Case Study V

Krushang Shah

Prof. Richard Khan

Senior Design I

ID 23821856

Convolutional Neural Network

In the field of deep learning, the Convolutional Neural Networks (CNN/ConvNet) are a type of Neural Network, made of Neurons. They are the regularized versions of the multilayer perceptrons, which mean that they are fully connected networks. In each layer, each neuron is connected to all the neurons of the next layer. The Convolutional Neural Networks are inspired from the similarly happening biological process inside the human body. Hence the connectivity pattern of the neurons of the Convolutional Neural Network are connected with each other as the neurons in the brains of humans and animals are connected with each other. The name Convolutional Neural Network itself indicates that the main process behind the network is the mathematical process of convolution.

A general Convolutional Neural Network is made up of input layer, several hidden layers, and an output layer. The input layer is the data that is inputted in the model, to get the prediction. The output layer is the output that is obtained after the prediction by the model with an accuracy and confidence. The hidden layers are the main part of the model which process the data for classification, these layers are set based on the requirement of the model, but there general collection can be as described below.

Convolutional Layer: This is the most important layer behind a convolutional neural network, which has an input as a tensor. It convolves upon the input using the mathematical process of convolution and passes it to the next layer. This layer consists of the dimensions as number of images x image height x image depth.

Local Connectivity: It is impractical to connect each neuron with all the neurons in the high dimensional inputs, and such type of spatial structure is not needed in a neural network. Instead, the neural network has a spatially local correlation, by enforcing a sparse local connectivity.

Spatial Arrangement: This layer is used to control the size of the output volume in a convolutional layer, using three parameters which include depth, stride and zero-padding. The depth controls the number of neurons. The stride controls the spatial dimensions like height and width. And the zero-padding controls the padding of zeros for convolution.

Parameter Sharing: This layer is used for determining the number of shared and free parameters in a convolutional neural network. It determines the depth of the slices for input, for which each slice of the input will have same biases and weights.

Pooling Layer: This is one of the other important layers in a convolutional neural network. This layer is used to down sample the size of the input in order to get a better performance while maintaining the same accuracy. It only takes the maximum node from the given slice to the next layer.

ReLU Layer: This stands for Rectified Linear unit, which rectifies the data, to only take the positive parts of it. It applies the softmax function, so the input stays the same if it is positive and becomes zero if it is negative.

Fully Connected Layer: This layer is responsible for classification and getting the output, this layer has all neurons of one layer connected to all neurons of the next layer, for getting the output class for the input.

Loss Layer: This layer determines the loss that occurred while training the model and is normally the last layer of a neural network, to determine performance and accuracy of the model.

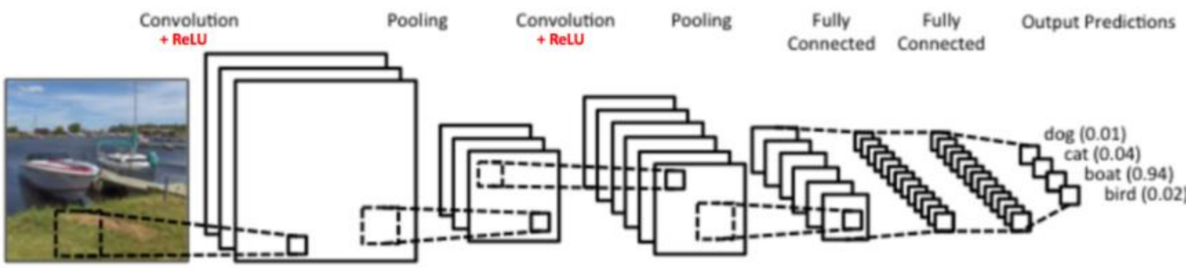

Figure 1: General example of a Convolutional Neural Network

The above shown image is a general example of a convolutional neural network, which is generally made up of two parts feature learning and classification. The feature learning part is used to increase the learning and do the operations on the input, which includes layers like convolutional, pooling, ReLU, Parameter sharing and local connectivity. The classification part is used to determine the class of the input after all the processes, which include layers like fully connected layer and loss layer. The above image has the first layer of convolution, which separates the image into pixels, applies convolution on it, which increases the dimensions of the input. The ReLU layer is applied to soften the input data. Then it sends the processed data to the next layer which is pooling, that is used to reduce the size of the input, based on the type of pooling and the size of the slices. Then we have another set of convolutions + ReLU layer and Pooling layer, which further flattens the input data. Then the classification part starts, which first has the fully connected layer. This layer determines the class of the processed data, based on the previous learning from the training of the model. Then finally, we get an output as a type of classes based on the described classes. In the example, you can see that the input is an image of boat, which is then processed through the layers, to classify it based on the classes like dog, cat, boat bird etc. This explains the working of a general convolutional network, next we will focus on the fully connected layer, to know more about it.

Fully Connected Layer

The fully connected layer in a convolutional neural network is a part of the classification in the model. This layer has an objective to take the results obtained from the feature learning process including layers like convolution, ReLU and Pooling, process them and apply them to classify the image into a label.

For this process, the output from the convolution/pooling is flattened into a single vector of values by the flattening layer. It consists of the different objects present inside the classes. For example, if the input image is of a cat, them the features representing characteristics like fur and whiskers should have a high probability for the tag or label “cat”. When you take into consideration some popular models, you will find that the few layers towards the end are made up of fully connected layers, which compile the data extracted from previous layers to form the final output. It is the most time-consuming layer after convolutional layer.

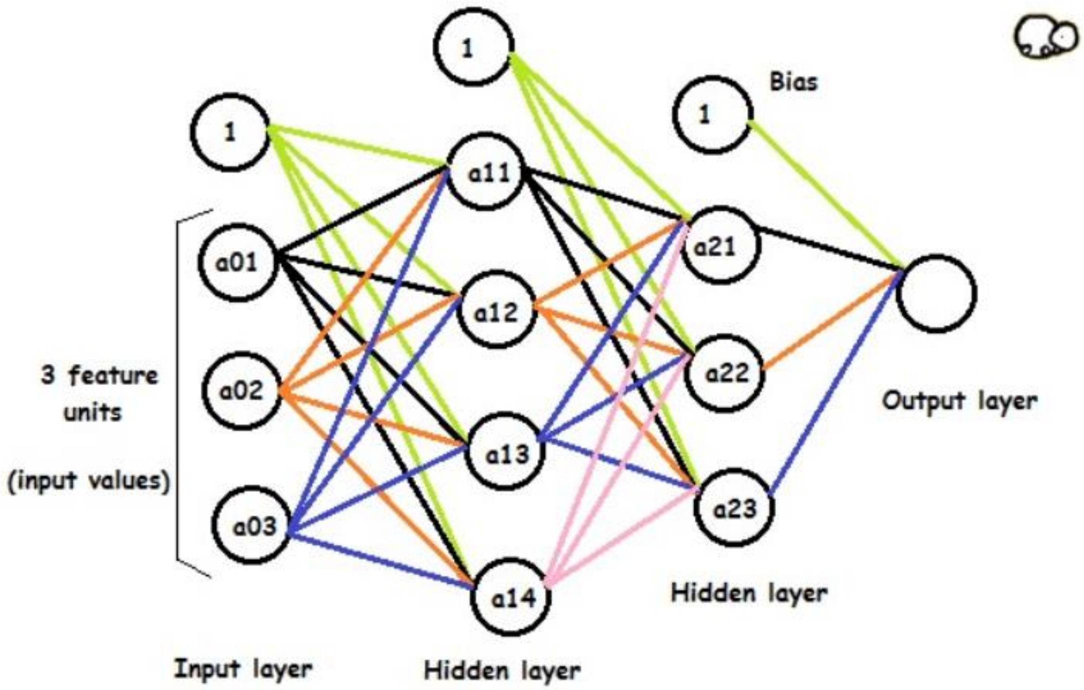

Figure 2: A network with fully connected layers

The above image shows a clear example of the network made up of multiple fully connected layers. The input layer is the input from the previous layers from the feature learning process. It has three feature units all connected to the four activation functions of the next hidden layer. The 1's present in each layer is the bias units. The a01, a02 and a03 are the input values which are the features from the training example. As it can be seen the four activation functions of the first layer are all connected to the three activation functions of the second hidden layer.

So, let us discuss that why are the fully connected layers needed in a neural network. The most important contribution is that it preserves the data obtained from the feature extraction process, in order to use these features to classify the input image into various classes based on the training dataset. Another important feature is that it is also a cheaper way to learn non-linear combination between these features. As the layers like convolution and pooling are only locally connected, the learning from those is only applicable to local features. While from the fully connected layer, a learning of the combination of features can be obtained due to their non-linearity and connections. The sum of the outputs of the probabilities of a fully connected layer need to be 1. This process is done by using the softmax function for activation in the output layer after the fully connected layer. It takes the vector which has arbitrary real numbers as values and processes them to obtain the output which is between zero and one, and it also ensures that their sum is one.