

Hack-a-thon 5 Homework

Conceptual Problems

1. Write a few sentences about the differences between the ResBlock and Bottleneck layers above. Why might the Bottleneck block be more suitable for deeper architectures with more layers?
 - A ResBlock is block of layers within the range of skip connection, the skip connection is a judging condition to judge whether let the signal (predicted value in each layer) pass to layers in block computation to eventually make signal value smaller(which leading to vanishing gradient Problem), or passing through the skip connection which in order to bypass the computation using if-else condition to prevent vanishing gradient due to the smaller and smaller value after each layer's computation. The block of layers within the if statement(or skip connection) is one ResBlock/Residual Block. [The Residual block help the model reduce its error rate below 30% compare to PlainNet in page 5 of paper.](#)
 - A Bottleneck Residual Block (or Deeper Bottleneck Architectures) is less computationally intensive, the use of a bottleneck reduces the number of parameters and matrix multiplications, in other words, to [reduce training time](#) using identity matrix(shortcut). Identity shortcuts are particularly important for not increasing the complexity of the Deeper bottleneck architectures.

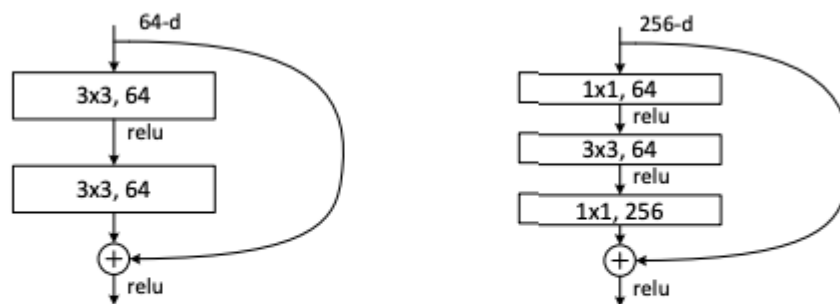


Figure 5. A deeper residual function \mathcal{F} for ImageNet. Left: a building block (on 56×56 feature maps) as in Fig. 3 for ResNet-34. Right: a “bottleneck” building block for ResNet-50/101/152.

As picture showed above from the [paper](#), the bottleneck block has one more 1×1 convolution with 56×56 feature maps.

The 1×1 layers are responsible for *reducing and then increasing (restoring) dimensions*, leaving the 3×3 layer a bottleneck with smaller input/output dimensions. Both design with different dimensions has similar time complexity thanks to the identity shortcut(skip connection).

- The skip work like this, if the signal is too small, we will decide to use the value before the block computation, which has larger value to pass to activation. If the value after compute is large enough, we can make a decision to use the actual computed value.
2. Write some python code which builds a network using the general structure described above and either ResBlock or Bottleneck blocks. It doesn't have to be a full set of code that runs, just a function or class that builds a network from these blocks. You might find this architecture useful for homework 1. (Code in notebook)