Deep Learning

Dr.Jawwad Ahmad Shamsi ,
National University of Computer and Emerging Sciences, Karachi Campus, Pakistan

March 4, 2024

1 Inception Network

In the field of CNN classifiers, the Inception network was a major breakthrough moment. Most common CNNs, prior to inception, simply stacked convolution layers deeper and deeper in the hopes of improving efficiency. However, in real time applications, the size of important sections of the picture will vary dramatically. For example, a dog image may be one of the following, as shown below. In each picture, the dog occupies a different location.

From left: A dog occupying most of the image, a dog occupying a part of it, and a dog occupying very little space (Images obtained from Unsplash).

Figure 1: Predicted vs actual output

Choosing the right kernel /filter size for the convolution operation becomes difficult due to these large variation in the position of the salient objects. For information, that is distributed more globally (uniformally over the entire image), a larger kernel is preferred, whereas for information that is distributed more locally, a smaller kernel is preferred.

Overfitting is an issue for very deep networks. It's also difficult to propagate gradient updates through the entire network. Stacking massive convolution operations in an inefficient manner is computationally costly.

1.1 Filter Size

We can choose different size of filters. For instance, 5X5, 3x3, and 1x1. These three sizes can be used to detect different sizes of objects.

5x5 can detect larger objects, 3x3 can detect medium-sized objects, whereas 1x1 can detect small objects. Figure 2 explains the concept.

The three different size of filters should produce the same size of feature map. This is because the convolution from the three sizes will be cascaded together. In other words X and Y dimensions of the three feature maps produced through convolution of these three filters (5x5, 3x3, and 1x1) should be same. The Z dimension will be cascaded together.

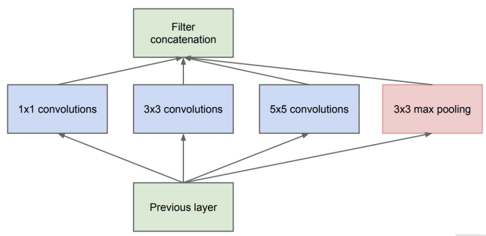

Figure 2: Inception

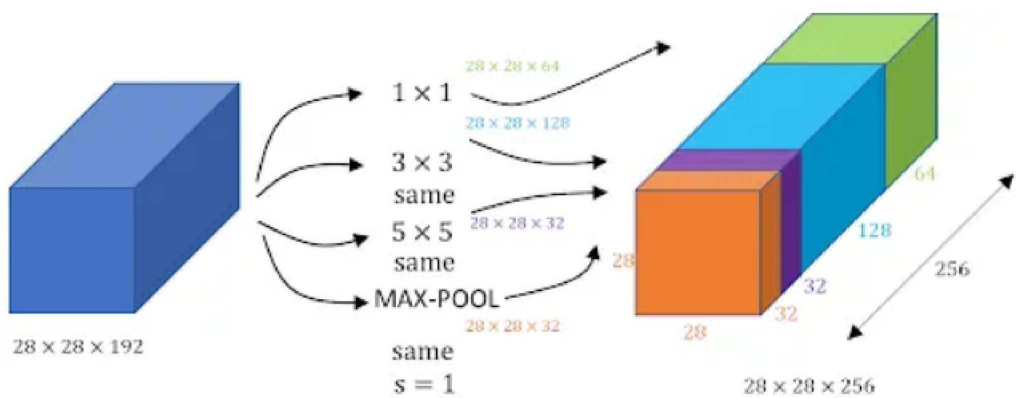

Figure 3: Inception Example

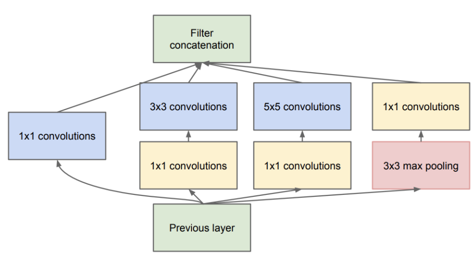

Figure 4: Adding 1x1 Conv

Example 1 ***Similar size of feature maps***
To yield similar size feature maps, we should use padding.

Remember the size of feature map is given by

$sizeoffeaturemap = (n + 2(P) - f)/S + 1$

where n=size of input f=size of filter p=size of padding s= stride

A 5X5X192 filter on the image size of 28X28X192 with no padding and single stride will give us 28+2(0)-5/1 +1 will give us the output of 24 x 24

Similarly, 3X3 filter will yield a feature map of size 26X26 and a 1x1 filter will yield a feature map of size 28X28.

In order to have a similar size, we can apply 2X2 padding on the input for 5X5 filter and 1X1 padding on images with 3X3 filter.

Figure 5 illustrates the concept of adding of 1X1 conv before the actual filter of 5x5, 3x3, and 1x1.

Example 2 ***How 1X1 conv helps***

A 5X5 filter with 2X2 padding on a 28X28 input image will yield a 28x28 feature map. Total number of operations are 5X5X192X28X28X32 = 120 Million.

Remember that 5X5X192 is the dimension of the input filter, whereas 28X28X32 is the resultant feature map. In that 32 is the no of 5x5 filters being convoluted.

This number can be reduced by applying a 1x1 conv first

If we apply 16 filters of 1x1x192, then the resultant feature map will be 28X28X16. Applying 32 filters of 5X5X16 on the this feature map (after padding), will result in the feature map of 28X28X32. However the total no of operations will be

1) 28X28X16X1X1X192 =2.4 Million 2) 28X28X32X5X5X16 = 10 Million

Total = 12.4 million. This is a 10-time reduction.

Similarly inception with 1X1 conv. can be applied before 3X3 and 1X1.

The whole idea can be understood from figure 5

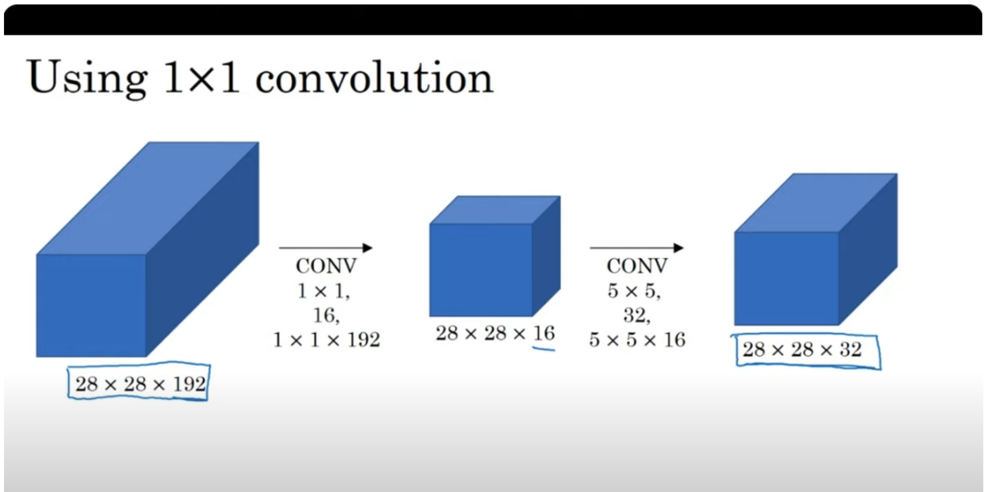

Figure 5: 1x1 and 5x5