Residual Networks Summary

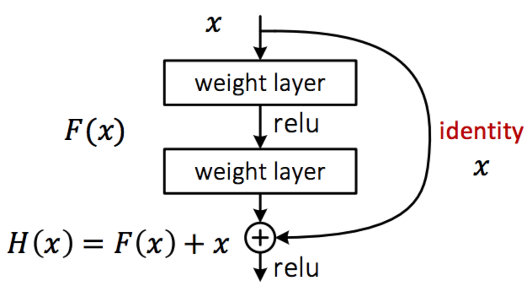
Danny Luo

Summary

He et al. 2016. Deep Residual Learning for Image Recognition.

Deep neural networks have higher training error than shallower versions, making them more difficult to train. To train deep neural networks more easily, the layers are reformulated to learn the residual function of $\mathcal{H}(x) - x$, instead of underlying representation $\mathcal{H}(x)$. The authors demonstrate that the assumption optimal mapping is closer to the identity mapping is a reasonable precondition.

To implement a residual block, a shortcut identity connection is added between the input and the output, so that the stacked nonlinear layers learn the residual mapping $\mathcal{F}(x) := \mathcal{H}(x) - x$.



Adding shortcut connections allows for lower training and validation error over the same number of iteration when compared to architectures without shortcut connections. This allows residual neural networks to be stacked very deep without becoming very difficult to train. The idea of residual learning can be transferred to any style of neural networks and to many other types of machine learning models.

Deep ResNets can have up to 8x more layers than VGG while having lower complexity. ResNets of up to 152 layers achieved first place on ImageNet 2015 and COCO 2015.

Questions

1. Do Resnets resolve the vanish gradient problem by constructing many shallow neural networks that attempt to learn residuals and hence require substantial gradients?

2. Could another function other than the identity be used provided we have some knowledge of the input in relation to the output? Could that be learned?

3. You can only add the identity shortcut function for blocks with the same input and output size. If they have different sizes, could one modify the input – copy, chop off, sum – to make it work?