25

Convolutional Neural Network

→ A CNN is a type of artificial neural network used in image recognition and processing. #

→ It is not fully Connected.

→ It works well with images.

→ In previous, we were comparing pixel to pixel. But

→ In CNN, we compare the patch of 1 Sample to another Sample.

26

→ We Can also apply CNN with text Classification.

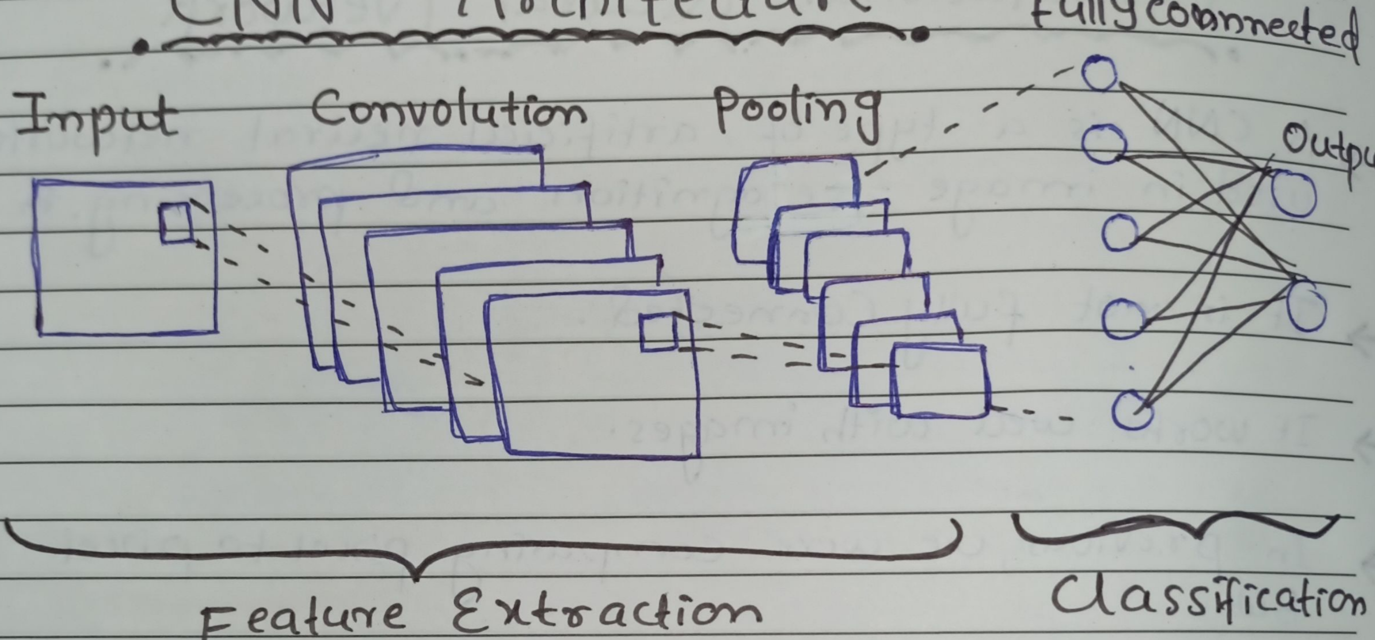
→ CNN is data hungry. It requires lots of data(images) to perform well.

→ CNN Learns the features based on images.

- Difference in CNN and ANN?

  In CNN, the only last layer of Network is fully connected wheras in ANN, each neuron is connected to every other neuron.

CNN Architecture



① Input [image]

② Convolutional Layer } You can use convo-pooling layer more than once. } stack

③ pooling layer

④ fully-Connected

⑤ Output layer

* Convolutional Layer: Operation:

It is a mathematical operation that allows merging of two sets of information. In CNN, it is applied to the input data to filter the information and produce a map.

$$\underbrace{\begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}_{5\times5}}_{\text{Input}} * \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}_{3\times3}$$

Kernel/Filter size = 3

31

stride by 1

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}_{5\times5} * \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}_{3\times3} = \begin{bmatrix} 2 & 3 & 5 \\ & & \\ & & \end{bmatrix}_{3\times3}$$

$2 = 0\times1 + 1\times0 + 1\times1 + 0\times1 + 1\times1 + 0\times1 + 1\times0 + 1\times0 + 0\times1$

$3 = 1\times1 + 1\times0 + 0\times1 + 1\times1 + 1\times1 + 0\times1 + 1\times0 + 1\times0 + 0\times1$

$5 = 1\times1 + 0\times0 + 1\times1 + 1\times1 + 0\times1 + 1\times1 + 1\times0 + 0\times0 + 1\times1$

So on....

01

- Filter/Kernel: This filter is also called a kernel, or feature detector and its dimension can be for example 3x3...

- padding: padding expands the input matrix by adding fake pixels to the borders of the matrix.

  This is done because Convolution reduces the size of the matrix.

  For example: a 5x5 matrix turns in to 3x3 matrix when a filter goes over it.

02

- Stride: The idea behind Stride is to skip Some areas when the kernel slides over:

  For example, Skipping every 2 or 3 pixels.

  It reduces Spatial resolution and make Computation efficient.

* Convolution with Stride:

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}_{2\times2} = \begin{bmatrix} 2 & 1 \\ & \end{bmatrix}_{2\times2}$$

Stride by 2

$0\times0 + 1\times1 + 0\times1 + 1\times0 = 0 + 1 + 0 + 1 = 2$

* Convolution with padding:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{6\times6} * \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}_{2\times2} = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}_{3\times3}$$

Adding fake pixels by adding row and column.

- pooling layer: A pooling layer receives the result from a Convolutional layer and compresses it. The filter of pooling layer is smaller than a feature map.

Usually it takes a 2x2 square (patch) and compresses it into one value.

* Pooling Operation:

- → max pooling
- → min pooling
- → averge pooling

} 3-types of pooling

$$\begin{bmatrix} 1 & 3 & 4 & 2 \\ 6 & 8 & 7 & 2 \\ 1 & 6 & 2 & 3 \\ 0 & 2 & 4 & 1 \end{bmatrix}_{4\times4}$$

- Stride by 1
- maxpooling

Consider the maximum value for eatch patch.

$$= \begin{bmatrix} 8 & 8 & 7 \\ 8 & 8 & 7 \\ 6 & 4 & 4 \end{bmatrix}_{3\times3}$$

The process of pooling with stride will further downside the image.

* fully- connected

* output layer.