

# Detection of COVID-19 Using Deep Learning Technique

Aanchal Dusija

Meghnad Desai Academy of Economics

(Dated: July 18, 2020)

Abstract goes here ...

JEL classification codes: ...

Keywords ...

## I. INTRODUCTION

COVID-19 is an infectious disease caused by a recently discovered coronavirus. The outbreak commenced in December 2019 in the city of Wuhan, China which soon turned into a pandemic due to its contagious nature. It is also called as Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2).

Coronaviruses (CoV) are a family of hundreds of viruses that usually infect animals like chickens, bats, camels, cats. These viruses within the animals can mutate permitting them to transmit the virus within them to other species leading to a spill-over. It has been observed that the virus is mainly attacking the respiratory system in humans, ranging from the common cold to more deadly diseases like Middle East Respiratory Syndrome (MERS) and Severe Acute Respiratory Syndrome (SARS).

The symptoms of COVID-19 can be very mild, some people may not show any symptoms at all but can infect other people. It is not as deadly as SARS and MERS. The most common symptoms of COVID-19 are fever, dry cough, and fatigue. Other symptoms that are less common in nature are headaches, nasal congestion, sore throat, loss of taste and smell, conjunctivitis, and discoloration of fingers or toes.

Around 80% of people recover without being hospitalized. The rest become seriously ill and have difficulty in breathing. Usually, older people having other underlying medical problems like high blood pressure, heart and lung problems, diabetes, or cancer are at a higher risk of developing a serious illness. However, anyone can get the virus and get seriously ill.

SARS, which was first detected in November 2002, bears several similarities to COVID-19. Both the coronaviruses are believed to have originated from bats, jumping to humans via an intermediate animal host. According to the WHO, SARS punched holes in the lungs, giving them “a honeycomb-like appearance”—and these lesions are present in those afflicted by a novel coronavirus, too. However, the mortality rates for SARS are higher. In 2012, there was another outbreak of a newly found coronavirus called the Middle East Respiratory Syndrome (MERS). The very first case was found in Saudi Arabia. The difference between SARS, MERS, and COVID-19 is that  $R_0$  for SARS is 3,  $R_0$  for MERS is less than 1 but the  $R_0$  for COVID-19 is 5, meaning that every infected person is likely to infect 5 other people. This shows how the infectious nature of COVID-19.

COVID-19 begins and ends in patients' lungs, because like the flu, coronaviruses are respiratory diseases. They spread typically when an infected person coughs or sneezes, spraying droplets that can transmit the virus to anyone in close contact. Coronaviruses also cause flu-like symptoms: Patients might start with a fever and cough that progresses to pneumonia or worse.

COVID-19 is detected in the upper and lower respiratory specimens of the individuals with the help of RT-PCR (real-time transcription-polymerase chain reaction) test. Since the origin of the outbreak, the availability and quantity of the testing kits have been low. The stability and reproducibility of the detection kits are being questioned. These factors play a determinant role in the accuracy of test results. In several areas, the accuracy of the kits has found to be only 50% and has to hence be repeated several times before the cases can be confirmed.

X-ray images of the chest can be used to diagnose COVID-19 with technological advancements made in the field of machine learning. One of the most important methods of machine learning is deep learning. Deep learning focuses on extracting features and classifies images which are applied in detecting objects or in medical cases, classification of tasks. Machine learning and deep learning have become established disciplines in applying artificial intelligence to mine, analyze, and recognize patterns from data. [1]

With the recent innovation in Artificial Intelligence (AI), COVID-19 can be detected, quantified, and monitored and making it easier to isolate patients who have been infected for faster treatment. Chest x-rays scan the infections present in the lungs of the patient. It is a faster, easier, cheaper, and less harmful method of testing patients and hence must be used. With the increasing mortality rates all thanks to COVID-19, Chest X-rays must be implemented by all countries on an urgent basis. Although this technological advancement seems helpful, the images of various types of pneumonia are similar and overlap with other infectious and inflammatory lung diseases making it difficult to distinguish between COVID-19 from other viral cases of pneumonia.[3]

In the study, a prediction of the COVID-19 detector will be modeled using Convolutional Neural Network (CNN) based on pre-trained models and Chest x-rays images. CNN helps in the extraction of the features by enhancing low-light images with the help of training data. In the early stages of COVID-19, bilateral distribution of patchy shadows and ground-glass opacity has been observed which are similar to the viral pneumonia symptoms with slight

differences. With the aid of the CNN model, unique features can be identified which are difficult for visual recognition.[4]

Despite the limitations of the current study, which are related to the data availability limitations, opens the horizons for more specialized research into the possible existence biomarkers related to the Covid-19 disease in X-ray images. The possible significance of those biomarkers can be confirmed or denied by other feature extraction techniques such as Radiomics in future research.[2]

## II. CNN MATH

Convolutional Neural Network (CNN) is a robust deep learning algorithm, mainly used to identify and classify images. CNN is a distinctive type of neural network in which analyses visual inputs to identify for segmentation, detection, and classification. The concept is based on combining low-level features in the image to higher and higher-level features. The filters extract features from images in such a way that the position information of pixels is retained. The algorithm is mainly used for face recognition, analysing documents, managing traffic in smart cities, and recommendation systems.

CNN is trained on the back-propagation algorithm. After we get the output from Forward propagation, the actual value is compared with the output to calculate the error rate. The parameters are then updated, and the entire process is repeated to get the optimal values. [1]

CNN is an amalgamation of biology, art, and mathematics. The biology of the eye inspires the entire architecture. CNN mimics the connectivity of neurons within the brain. While we as humans perceive a visual image as a detailed, coloured image of the world around us, there is quite a lot of processing done in our brain to get to this point.

CNN has proved particularly successful in working with image data and ever since being used in ImageNet competition in 2012. They have been the frontrunners in research and industry while dealing with images.[2]

CNN can also be used for deep learning applications in healthcare, such as medical imaging. CNN has been used for features learning on breast ultrasound images, blood analysis, brain lesion segmentation, detection of Alzheimer's and Parkinson's Diseases, tumour, lung cancer, pneumonia, and various other diseases. [3]

The neurons within a CNN are split into a three-dimensional structure, with each set of neurons analysing a small region or feature of the image. In other words, each group of neurons specializes in identifying one part of the image. CNNs use the predictions from the layers to produce a final output that presents a vector of probability scores to represent the likelihood that a specific feature belongs to a certain class. [4]

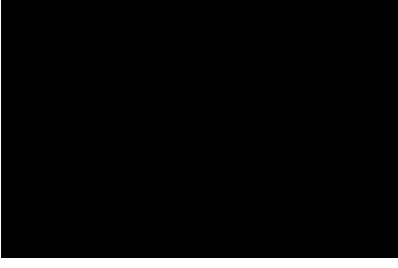
CNN is composed of several kinds of layers:

### 1) Convolutional Layer:

This layer consist of Image Filters that extract patterns from the image. What information

these filters extract is learned, just like in the brain. When we train a Convolutional Neural Layer, we try to generate the best possible filters, e.g. the filters that extract the most meaningful information.

Convolutional layers detect low-level features such as edges and curves. This layer creates a feature map to predict the class probabilities for each feature by applying a filter that scans the whole image, a few pixels at a time.



Let's consider a single image case. For simplicity, I would like to define the image size and its convoluted image size as above.

So notation here is like below.

$x$ : input

$a^k$ : after convoluted image

$k$ : index of kernel (weight filter)

$W$  : kernel (weight filter)

$b$ : bias

$E$ : cost function

CNN follows the forward propagation procedure in which the weights, biases and filters are inputted, processed and passed on to its successive layers. These values act as parameters of the CNN model.

$$a_{ij}^{(k)} = \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} W_{st}^{(k)} x_{(i+s)(j+t)} + b^{(k)} \quad (1)$$

The activation function is the non-linear transformation that we do over the input signal. This transformed output is then advanced to the next layer of neurons as input. When it comes to the selection of activation functions, indeed we have some options, for example, ReLU, sigmoid or hyperbolic tangent. They are given as follows:

Sigmoid Function:  $\frac{1}{1+e^{-x}}$

tanh Function:  $\tanh(x)$

ReLU Function:  $\max(0, x)$

Leaky ReLU Function:  $\max(0.1x, x)$

Maxout Function:  $\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU Function:  $\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$

ReLU function is the most widely used activation function in neural networks. ReLU converts all negative inputs to zero, and the neuron does not get activated, making it very computational efficient as few neurons are activated per time. It does not saturate at the positive region. The ReLU activation functions is used for the hidden layers.

$$a_{ij} = \max(0, x_{ij}) \quad (2)$$

## 2) Pooling Layer(Down-Sampling):

Even though the size of the images will slightly decrease with each filter in each layer, we generally generate exponentially more and more data with each convolutional layer we add. To combat this issue, we use a process called Pooling. In this layer, the amount of information generated by the convolutional layer is scaled-down and only saves crucial information. Most prominently, we use Max Pooling, meaning we take the maximum pixel value of a pixel neighbourhood.

$$a_{ij} = \max(0, x_{(i+s)(j+t)}) \quad (3)$$

where  $s \in [0, l]$  and  $t \in [0, l]$  and  $l$  is filter size.

## 3) Fully Connected Layer:

This layer “flattens” the outputs generated by previous layers to turn them into a single vector that can be used as an input for the next layer. It converts a two-dimensional convolutional layer into one dimension which can be used to connect to the fully connected layer. The weights are applied over the input generated by the feature analysis to predict an accurate label.

## 4) Fully Connected Output Layer:

A fully connected layer uses the output of the convolution layer to predict the best description for the image. This layer generates the final probabilities to determine a class for the image. Sigmoid function is a S-Shaped curve which is differentiable and monotonic in nature, existing between 0 and 1. The sigmoid activation function is used for the output layer

because there is a distinction to be made between two classes.

$$a_{ij} = \frac{1}{1 + e^{-x}} \quad (4)$$

A technique called forward-backward propagation is used to check and optimize the performance of the model. In forward propagation, the input data is fed in the network, and the output data is generated. The loss is calculated to see how far away the prediction of the network is from the actual value. We calculate the cross-entropy error of the model:

In the backward propagation process, the parameters of the model are updated in a manner that the overall predictions are more accurate than the previous model. This composition has to be optimized on a per-layer basis. The error can be propagated back from the end to the start by de-stacking through the function calls. This technique is called auto-differentiation and requires only that each function is provided with the implementation of its derivative.

Back propagation of the Convolutional layer:

i) Back propagation to Update the Weights:

$$\begin{aligned} \frac{\partial E}{\partial W_{at}^{(k)}} &= \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\partial E}{\partial a_{ij}^{(k)}} \frac{\partial a_{ij}^{(k)}}{\partial W_{nt}^{(k)}} = \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\partial E}{\partial a_{ij}^{(k)}} x_{(i+a)(t+t)} \\ \frac{\partial E}{\partial b^{(k)}} &= \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\partial E}{\partial a_{ij}^{(k)}} \frac{\partial a_{ij}^{(k)}}{\partial b^{(k)}} = \sum_{i=0}^{M-m} \sum_{j=0}^{N-n} \frac{\partial E}{\partial a_{ij}^{(k)}} \end{aligned} \quad (5)$$

Bear in mind that the propagated error can be noted like below.

$$\delta_{ij}^k = \frac{\partial E}{\partial a_{ij}^{(k)}}$$

ii) Back propagation to Previous Layer:

$$\frac{\partial E}{\partial x_{ij}} = \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} \frac{\partial E}{\partial a_{(i-s)(j-t)}^{(k)}} \frac{\partial a_{(i-s)(j-t)}^{(k)}}{\partial x_{ij}} = \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} \frac{\partial E}{\partial a_{(i-x)(j-t)}^{(k)}} W_{st}^{(k)} \quad (6)$$

iii) Back propagation to ReLu Layer:

$$\frac{\partial E}{\partial x_v} = \frac{\partial E}{\partial a_{ij}^{(k)}} \frac{\partial a_{ij}^{(k)}}{\partial x_{ij}} = \begin{cases} \frac{ax}{a < \infty} & (a_y^{(a)} \geq 0) \\ 0 & (\text{athervise}) \end{cases} \quad (7)$$

iv) Back propagation to Pooling Layer:

$$\frac{\partial E}{\partial x_{(i+s)(j+t)}} = \frac{\partial E}{\partial a_{ij}^{(k)}} \frac{\partial a_{ij}^{(k)}}{\partial x_{(i+s)(j+t)}} = \begin{cases} \frac{\partial E}{\partial a_{ij}^{(k)}} & (a_{ij}^{(k)} = x_{(i+s)(j+t)}) \\ 0 & (\text{otherwise}) \end{cases} \quad (8)$$



v) Back Propagation to Sigmoid Layer:

### III. DATA

EFGH

#### IV. METHODOLOGY

IJKL

## V. ANALYSIS

MNOP

## VI. RESULTS

QRST

## VII. CONCLUSION

UVWX

## VIII. REFERENCES

YZ