it is very important to review the diagnosis of the suspected case, in order not only to facilitate the next step for the patient, but also to reduce the number of infected people. X-ray examination is considered to be the most widely used X-ray examination method due to its low cost, wide range of applications, and high speed. It plays an essential role in the screening of COVID-19 patients and the detection of the disease. Since COVID-19 attacks human respiratory epithelial cells, we can use x-rays to see the health of the patient's lungs. The proof of these properties by x-ray has the highest priority. The Deep Convolutional Neural Network has achieved remarkable development in image recognition, particularly in the area of auxiliary medical diagnostics. Neural networks have been used with success in the x-ray identification of pneumonia, overtaking radiologists. One of the most important developments is the COVID-Net model of Darwin AI, Canada [2]. At the same time, the chest x-ray dataset that Covid had submitted for COVID-Net training and assessment was open source. Ezz El-DinHemdan, et al. COVIDX-Net proposed based on seven different architectures of DCNN [3], namely VGG19, DenseNet201 [4], InceptionV2, ResNet101, InceptionV3, Xception and MobileNetV2 [5]. Asmaa Abbas et al. proposed a Decay Transfer Composition (DeTraC) approach to classify COVID-19 chest x-rays [6]. This model achieves a sensitivity of 97.91%. The ResNet50-based Unet can also accurately identify a duplicate lung area on a chest x-ray and infer the area of pneumothorax [7]. This article aims to transfer learning and combine models. Our model is based on COVID-Net, an open source approach to identify COVID-19. First, load the data and solve the problem of unbalanced data. Then combine transfer learning with a modified deep network. Finally, depending on the precision and the loss value, go for the ResNet-101 and ResNet-152 models with good fusion properties and intensely improve your weight ratio during the training activity form a closed loop and iteratively update.

CNN has demonstrated its exceptional ability to automatically process large volumes of medical images and identify complex associations in large amounts of data for disease diagnosis and treatment planning.

The dataset used included CXR images from two groups. One group contains CXR images from COVID-19 patients and the other contains CXR images from normal patients. These groups were divided into two subgroups. One of them is a learning set and the other is a test set.

2. RELATED WORK

This section covers the work of scientists in many fields, such as Image Segmentation, DeTrac Deep Convolutional Network, and Deep Learning Framework for Screening of COVID19 Etc.

To use the available annotated examples more effectively, we can use the data-intensive network and training strategy. The increase in data is essential in order to teach the network the desired properties of invariance and robustness when only a few training samples are available [7].

In the CheXNet model there is a 121-layer convolutional neural network that was trained at ChestX-ray14, they used a chest x-ray dataset that contains over 100,000 frontal x-rays with 14 diseases. This algorithm can detect pneumonia in the chest X-ray better than practicing radiologists [1]. They presented a model that is able to automatically detect pneumonia on chest x-rays better than practicing radiologists. This technology will improve health care and improve access to medical imaging skills in regions of the world where access to qualified radiologists is limited [1] with experts in automation. Chest x-rays of patients with COVID-19 show some abnormalities on the x-ray [2]. The goal of her work is to create open source and open access datasets and to present an accurate framework for a convolutional neural network to differentiate cases of COVID-19 from other cases of pneumonia. One of the most encouraging results is the model's ability to achieve higher sensitivity and PPV in the normal class. This ensures that false positive cases occur not only with COVID-19, but also with the other two classes of infection and that the burden on the health system is relieved [2].

When diagnosing COVID-19 disease, the first imaging test that plays an important role plays a chest x-ray. Transfer learning is an efficient mechanism that can offer a promising solution by transferring knowledge from generic object recognition tasks to domain-specific tasks [6]. The Convolutional Neural Network (CNN) [6] is the most widely used algorithm that has demonstrated its ability to diagnose medical images with high accuracy. Various segmentation and classification methods are available. There are two different strategies used to train CNN. Transfer learning offers an effective solution with the limited availability of a large imaging dataset with benchmarking of the specific medical imaging task. There is a deep CNN architecture for DeTrac that uses a class breakdown approach to classify COVID-19 images into a comprehensive breast imaging dataset [6]. ImageNet is a template used to fill the majority of WordNet's 80,000 synsets with an average of 500-1,000 sharp images at full resolution [12], resulting in tens of millions of annotated images organized according to Job's semantic hierarchy are. ImageNet is much larger in scope and variety and much more precise than current image data sets. They illustrate

the benefits of ImageNet through three simple applications in object recognition, image classification and automatic object grouping [12].

The choice of method depends on the properties of the image [10]. X-rays have long been used to obtain images of the internal structure of the human body. It is one of the most widely used diagnostic tools in the medical field. X-rays are used to take pictures of the internal structure of the body, which is very helpful for radiologists to identify internal problems [10].

2. DESIGN METHODOLOGY

Imaging studies (mammography, ultrasound and MRI, x-rays of COVID-19) have been used to detect the disease for many years. But their use has certain limitations; more specifically, it is exposure to the effects of radiation on the human body. Hence, the machine learning method used to predict COVID-19 or a normal disease is as follows.

We collected Chest x-ray sheets images from online resource that contain 2 divisions for with covid and without covid images, then that will be passed to the CNN layers. Here we implemented CNN algorithm like RESNET. It is applicable for image process. Once Data Collected it will move to the algorithm section to process those data. In the end we study both algorithms in the result with confusion matrix.

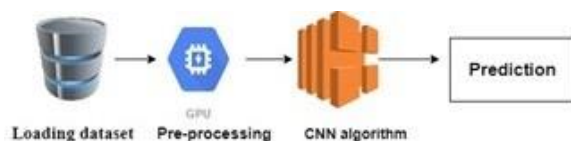


Fig 1: Overview of prediction algorithm

This is the image processing technique which environments covid on x-ray form of images. First we collect the dataset of chests of persons with 2 categories like healthy and covid. Then it will move to the algorithm, now we are used vgg16 ml model, it will process and train. Finally we will classify which categories. The Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image and be able to differentiate one from the other. When compared to other algorithms the initialization steps required in a ConvNet is very less. ConvNets need not be restricted to only one Convolutional Layer. At first ConvLayer should capture the Low-Level features such as edges, color, gradient orientation, etc.

3.1 Algorithm

ResNet-50 is a Deep Convolutional Neural Network. There is a ready-made version of the Trained Network in which we can download over a million images from the ImageNet database [1]. As a result, the network has assimilated rich feature representations for a wide range of images. The network has an input image size of 224 x 224. One of the great advantages of resnet is that it increases the depth of the network.

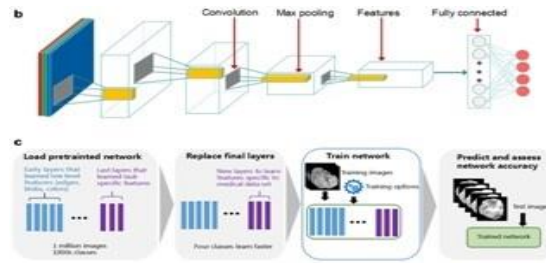


Fig 2: Resnet-50 Model

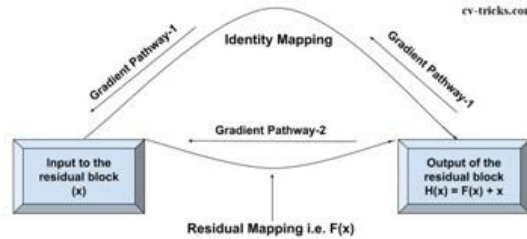


Fig 3: Gradient Pathways in Resnet

First we will see how the remainder block $F(x) y = F(x, \{W_i\}) + x$ is represented mathematically, where y is the output function, x is the input of the remainder block, and $F(x, \{W_i\})$ is the remainder block. In addition, the term $F(x, \{W_i\})$ for 2 weight layers in a remainder block can be simplified and written as follows: $F(x, \{W_i\}) = W_2 \sigma(W_1 x)$ where σ is the ReLU. is function activation and the second non-linearity is added after the addition with identity mapping, i.e. $H(x) = \sigma(y)$. When the calculated gradients pass Gradient Pathway2, two weight layers are found: W_1 and W_2 in our residual function $F(x)$.

3.2 Architecture

System architecture is a perspective of the whole system. In this architecture, the input is a CXR image and the output is a preview image. In this case, it predicts whether the image will be affected by COVID-19. The input form is 224×224 and there are three channels.

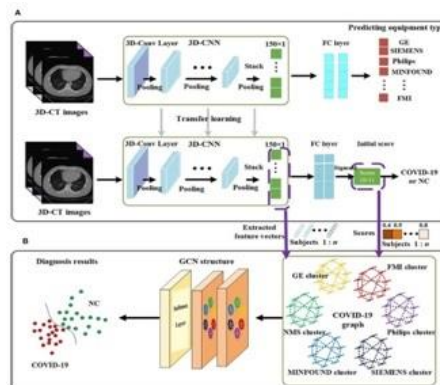


Fig 4: Architecture of resnet50 model in covid-19 prediction

The chest computed tomography (CT) datasets that are now available are diversified according to device types. There is a three-dimensional (3D) deep learning method that allows us to quickly diagnose Coronavirus Disease 2019 (COVID-19). First, we can apply a 3D Convolutional Neural Network (3D-CNN) to extract the image properties from the initial 3D CT images. In the next step, we can design a COVID-19 chart on CNN based on the characteristics extracted. The table divides all the samples into several groups and the samples with the same equipment form a group. To calculate the exact edge weights, we propose to combine the correlation distance of the extracted features and the subject score differences of the 3D CNN structure. Ultimately, we can enter the COVID-19 graph on CNN and get the final diagnostic results.

3.3 Training the Model

In the design methodology of our paper we will be seeing the steps covered in the training our resnet model. Below are the steps: Step 1) Run the Tensor Flow Dockers container. Step 2) Download and preprocess the ImageNet. Step 3) Download Tensor Flow models. Step 4) Export PYTHONPATH. Step 5) Install Dependencies (You're almost ready!) Step 6) Set training parameters, train the Resnet

3. IMPLEMENTATION

1. Mounting the drive

Before the computer can use any type of storage device (such as a hard drive, CD-ROM, or network share), you or your operating system must make it accessible through the computer's file system. This process is known as assembly. You can only access files on hosted media. In our project we put the encoding files on a drive and mount them in our project.

2. Import all dependencies

Python 3.9 support requires TensorFlow 2.5 or higher. We use an open source software library for numerical calculations, i.e. TensorFlow. This TensorFlow architecture enables simple calculation and provision on different platforms (CPU, GPU and TPU). Python Keras is a deep learning API written in Python that runs on TensorFlow. Scikit-learn (Sklearn) is Python's most useful and robust machine learning library. With this we can make a selection of effective machine learning and statistical modeling tools that include classification, regression and grouping in Python. In Python there is a cross-platform library function for plotting and data visualization called Matplotlib and its numeric extension NumPy. As such, it offers a viable open source alternative to MATLAB. Seaborn is also an open source Python library function based on matplotlib. Pandas Data Blocks and Library use Seaborn to work efficiently. The graphics created can also be easily adapted. NumPy, which stands for Numeric Python, is a library containing multidimensional array objects and a collection of routines for processing these arrays. Using NumPy, you can perform math and logic operations on arrays. NumPy is a Python package. The glob module is a useful part of the standard Python library. glob (short for global) is used to return all file paths that match a specific pattern.

3. Defining the parameters

The process in which the ENTIRE dataset is sent back and forth through the neural network only ONCE is called an epoch. The era is too long for a computer to work, so we have broken it down into several smaller parts. We use variables to define paths for covid images and non-covid images.

4. Fetch images and classes from files

We are appealing the covid and non-covid images from the path using imread operator. We are using cvtcolor operator to convert the images colors and append is used to add data to the images.

5. Visualize images from dataset

We will be plotting few images from the dataset and displaying it.

Here we use subplots function which creates a full grid of equal-sized axes in a single line, returning them in a NumPy array. We will be displaying images from covid and non-covid dataset respectively.

6. Normalization

Model takes images in the form of array of pixels. Hence convert into array and normalize it. There are preprocessing library in python, which contains the normalize function to normalize the data. We use an array as input and normalizes its values between 0 and 1. It then returns an output array with the same dimensions as the input. We can normalize our input images to the interval of [0, 1].

7. Train Test Split

Here we will be splitting our dataset images into two categories i.e. Training data and Testing data. We are using the train_test_split library function. train_test_split is a Sklearn model selection function for splitting datasets into two subsets: training the data and testing it. With this function, there is no need to manually split the dataset. By default, Sklearn train_test_split will randomly split the two subsets.

8. Visualize few images from training and testing data sets

We will be plotting few images from the training and testing dataset and displaying it. We use plot_images function to display the images from respective set.

9. Building and Visualizing model

This is the main part of our implementation. The implementation of our resnet algorithm takes place here. We are initializing our resnet50 function with required variables. Flatten() function is used to flatten a matrix to one dimension. To avoid over-adjusting the ignore level, randomly set the input units to 0 with a frequency at each step during your workout.

Code Snippet from our algorithm is shown below:

```
res = ResNet50(weights="imagenet", include_top=False, input_tensor=Input(shape=(224, 224, 3)))
outputs = res.output
outputs = Flatten(name="flatten")(outputs)
outputs = Dropout(0.5)(outputs)
outputs = Dense(2, activation="softmax")(outputs)
```

```
model = Model(inputs=res.input, outputs=outputs)
```

```
for layer in res.layers:
```

```
    layer.trainable = False
```

```
model.compile(
```

```
    loss='categorical_crossentropy',
```

```
    optimizer='adam',
```

```
    metrics=['accuracy']
```

```
)
```

```
train_aug = ImageDataGenerator(
```

```
    rotation_range=20,
```

```
    width_shift_range=0.2,
```

```
    height_shift_range=0.2,
```

```
    horizontal_flip=True
```

```
)
```

The compilation of training process begins with loss function – categorical_crossentropy. The output tag is assigned a unique category encoding value as 0 and 1 for optimization purposes, we use adam Optimizer. Adam is a stochastic gradient descent replacement optimization algorithm for training deep learning models.

We use ImageDataGenerator function for the training purpose. We use Keras ImageDataGenerator to increase the count of images in real-time while our model is still training. This will make your model robust and also save up on the overhead memory. This function will produce 10 images in each iteration of the training. An iteration is defined as steps per epoch i.e. the total number of samples / batch_size.

6. Training the model

We use model.fit () for training purposes. This corresponds to the training data. The extent to which the machine learning model generalizes to data similar to what it was trained on is known as model fitting. The model that gives the most accurate results is said to be the most accurate. An overfitted model fits the data too much. If we provide "s" stages per epoch, each "s" stage has "x" batches each consisting of "n" samples sent to fit_generator. So if you specify 5 levels per epoch, each epoch calculates "" x 'many each. Assembled 5 times from 'n' samples, then the next epoch is started.

7. Making Predictions

We use model.predict (). The Python Predict () function allows us to predict the labels of data values based on the trained model. Therefore, the Prediction () function works with the trained model and uses the linear label to map and predict the labels of the data under test.

8. Visualizing first 10 predictions

This step predicts all the training data and classifies into covid and non-covid. The prediction method uses probability method and checks whether it is covid report or not. The enumeration of the dataset takes place and if the probability is greater than 0.5 it comes under covid else it comes under non-covid.

9. Plot ROC curve

A receiver operating characteristic curve (ROC) is a graph that shows the performance of a classification model at all classification thresholds. Two parameters are recorded on this curve: the rate of true positives and the rate of false positives.

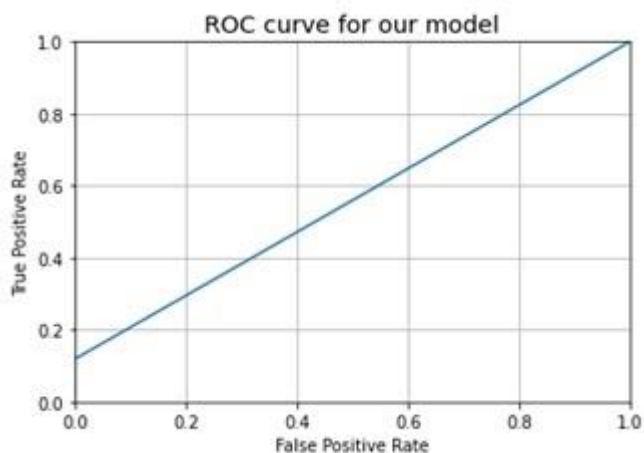


Fig 5: ROC curve for our model

10. Plot Confusion Matrix

The summary of the results of a prediction into a classification problem is known as a confusion matrix. The number of valid and invalid predictions is summarized with counts and broken down into classes. This is the key to the confusion matrix. We use Heatmap () which contains values that represent multiple shades of the same color for each value to be represented. The darker tones in the graph represent higher values than the lighter tones.

We are plotting two matrixes: Confusion Matrix with Normalization and Confusion Matrix without Normalization

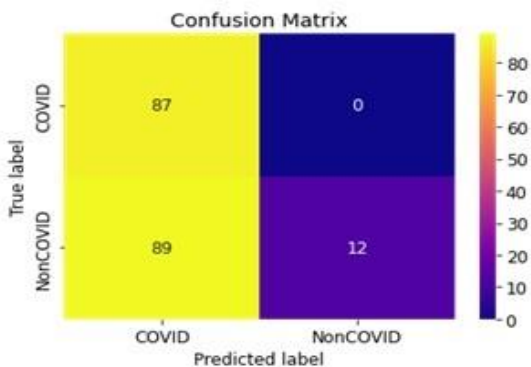


Fig 6: Confusion Matrix without Normalization

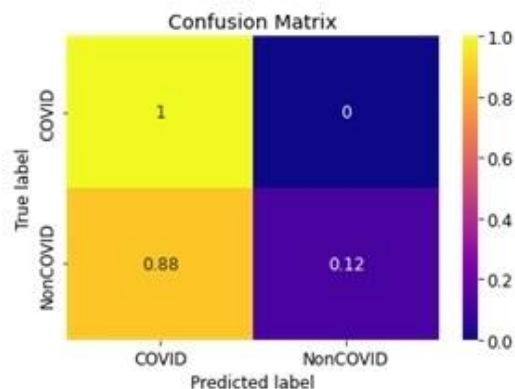


Fig 7: Confusion Matrix with Normalization

11. Classification Report

The report used to measure the quality of the predictions of a ranking algorithm is called a ranking report. We use the use of true and false positives, true and false negatives to calculate the metrics. The report shows the highest ranking, accuracy, recall and f1 scores for each class.

	Precision	Recall	F1-Score	Support
0	0.49	1.00	0.66	87
1	1.00	0.12	0.21	101
Accuracy			0.53	188
Macro Avg	0.75	0.56	0.44	188
Weighted Avg	0.77	0.53	0.42	188

Table 1: Classification Report

12. Accuracy and Loss plots

We use legend() to plot this graph. A legend is an areadescribing the elements of the graph. In the matplotlib library, there's a function called legend () which is used to Place a legend on the axis. In our work we plot the graph between training and testing data. For saving our data we use savefig(). After plotting the data we should save the figure to save it we use savefig(). We can also save the created figure to our local machines by using this method.

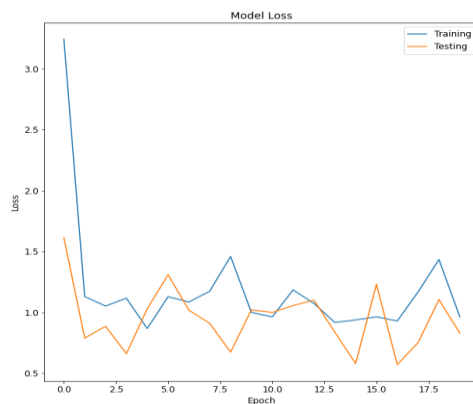


Fig 8: Accuracy and Loss Plot Graph

4. RESULT

We input both positive and negative chest-x-ray images of covid and non covid images respectively

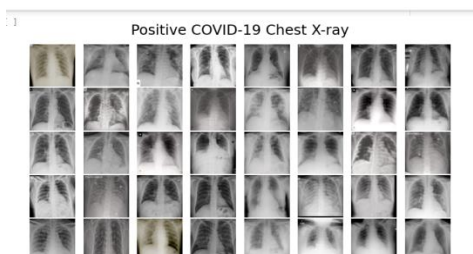


Fig 9: Positive covid-19 Images

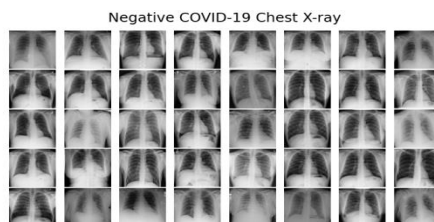


Fig 10: Negative Covid-19 Images

After the training process it does the prediction as shown in the below images.

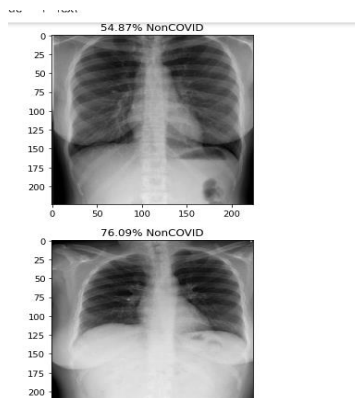


Fig 11: Percentage of covid-19 prediction

After the prediction we plot all the graphs as shown in the implementation.

5. CONCLUSION

In response to this COVID-19 outbreak, deep learning has played an indispensable role in accurately assessing and responding to the outbreak. We examined the diagnostic and analytical capabilities of deep learning in chest x-rays and introduced a COVID-Net-based image classifier to classify chest x-rays. Our algorithm is aimed at model integration and classifies chest x-rays by three days: normal, COVID-19 and viral pneumonia. According to the accuracy and the amount of loss, choose a CNN model with good melting effect, and dynamically improve the weight ratio during training. It provides a benchmarking method for healthcare facilities, government agencies, and even a general diagnosis of the COVID-19 epidemic situation. In this article, we use a deep CNN architecture based on a touchy class-based approach to classify COVID-19 images. In a large-scale data set of chest x-rays has shown effective solutions for classifying COVID-19 images or non-COVID-19 images (normal or abnormal). Future work could use this framework in different pharmaceutical regions. We can incorporate this algorithm into various IOT devices that can be useful in future medical problems.

7. REFERENCES

- [1] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M.P., Ng, A.Y., 2017. CheXNet: Radiologist-level pneumonia detection on chest x-rays with deep learning. arXiv:1711.05225.
- [2] Muhammad Farooq and Abdul Hafeez. Covid-resnet: A deep learning framework for screening of covid19 from radiographs. arXiv preprint arXiv:2003.14395, 2020.
- [3] Ezz El-Din Hemdan, Marwa A Shouman, and Mohamed Esmail Karar. Covidx-net: A framework of deep learning classifiers to diagnose covid19 in x-ray images. arXiv preprint arXiv:2003.11055, 2020.
- [4] Xiang Yu, Nianyin Zeng, Shuai Liu, and Yu-Dong Zhang. Utilization of densenet201 for diagnosis of breast abnormality. Machine Vision and Applications, 30(7-8):1135–1144, 2019.
- [5] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4510–4520, 2018.
- [6] Asmaa Abbas, Mohammed M Abdelsamea, and Mohamed Medhat Gaber. Classification of covid-19 in chest x-ray images using detrac deep convolutional neural network. arXiv preprint arXiv:2003.13815, 2020.
- [7] Thomas Brox Olaf Ronneberger, Philipp Fischer. U-net: Convolutional networks for biomedical image segmentation. arXiv:1505.04597 [cs.CV], 2015.
- [8] North America, R. S. RSNA pneumonia detection challenges. <https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/data> (2019).
- [9] Cohen, J.P., Morrison, P., Dao, L., 2020. Covid-19 image data collection. arXiv 2003.11597 URL: <https://github.com/ieee8023/covid-chestxraydataset>.

- [10] Navdeep Kanwal, Akshay Girdhar, and Savita Gupta. Region based adaptive contrast enhancement of medical x-ray images. In 2011 5th International Conference on Bioinformatics and Biomedical Engineering, pages 1–5. IEEE, 2011.
- [11] Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y., 2016. Deep learning. volume 1. MIT press Cambridge.
- [12] J. Deng, W. Dong, R. Socher, L. Li, Kai Li and Li Fei- Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, 2009, pp. 248255, doi: 10.1109/CVPR.2009.5206848.
- [13] Golbeck J, Robles C, Edmondson M, Turner K. Predicting personality from twitter. In: 2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing. IEEE; 2011. p. 149–156.
- [14] Wong KT, Antonio GE, Hui DS, Lee N, Yuen EH, Wu A, Leung CB, Rainer TH, Cameron P, Chung SS, Sung JJ (2003) Severe acute respiratory syndrome: radiographic appearances and pattern of progression in 138 patients. Radiology 228(2):401–406
- [15] Apostolopoulos ID, Mpesiana TA (2020) Covid-19: automatic detection from x-ray images utilising transfer learning with convolutional neural networks.