Convolution Neural Network (CNN)

Artificial Intelligence has been performing wonders in filling the gap between human and the computer. One of many such areas is the domain of Computer Vision. The agenda for this field is to enable machines to view the world as humans do, perceive it in a similar manner and even use the knowledge for a multitude of tasks such as Image & Video recognition, Image Analysis & Classification, Media Recreation, Recommendation Systems, Natural Language Processing, etc. The advancements in Computer Vision with Deep Learning has been constructed and perfected with time, primarily over one particular algorithm i.e. a **Convolutional Neural Network**.

**What is CNN:**

Convolution Neural Network is an algorithms that let the computer to learn and perceive visuals as a human. CNN takes input as images, assign weights and biases to different objects in the image and then classify it.

There are four steps in implementing a CNN.

1. CONVOLUTION: In convolution, we first convert the input image into pixels and then convolve feature detector on the image that results in feature map. More feature detectors results in more feature maps.
2. Max Polling: In pooling, we select 4 pixels and then take max of them in the new matrix. Hence we reduced the size of feature vector.
3. Flattening: Flattening a feature means making feature into 1D column vector.
4. Full Connection.

**Convolution:**

The image is nothing but a matrix of pixel values. In convolution, convolve the whole image with the feature vector also known as filter. A filter may be of any size like 3x3, 5x5 etc. The filter is smaller than the input data and the type of multiplication applied between a filter-sized patch of the input and the filter is a dot product. A dot product is the element-wise multiplication between the filter-sized patch of the input and filter, which is then summed, always resulting in a single value. Because it results in a single value, the operation is often referred to as the “scalar product“. As we apply scalar product to each step, we get a single value which is stored in a separate array called feature map. The result is a two-dimensional array of output values that represent a filtering of the input.

**ReLU:**

Once a feature map is created, we can pass each value in the feature map through a nonlinearity, such as a ReLU, much like we do for the outputs of a fully connected layer.

**Polling:**

Pooling layers reduce the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. In max polling, we put the max polling, it uses the maximum value from each of a cluster of neurons at the prior layer. A*verage pooling* uses the average value from each of a cluster of neurons at the prior layer.

**Flattening:** In flattening, we simply convert a 2D array into a column matrix by putting successive rows down by down.

**Full Connection:**

It composed of neural network having one input/output layer with one middle layer. In ANN, we called middle layers as hidden layers. Here we named them as fully connected layers. In ANN, the layers don’t have to be fully connected but in CNN, layer must have to be fully connected. After flattening, the feature vector is then passed as input to the input layer. In CNN, it combines all attributes and predict the class of higher probability.