COVID-19 detection in chest X-ray images using convolutional neural networks

Introduction

- Humans cannot determine a COVID-19 positive patient without a label on them.
- Computer vision and deep learning prove to beneficial by solving this problem by adopting an end-to-end learning architecture, utilizing chest x-rays.

Introduction

- Using CNN to classify images might avoid challenging and expensive feature engineering because it is a great feature extractor.
- When we apply deep learning models like CNN, automated COVID-19 detection can be incredibly efficient, affordable, and scalable.

CNN

- CNN is a type of artificial neural network that recognizes patterns in image data.

- CNN consists of three layers:
  - Convolutional layer
  - Pooling layer
  - Fully connected layer.

CNN

- The convolutional layer is the first layer while the FC layer is the last.
- From the convolutional layer to the FC layer, the complexity of the CNN increases. It is this increasing complexity that allows the CNN to successively identify larger portions and more complex features of an image until it finally identifies the object in its entirety.

Convolution layer

- The process of convolution involves a kernel or filter inside this layer moving across the receptive fields of the image, checking if a feature is present in the image.
- A “Kernel” refers to a 2D array of weights. The term “filter” is for 3D structures of multiple kernels stacked together. For a 2D filter, filter is same as kernel. But for a 3D filter and most convolutions in deep learning, a filter is a stack of kernels.

Convolution layer contd.

- Over multiple iterations, the kernel sweeps over the entire image. After each iteration a dot product is calculated between the input pixels and the filter.
- The final output from the series of dots is known as a feature map or convolved feature.
- Ultimately, the image is converted into numerical values in this layer, which allows the CNN to interpret the image and extract relevant patterns from it.

Pooling layer

- Like the convolutional layer, the pooling layer also sweeps a kernel or filter across the input image.
- But unlike the convolutional layer, the pooling layer reduces the number of parameters in the input and helping the network recognize features independent of their location in the image.
- On the positive side, this layer reduces complexity and improves the efficiency of the CNN.

Fully connected layer

- The FC layer is where image classification happens in the CNN based on the features extracted in the previous layers. Here, fully connected means that all the inputs or nodes from one layer are connected to every node of the next layer.

- All the layers in the CNN are not fully connected because it would result in an unnecessarily dense network. It would also increase losses and affect the output quality, and it would be computationally expensive.

Dataset

- Public open dataset of chest X-ray and CT images of patients which are positive or suspected of COVID-19 or other viral and bacterial pneumonias.
- Total of 930 images of all diseases scans. Only 196 images of covid-19 patients with frontal (PA) view chest x-ray scans.
- Open database available on GitHub. Last updated on Mar 2021.

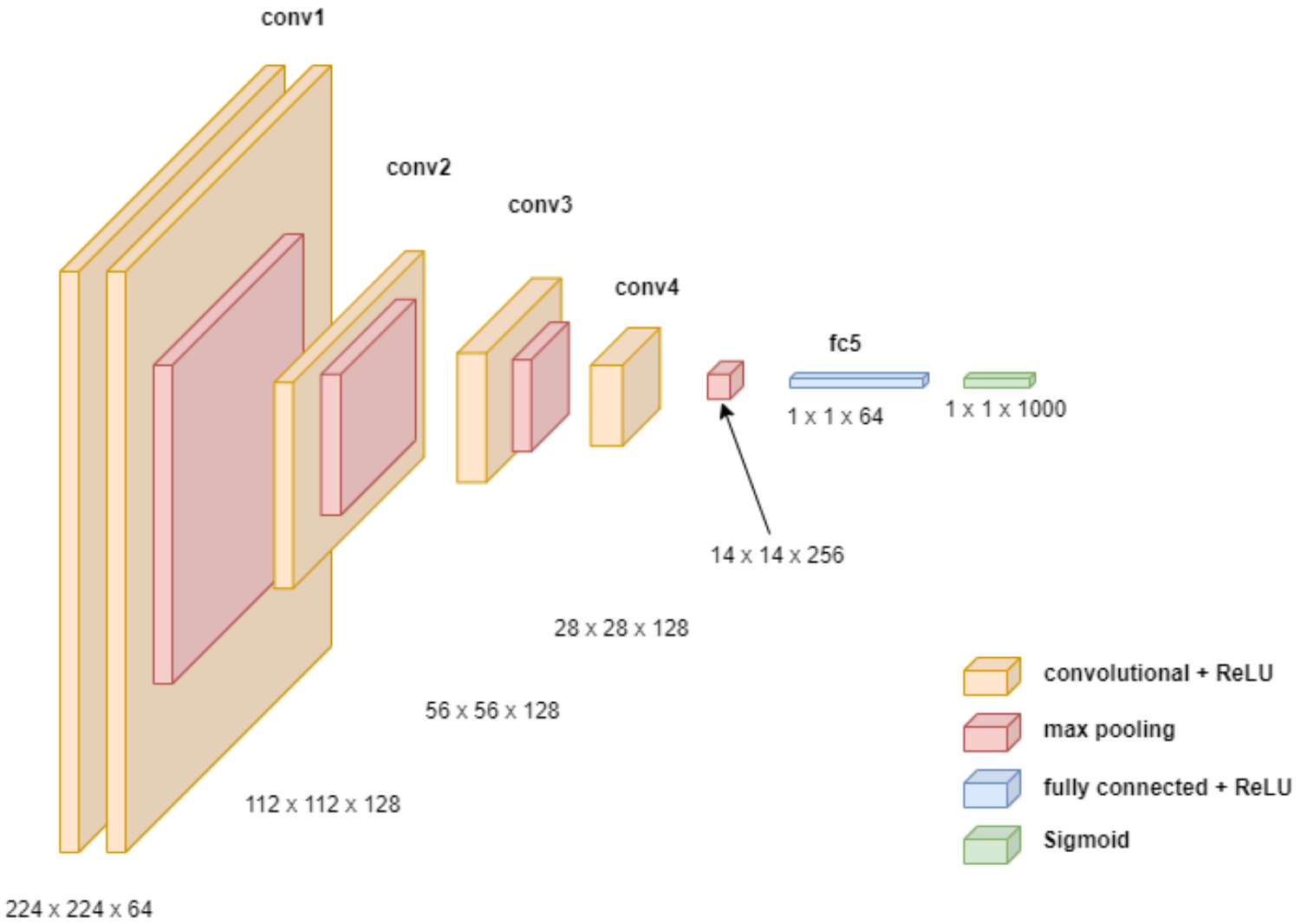
- There are 5,863 X-Ray images (JPEG) and 2 categories (Pneumonia/Normal). Only 1341 images of Normal category.
- Available on Kaggle, since 2017.

Normal vs COVID-19

Preprocessing

- Converting 224x224 RGB image to BGR image, because OpenCV library uses BGR image format.
- Scaling the pixel values are between 0 and 255 we divide each pixel value by 255, so that we scale them down to 0 and 1.
- Splitting of dataset into train, validation, test in the ratio of 0.8 : 0.1 : 0.1 respectively.
- Data augmentation: Horizontal flip, Slight zoom, Slight shear.

Architecture



Results

Model	Kernel size	Pool size	Padding	Training accuracy	Validation accuracy
Model1	2	2	same	0.984	0.922
Model2	2	3	same	0.961	0.943
Model3	2	3	valid	0.977	0.96
Model4	3	2	same	0.974	0.969
Model5	3	2	valid	0.941	0.906
Model5	3	3	same	0.984	0.93
Model7	5	2	valid	0.89	0.87
Model8	6	2	same	0.948	0.90
Model9	6	2	valid	0.863	0.844
Model10	7	2	same	0.95	0.92

Thank you