CSE4261: Neural Network and Deep Learning

Lecture: 17.06.2025

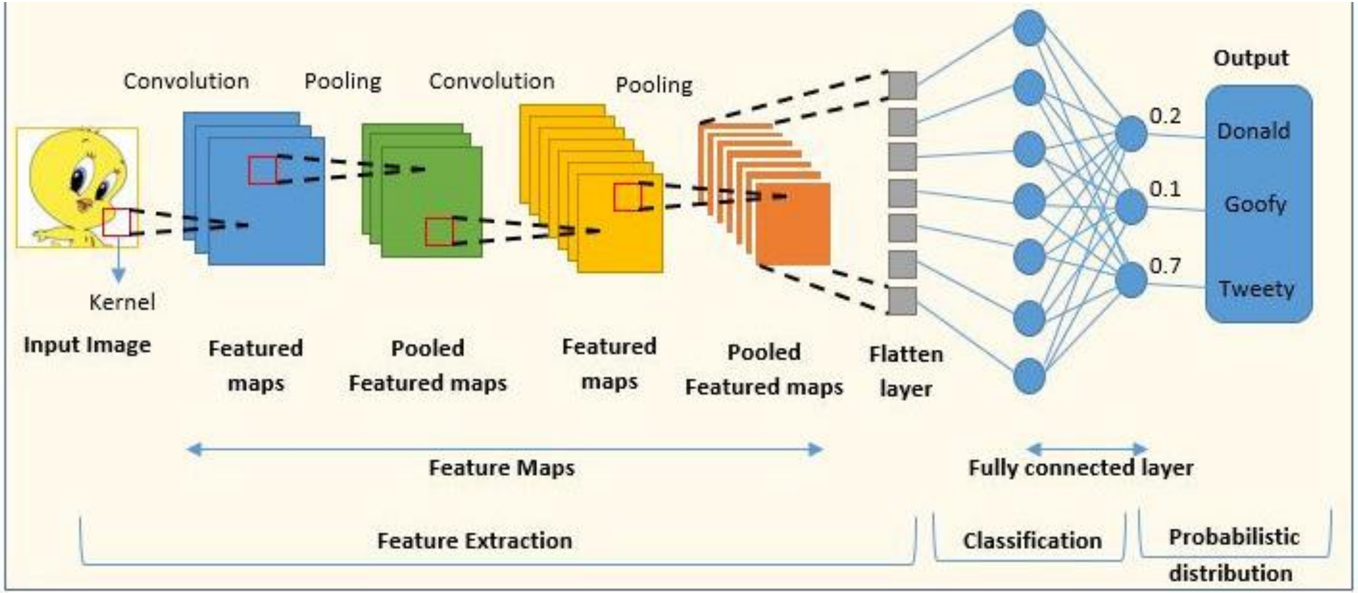
Sangeeta Biswas, Ph.D.
Associate Professor,
University of Rajshahi, Rajshahi-6205, Bangladesh

Convolutional Neural Network (CNN)

- CNN mainly consists of a series of convolutional layers and downsampling layers.
- For classification, fully connected layers are added after convolutional layers.

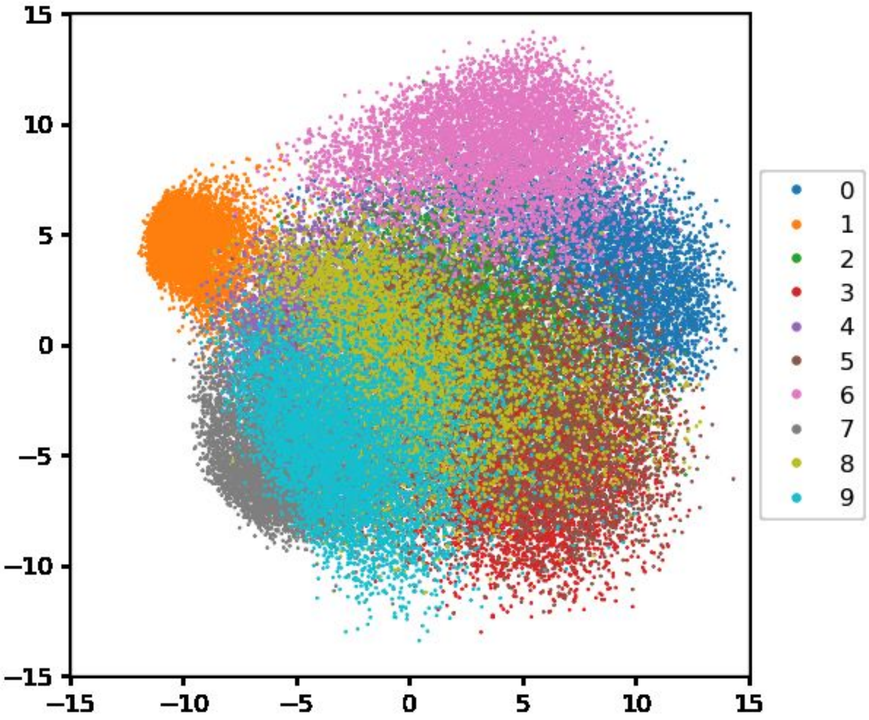
Popular CNNs:

* VGG
* InceptionNet
* ResNet
* EfficientNet
* MobileNet
* XceptionNet
* DenseNet



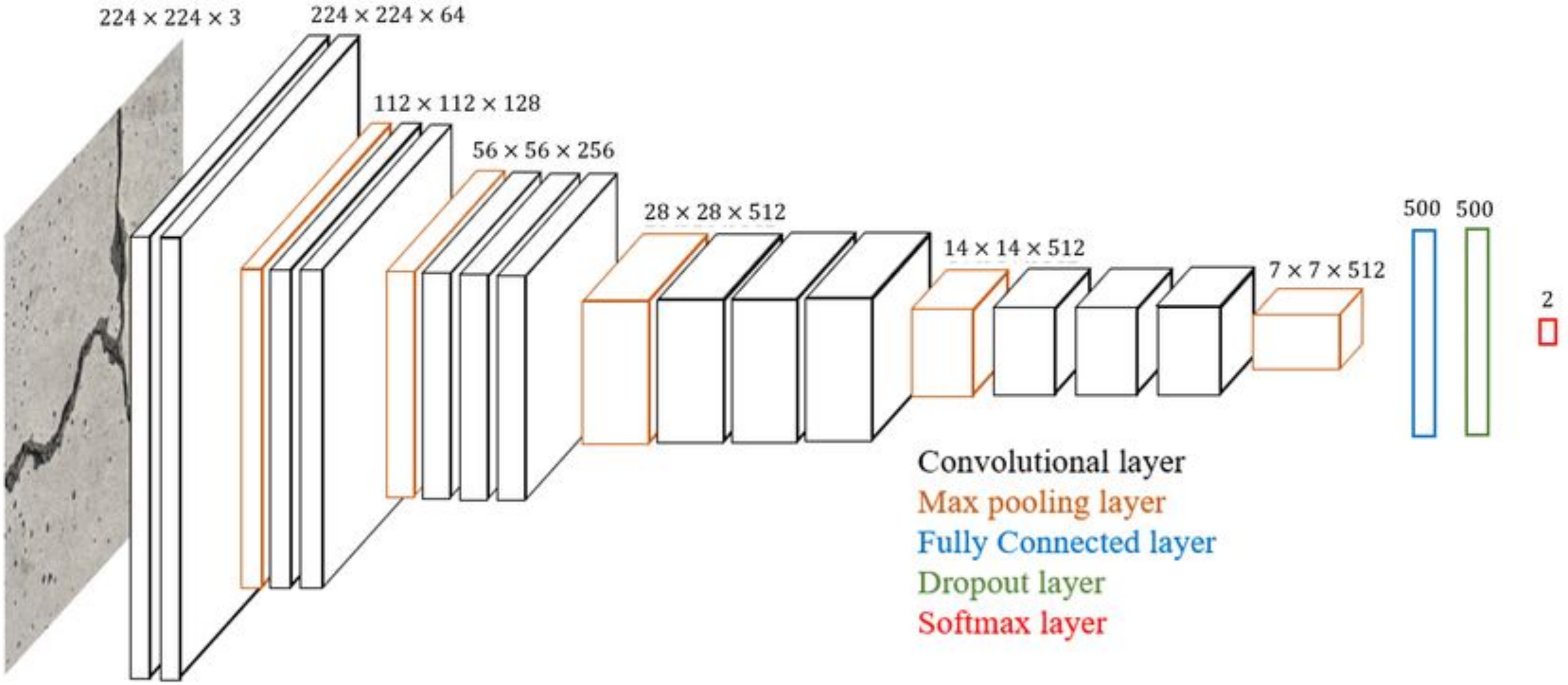
Different Feature Visualization

- PCA: Principal Component Analysis
- TSNE: t-distributed Stochastic Neighbor Embedding
- MDS: Multidimensional Scaling
- ISOMAP: Isometric Mapping
- LLE: Local Linear Embedding



Visualization of features of MNIST Digit

VGG16



224 × 224 × 3
224 × 224 × 64
112 × 112 × 128
56 × 56 × 256
28 × 28 × 512
14 × 14 × 512
7 × 7 × 512
500
500
2
Convolutional layer
Max pooling layer
Fully Connected layer
Dropout layer
Softmax layer

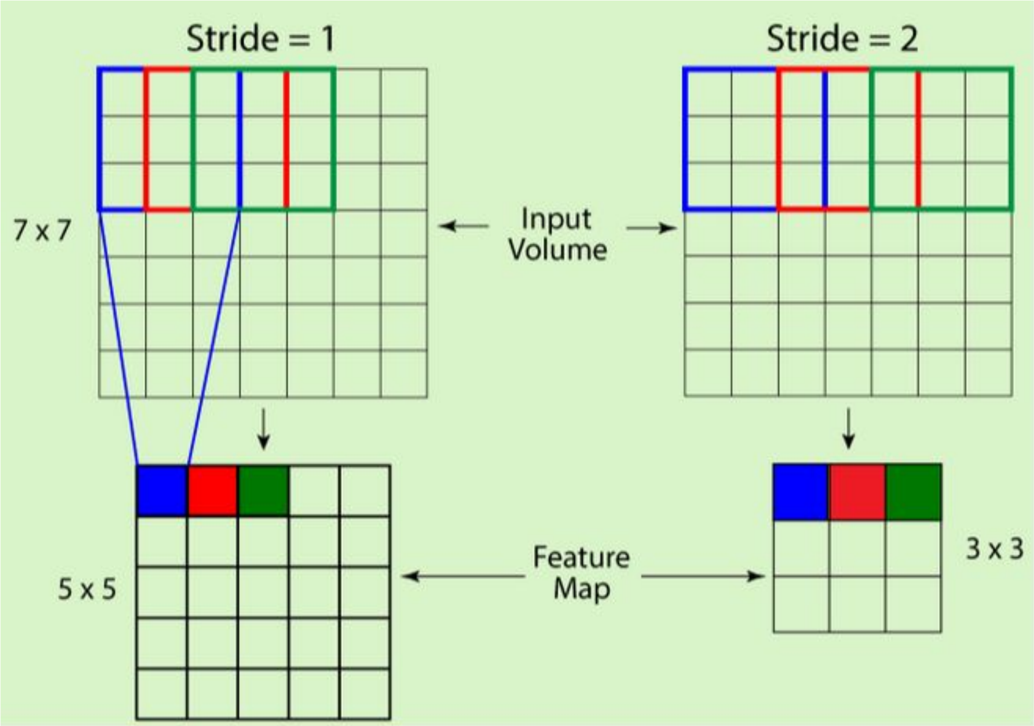
Downsampling Layer

A *downsampling layer* reduces the dimensionality of the features at cost of a some loss in information.

- Convolution layer with larger strides and no padding
  - It is a learnable operation, whereas pooling is a fixed operation
  - Some research works showed better performance
- Different pooling layers
  - Max pooling
  - Average pooling
  - Global average pooling

Downsampling by Convolutional Layer

Without padding, setting appropriate stride value, we can downsample feature map by convolution.



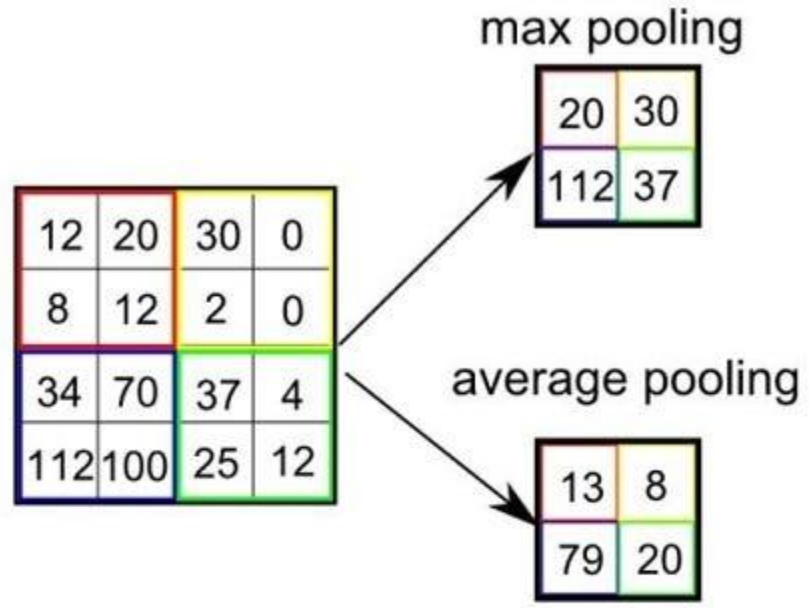
Pooling Layer

It reduces the spatial dimensions of feature maps, thereby:

- decreases computational cost
- makes the network more robust to variations in the input

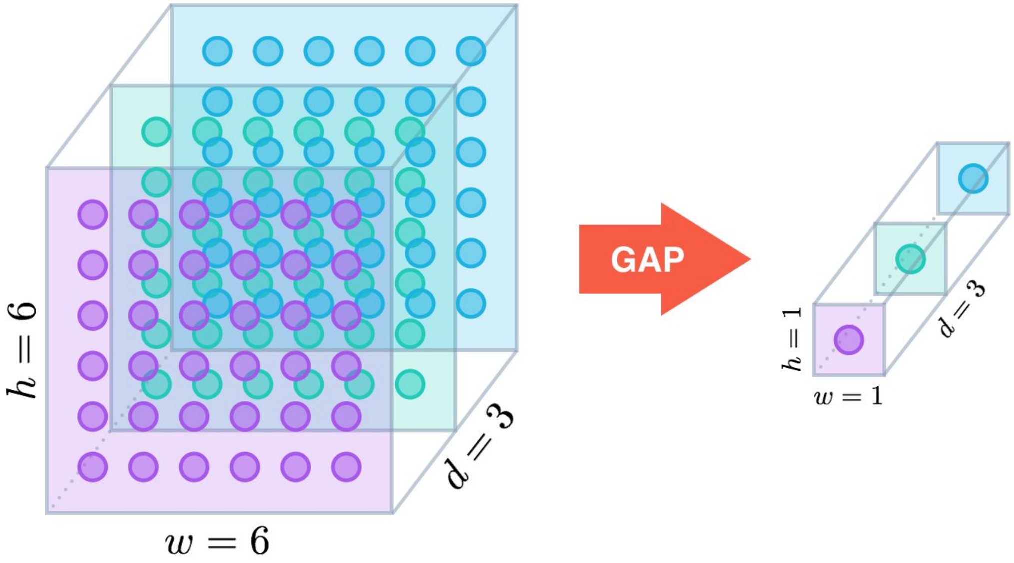
Different Types of Pooling Layers:

- Max Pooling
- Average Pooling
- Global Pooling
- Stochastic Pooling (not very common)

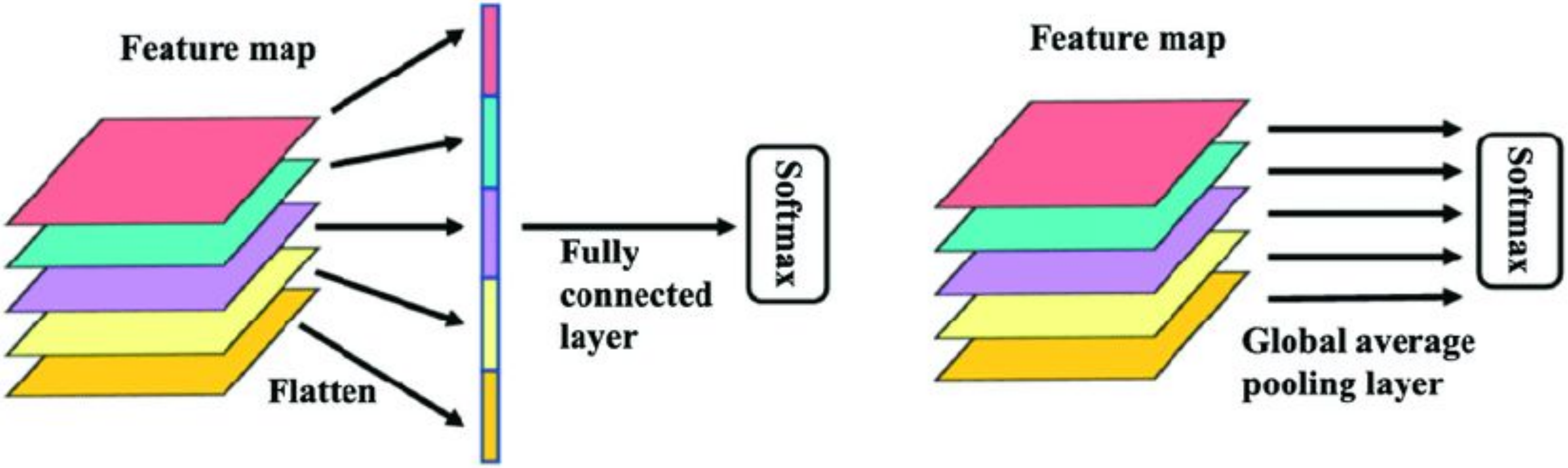


Global Pooling

- It reduces the spatial dimensions of feature maps to a single value per channel.



Usage of GAP



(a) Fully connected layer

(b) Global average pooling layer

InceptionNet

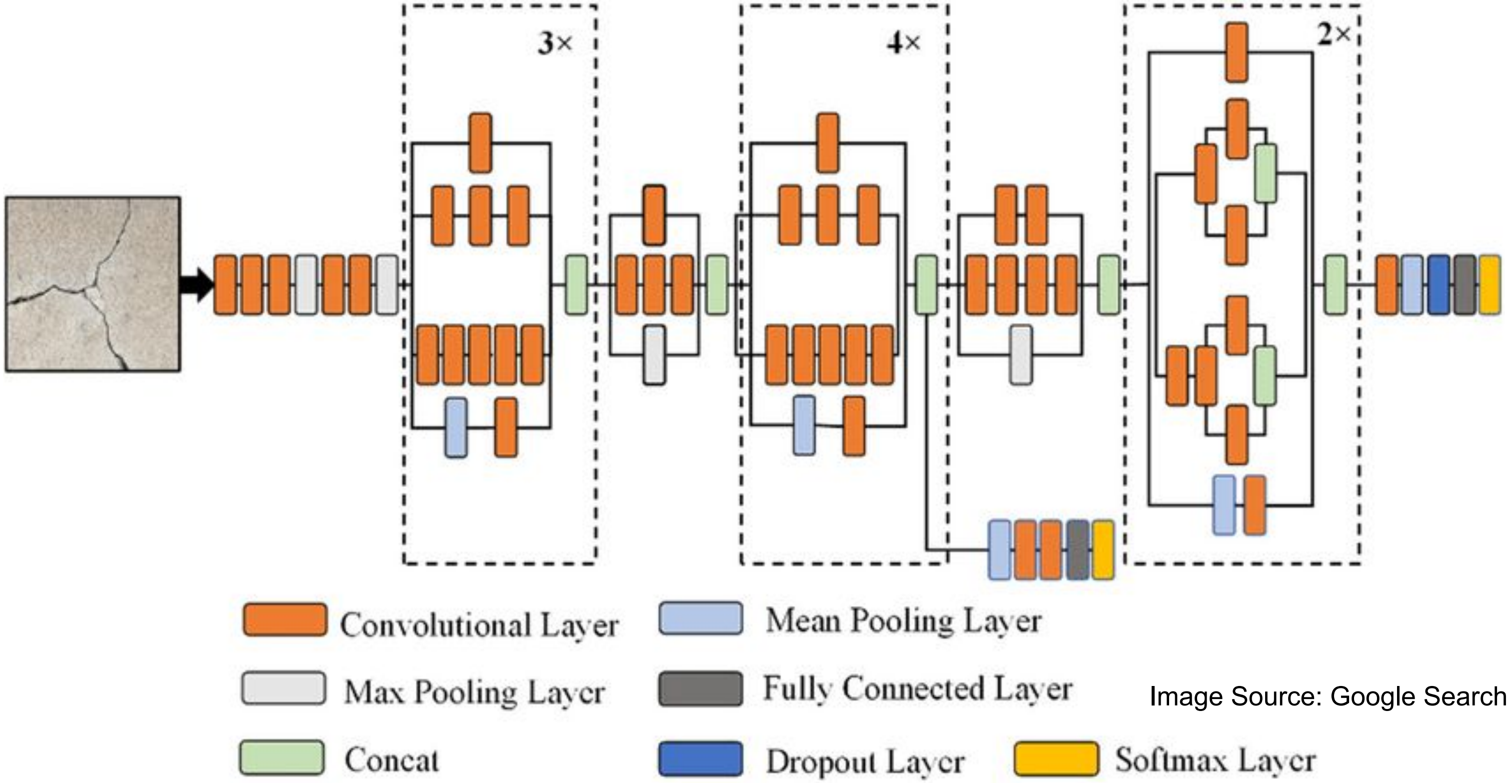
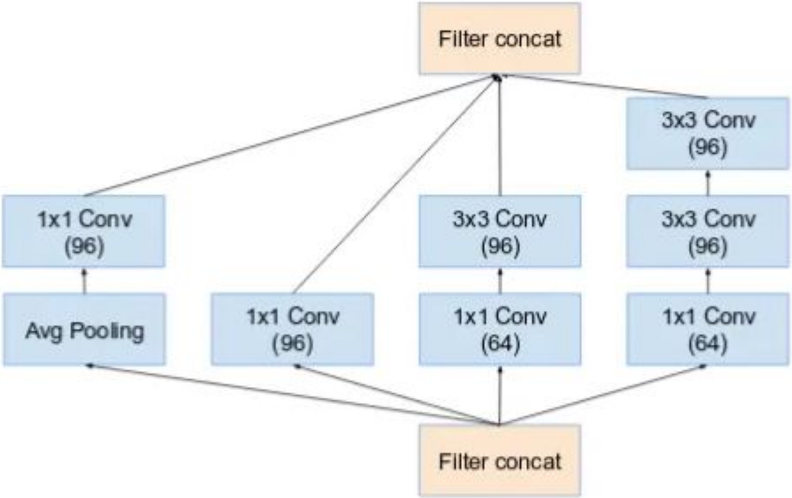
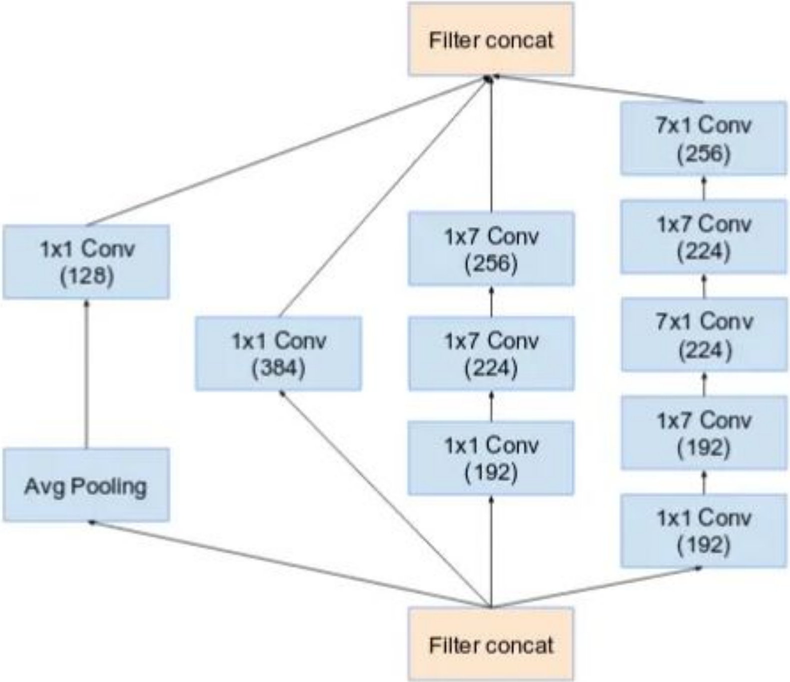


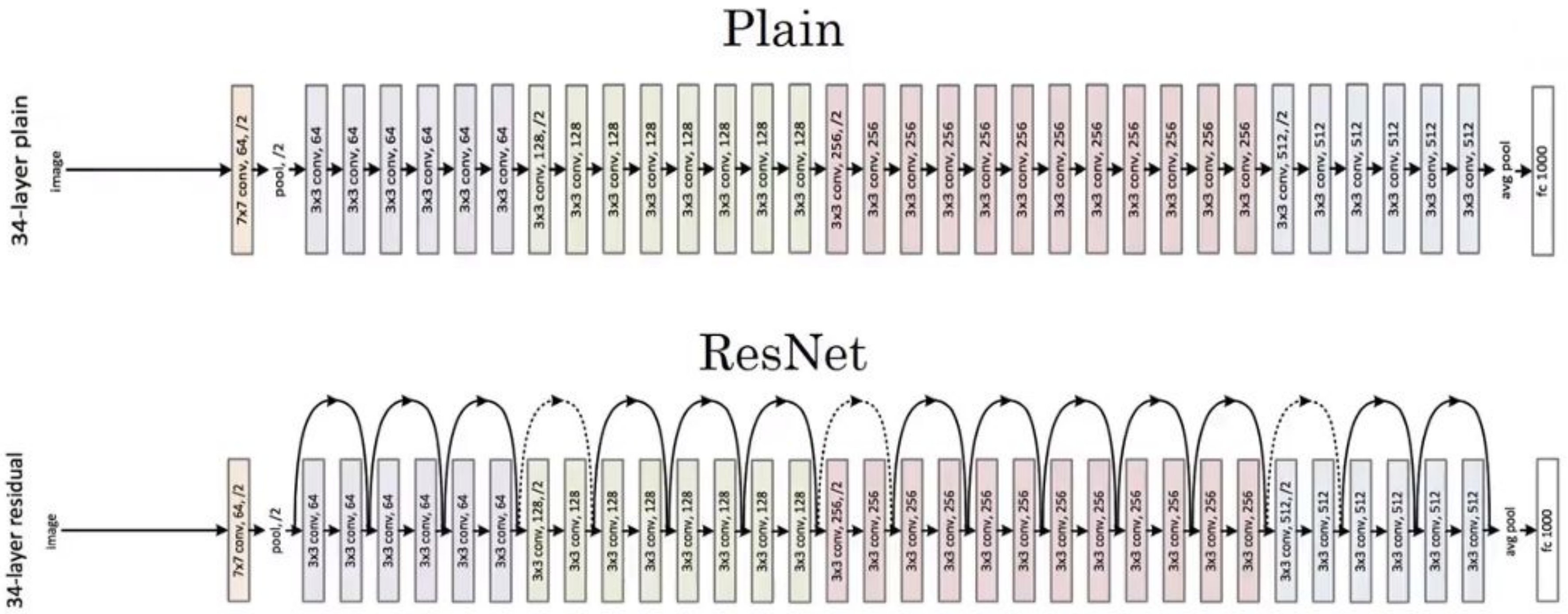
Image Source: Google Search Engine

Inception Modules

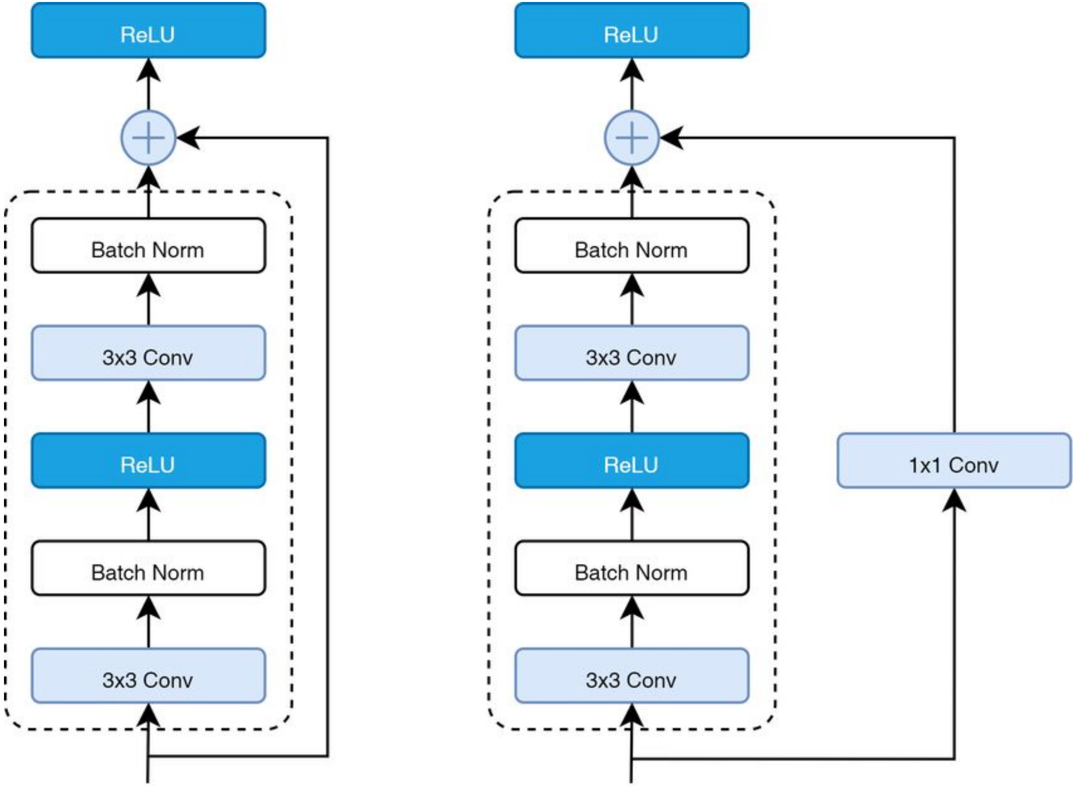




ResNet

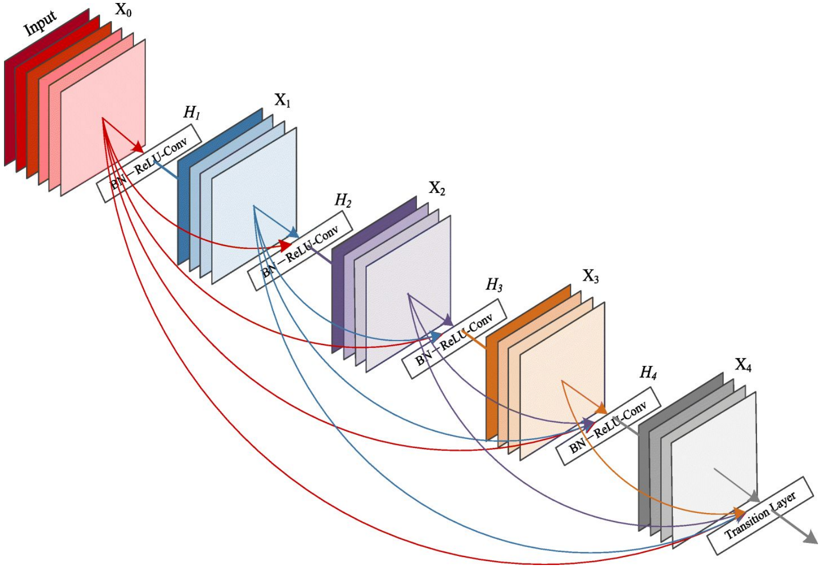

Plain
34-layer plain
image
7x7 conv, 64, /2
pool, /2
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 128, /2
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 256, /2
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 512, /2
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
avg pool
fc 1000
ResNet
34-layer residual
image
7x7 conv, 64, /2
pool, /2
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 64
3x3 conv, 128, /2
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 128
3x3 conv, 256, /2
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 256
3x3 conv, 512, /2
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
avg pool
fc 1000

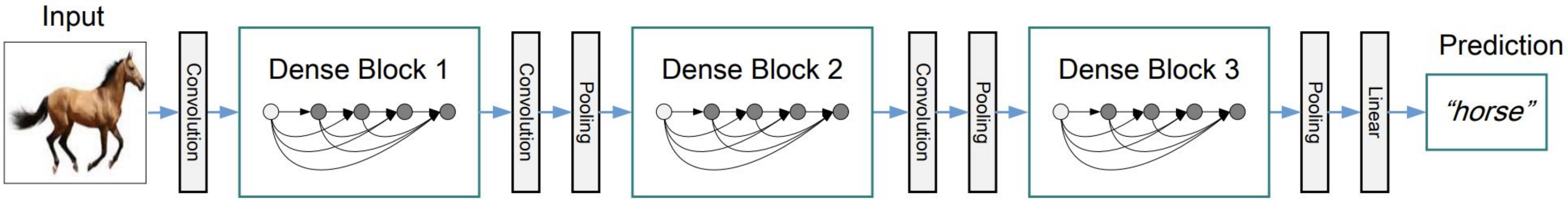
ResNet Block



DenseNet

Skip connections are used for having feature reusability.



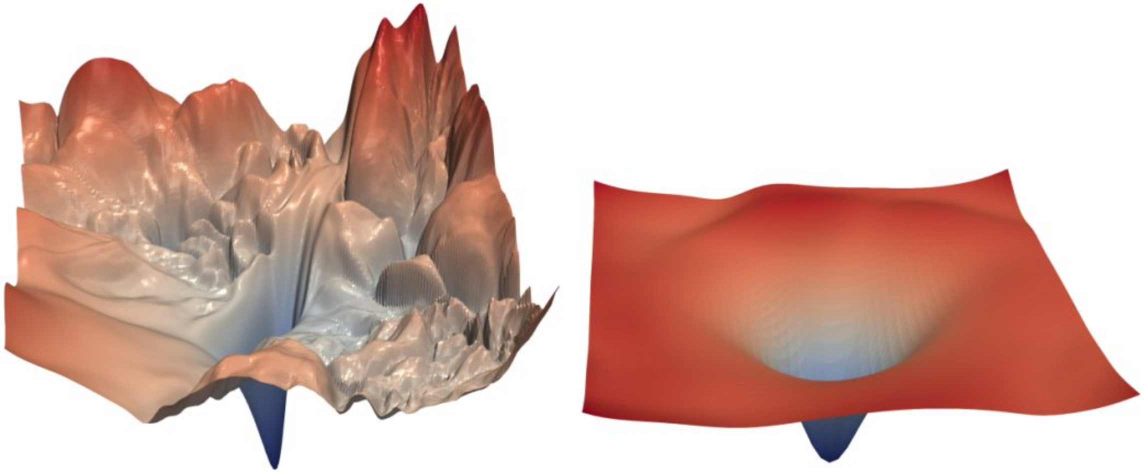


Skip Connections

Skip connections allow CNNs to bypass some layers and connect directly to deeper or shallower ones.

Skip connections lead to:

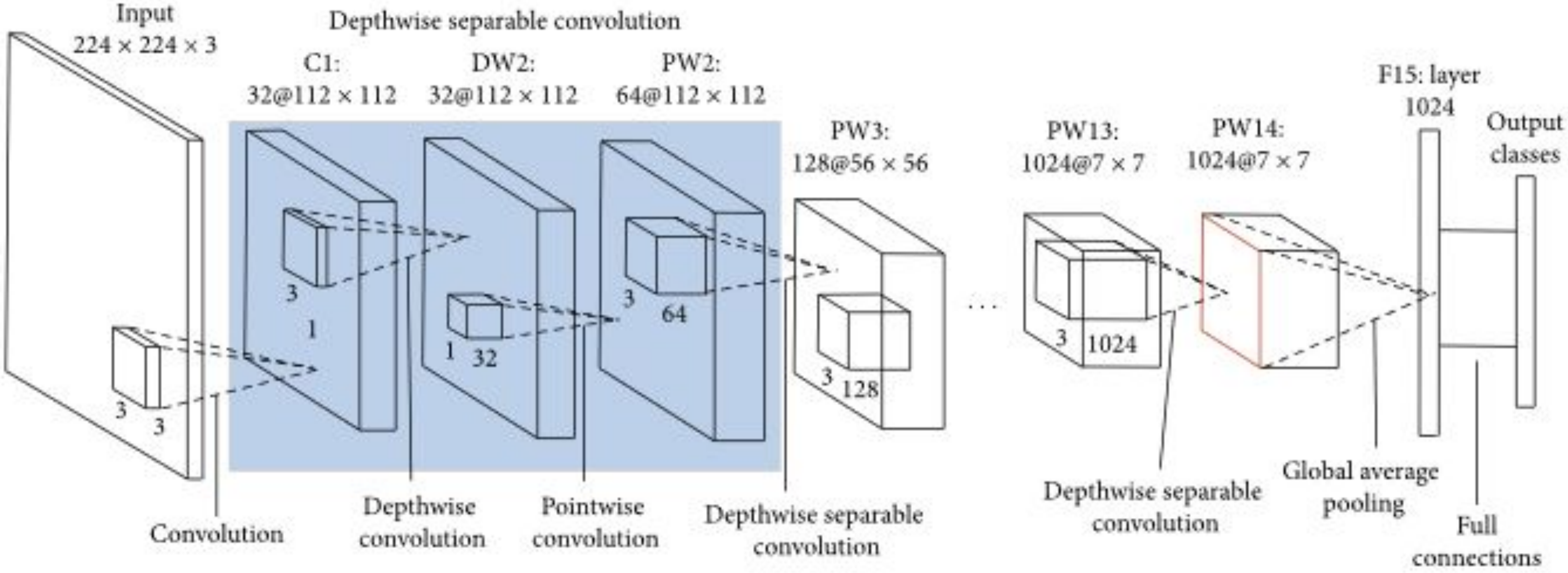
- faster convergence
- vanishing gradient problem prevention
- loss of information prevention



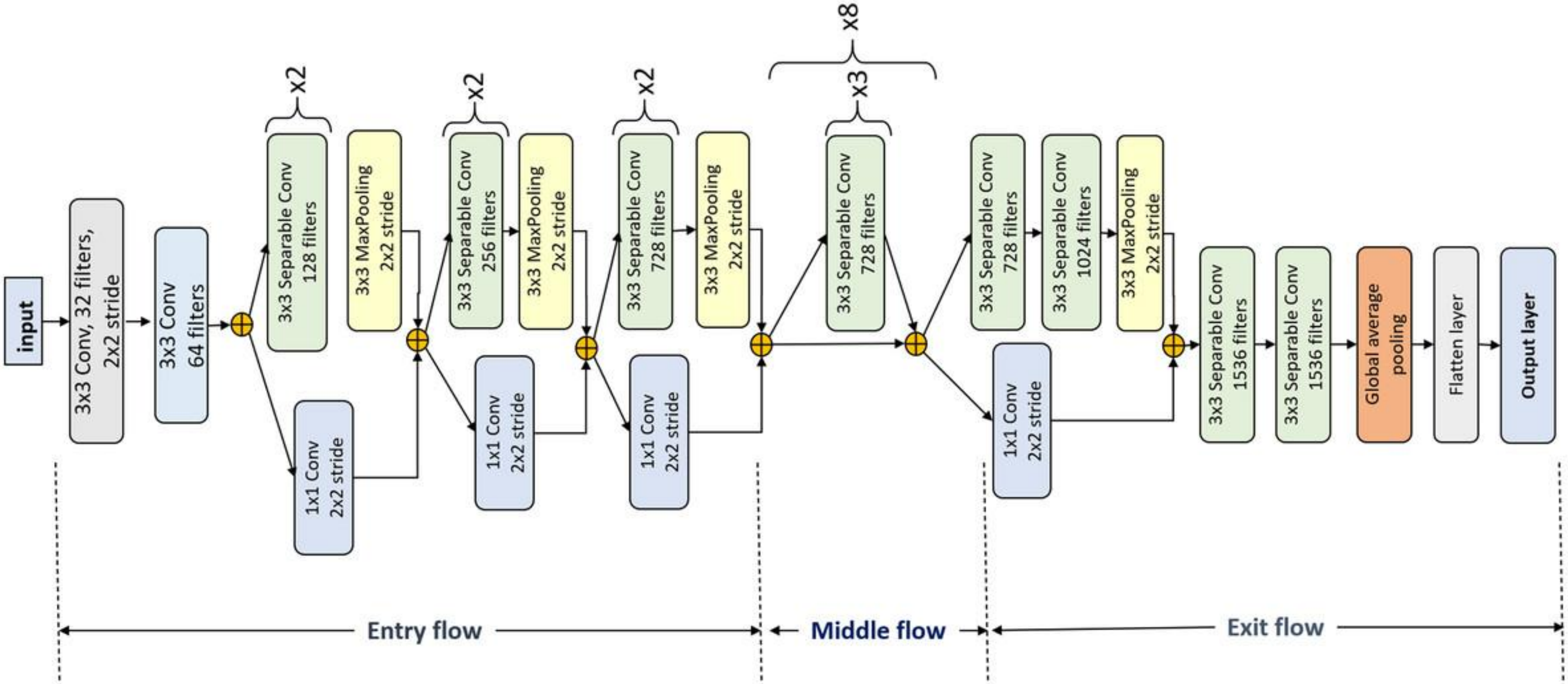
(a) without skip connections

(b) with skip connections

MobileNet V2



Xception (Extreme version of Inception)



How to Avoid Overfitting

- Model Validation
  - The error of an overfitted model is low when computed for the training data. We need to test performance of our model on a separate data set (i.e., validation data set) before using real-time application data.
- Regularization
  - Regularization applies either a penalty for complexity or roughness to the model. By introducing additional information into the model, regularization algorithms can deal with multicollinearity and redundant predictors by making the model more parsimonious and accurate.
  - https://www.youtube.com/watch?v=wNdYpD2YhSk&t=331s

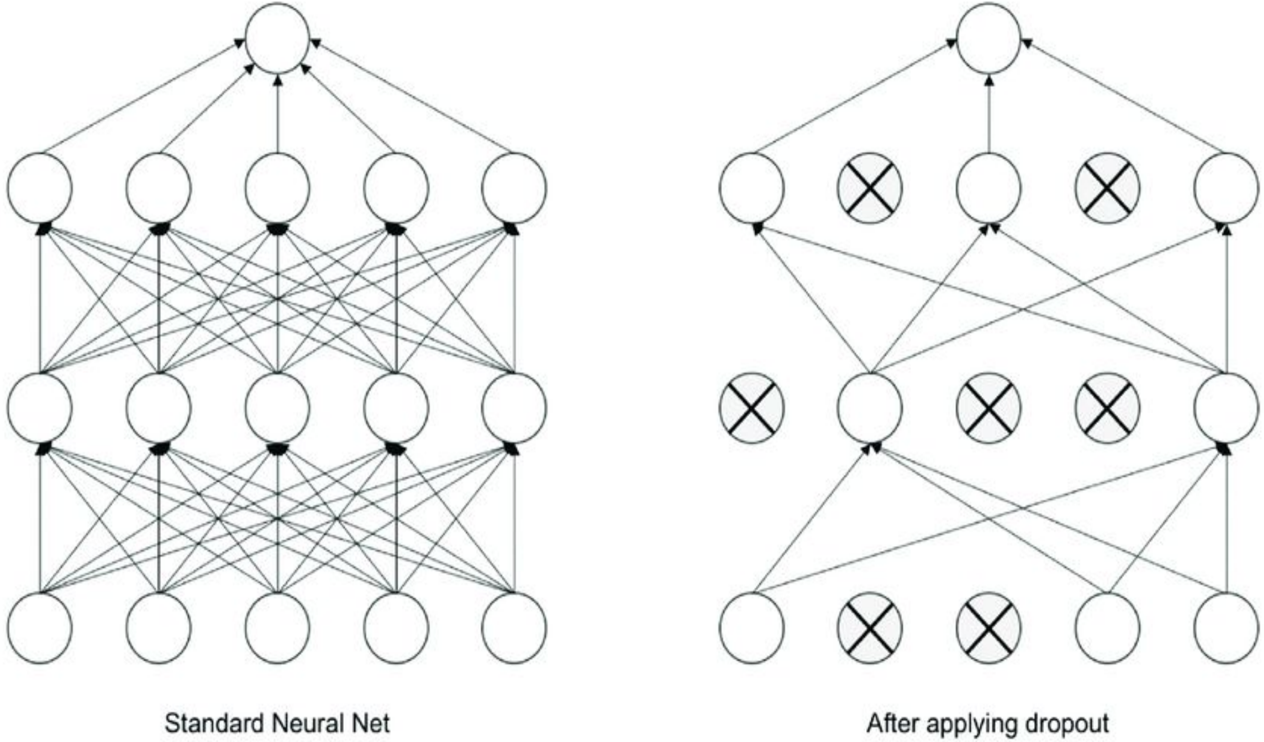
How to Avoid Overfitting

- Data Augmentation
  - When data availability is limited, data augmentation is a method to artificially expand the data points of the training data set by adding randomized versions of the existing data to the data set.
- Synthetic Data
  - Synthetic data generated by generative model can bring variability in available data, increase size of training dataset and reduce overfitting issue to some extent.
- Data Clean Up
  - Data noisiness contributes to overfitting. One common approach to reduce undesired data points is to remove outliers from the data.

Dropout Layer

In dropout layer:

- individual nodes are excluded in various training runs
- prevents overfitting



Dropout Layer

- It activates during training phase only, not in validation and testing phase
- Forward Pass:
  - A certain percentage of neurons in the dropout layer are randomly picked to be "dropped" (deactivated).
  - The output of the dropped neurons is set to zero, effectively removing them from the network's calculations for that specific iteration.
- Backward Pass:
  - The gradients for the dropped neurons are effectively ignored. They don't contribute to the weight updates.
  - The gradients for the active neurons are scaled by a factor that accounts for the dropout probability