DETECTION OF COVID-19

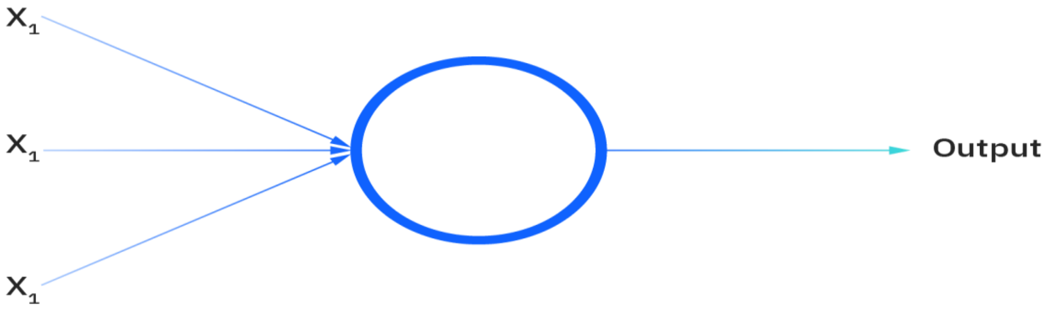
COMPUTER NETWORK- DR. M PRABU

MARCH 26, 2022

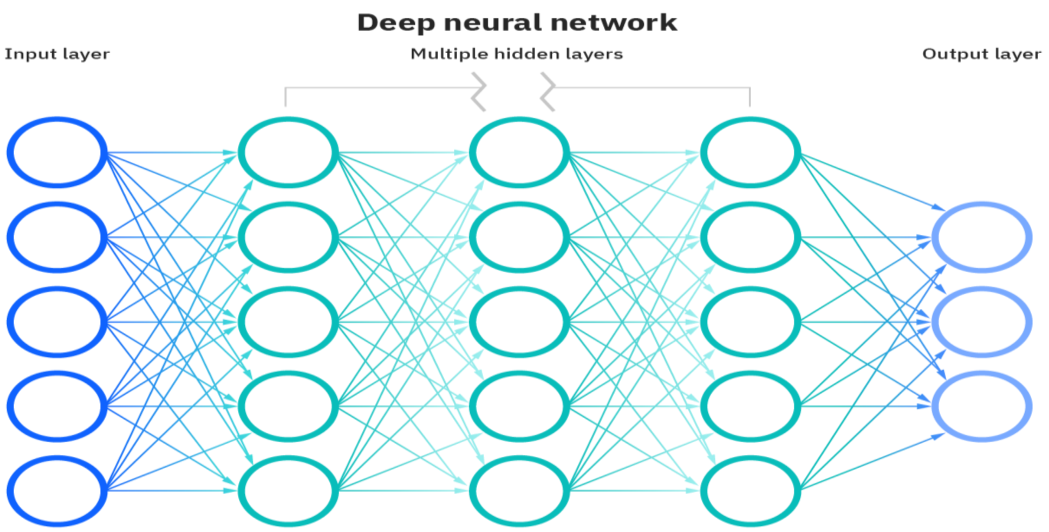
Group – 3
Sheenam Waris M200717CA
Janvi Agrawal M200695CA
Amanpreet Kaur M200680CA
Ankit Hammad M200683CA
Chetan Patidar M200688CA
Jayant Parganiha M200696CA

Introduction

Neural networks can be classified into different types, which are used for different purposes. The perceptron is the oldest neural network, created by Frank Rosenblatt in 1958. It has a single neuron and is the simplest form of a neural network:



Convolutional neural networks (CNNs) are similar to feedforward networks, but they're usually utilized for image recognition, pattern recognition, and/or computer vision. These networks harness principles from linear algebra, particularly matrix multiplication, to identify patterns within an image.



Details about the dataset: COVID-19 classification datasets

A total of positive and negative Chest X-ray images datasets was used in different stages. So, the dataset consists of COVID-19 X-ray scan images. It turns out that the most frequently used view is the Posteroanterior view and I have considered the COVID-19 PA view X-ray scans for my analysis.

To stratify our data we will take an equal number of images and will blend them and later will divide into test and train data.

How to download dataset & From where we have to download the dataset:

For positive cases, the COVID-19 Image Data Collection by https://github.com/ieee8023/covid-chestxray-dataset/tree/master/images , and negative cases of Normal chest X-ray by https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia. These databases can be downloaded from GitHub and Kaggle using above link.

How we interact with the data

First we set up google collab and create dataset. Import pandas then upload data images for further samples and mount drive to collab file. To develop this project, we used Python 3.10. All models were designed with using the Keras library. We used Google Collaboratory for most of the experiments. In this case, Tensor Processor Unit (TPU) was used when possible; otherwise, we used the Graphic Processor Unit (GPU) depending on the Collaboratory assignation.

What is the problem that we are handling?

The increasing frequency and magnitude of viral outbreaks in recent decades, by the current COVID-19 pandemic, has resulted in an urgent need for rapid and sensitive viral diagnostic methods. The symptoms of the disease can vary and include dyspnoea, high fever, runny nose, and cough according to WHO. Instead of waiting for blood testing or RT-pcr which takes approx. 5-6 hours to generate result. These cases can most commonly be diagnosed using chest X-ray imaging analysis

What is your approach to solve the problem?

Deep learning has shown a dramatic increase in the medical applications in general and specifically in medical image-based diagnosis. Due to the very promising results provided by CNNs in medical image analysis and classification, they are inspired by the visual system of the human brain. The idea behind the CNNs is to make computers capable of viewing the world as humans view it. This way CNNs can be used in the fields of image recognition and analysis, image classification, and natural language processing.

How do Convolutional neural networks work?

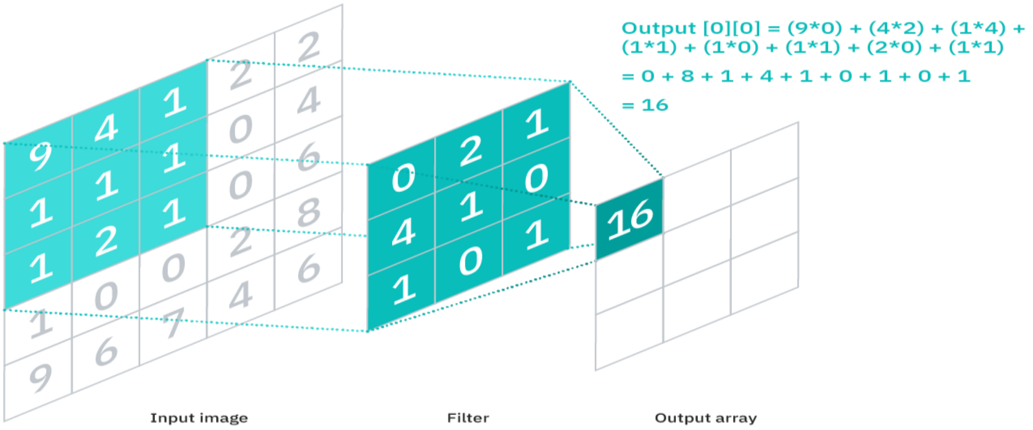
Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers, which are:

- Convolutional layer
- Pooling layer

- Fully-connected (FC) layer

Convolutional layer

The convolutional layer is the core building block of a CNN, and it is where the majority of computation occurs. It requires a few components, which are input data, a filter, and a feature map. Let's assume that the input will be a colour image, which is made up of a matrix of pixels in 3D. This means that the input will have three dimensions—a height, width, and depth—which correspond to RGB in an image. We also have a feature detector, also known as a kernel or a filter, which will move across the receptive fields of the image, checking if the feature is present. This process is known as a convolution.



as you can see in the image above, each output value in the feature map does not have to connect to each pixel value in the input image. It only needs to connect to the receptive field, where the filter is being applied. Since the output array does not need to map directly to each input value, convolutional (and pooling) layers are commonly referred to as "partially connected" layers. However, this characteristic can also be described as local connectivity.

1. The **number of filters** affects the depth of the output. For example, three distinct filters would yield three different feature maps, creating a depth of three.

2. **Stride** is the distance, or number of pixels, that the kernel moves over the input matrix. While stride values of two or greater is rare, a larger stride yields a smaller output.

3. **Zero-padding** is usually used when the filters do not fit the input image. This sets all elements that fall outside of the input matrix to zero, producing a larger or equally sized output. There are three types of padding:

- **Valid padding:** This is also known as no padding. In this case, the last convolution is dropped if dimensions do not align.
- **Same padding:** This padding ensures that the output layer has the same size as the input layer
- **Full padding:** This type of padding increases the size of the output by adding zeros to the border of the input.

After each convolution operation, a CNN applies a Rectified Linear Unit (ReLU) transformation to the feature map, introducing nonlinearity to the model.

Pooling Layer:

Pooling layers, also known as down sampling, conducts dimensionality reduction, reducing the number of parameters in the input.

There are two main types of pooling:

- **Max pooling:** As the filter moves across the input, it selects the pixel with the maximum value to send to the output array. As an aside, this approach tends to be used more often compared to average pooling.
- **Average pooling:** As the filter moves across the input, it calculates the average value within the receptive field to send to the output array.

While a lot of information is lost in the pooling layer, it also has a number of benefits to the CNN. They help to reduce complexity, improve efficiency, and limit risk of overfitting.

Fully connected Layer:

The name of the full-connected layer aptly describes itself. As mentioned earlier, the pixel values of the input image are not directly connected to the output layer in partially connected layers. However, in the fully-connected layer, each node in the output layer connects directly to a node in the previous layer.

This layer performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers tend to use ReLU functions, FC layers usually leverage a SoftMax activation function to classify inputs appropriately, producing a probability from 0 to 1.

What are the other applications of algorithm?

Decoding Facial Recognition

Analysing Documents

Understanding Climate

CNNs are poised to be the future with their introduction into driverless cars, robots that can mimic human behaviour, aides to human genome mapping projects, predicting earthquakes and natural disasters, and maybe even self-diagnoses of medical problem.

THANK YOU😊