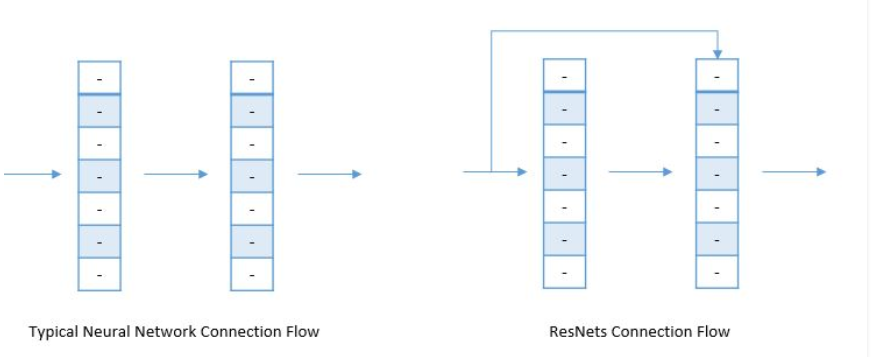
Artificial Intelligence Group Assignment

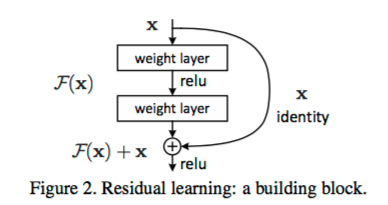
# ResNet

Deep neural networks have led to a vast number of breakthroughs in the field of image classification, where they excel due their ability to integrate low, medium and high-level features and classifiers in an end-to-end multi-layer fashion. The issue, however, is that deep neural networks are extremely hard to train, because at each training iteration, all weights receive an update proportional to the partial derivative of the error function respective to the current weight. If the gradient is small enough, then the weights will no longer be changed effectively, and it could stop the neural network from being accurately trained further – a phenomenon commonly referred to as vanishing gradients.

Microsoft research discovered that it was possible to split the deep neural network into three layered chunks, passing the input directly into each segment together with the residual output of the previous chunk minus the input that is reintroduced (known as skip-connections). This helps eliminate the vanishing gradient problem common in other deep neural network implementations without any changes to parameters or learning algorithms.

# Architecture

Because of their large depth and the fact that the vanishing gradient problem is solved, ResNet’s with a depth of 152 have achieved an accuracy of 95.51% in top-5 accuracies; all while being more computationally efficient than the VGGNet.



Each block in a standard ResNet architecture follows the same principle, a 3x3 convolution and pooling step followed by 4 layers of similar behaviour. This is then repeated several times depending on the size of the resnet, comparing different map dimensions (64, 128,256, 512) respectively – bypassing the input every second convolution. A standard ResNet implementation is generally built using two different types of blocks, the Identity block and the Convolutional block.

## Identity Block

The identity block is used in the standard case where the input activation has the same dimensions as the output activations.

Convolutional Block

On the other hand, whenever the input and output dimensions don’t match, a convolutional layer is added to the shortcut’s path – converting the identity block into a convolutional block.

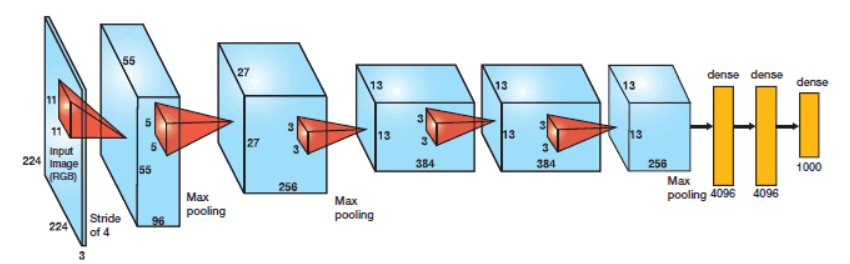
By repeatedly stacking different blocks together, the ResNet can achieve high accuracies through its deep neural network architecture

[1] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[2] He, K., Zhang, X., Ren, S. and Sun, J., 2016, October. Identity mappings in deep residual networks. In *European conference on computer vision* (pp. 630-645). Springer, Cham.

[3] Xie, S., Girshick, R., Dollár, P., Tu, Z. and He, K., 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1492-1500).

# AlexNet

The AlexNet architecture was one of the first deep neural networks that pushed the accuracy of the ImageNet Classification challenge by a significant amount when compared to previously traditional techniques.

The original architecture was composed of 5 convolutional layers followed by 3 fully connected layers as seen below. The main difference to previous neural network implementations was that the AlexNet made use of the ReLu (Rectified Linear Unit) for non-linear parts rather than using the Tanh or Sigmoid functions.

f(x) = max(0,x)

The advantage of this function is that is has the ability to train much faster because the previous methods suffered from the vanishing gradient problem (solved in the ResNet architecture described above).

Another significant improvement in the AlexNet architecture, was that after every fully-connected layer, a dropout layer was added to reduce over-fitting. By applying a probability to every node, randomly selected nodes activations would be turned off with the probability assigned in this layer. The idea behind this method is that since neurons are randomly dropped, they tend to avoid developing corresponding adaptations together – instead enabling them to develop important, node-independent features.

