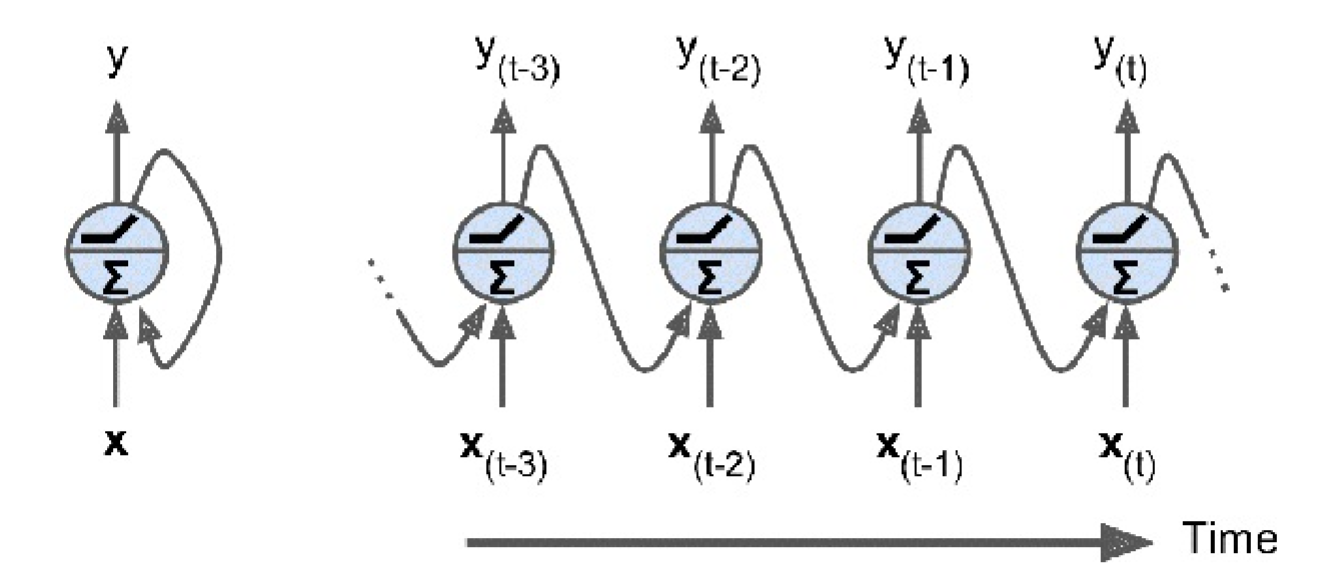
A short note on CNN And RNN

They are both different architectures of Neural Nets that perform well on different types of data. RNNs(Recurrent Neural Networks) are good with series of data (one thing happens after another) and are used a lot in problems that can be framed as “ what will happen next given… ” while CNNs(Convolutional Neural Networks) are especially good at problems like image classification.

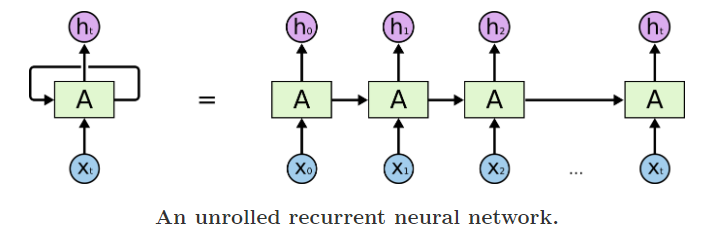
# Recurrent Neural Networks(RNNs)

RNNs (Recurrent Neural Networks) are a class of networks that help predicting the future.They can analyze time-series data such as stock prices and predict when to sell or buy.They can take sentences,documents or ‘audio samples’ as input sequences and output a sequence or a vector as per the application.It can have a sequence-to-vector network called encoder followed by a vector-to-sequences network called decoder , used for translating languages.



An RNN is nothing but a chain of simple standard neural networks which performs the same as the standard neural network but at every time step taking into consideration the present input as well as the past inputs in the form of previous hidden state.A RNN unit is fed with the present input and the previous hidden state and it outputs a prediction and a hidden state that becomes the input of the RNN unit at the next time step.

In simpler words,a recurrent neural network can be thought of as multiple copies of the same network, each passing a message to a successor. Consider what happens if we unroll the loop:



source:[colah’s blog](http://colah.github.io/posts/2015-08-Understanding-LSTMs/)

One must not confuse *time step* with the *real time*.It just signifies a sequence of data which can be a sequence in time as in the case of speech input or a sequence of words.

# Convolutional Neural Networks(CNNs)

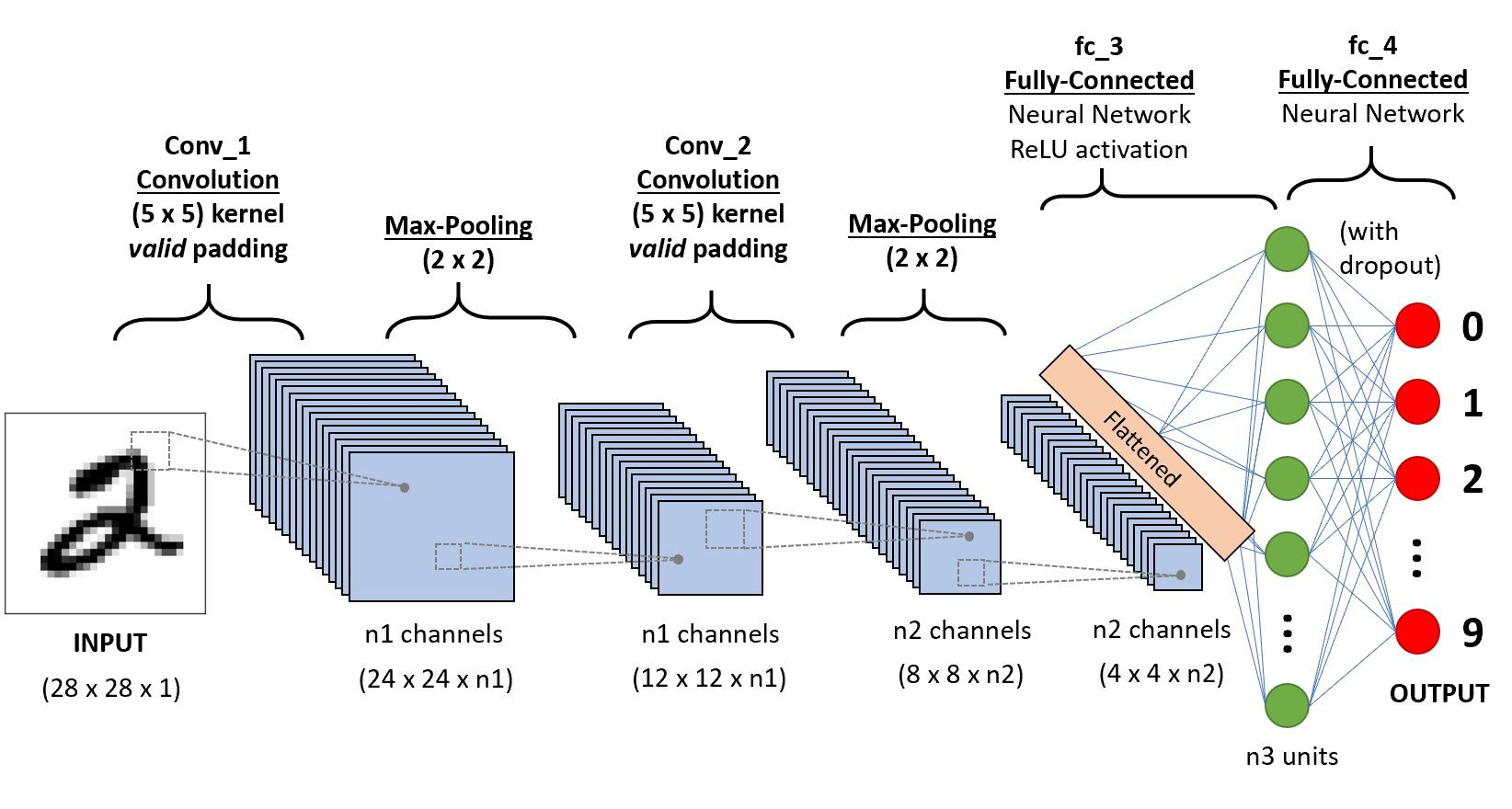
Deep learning is enabling computer vision where applications lies in image processing,obstacle detection,neural style transfer and many others.CNNs have proved to be excellent working on images.

CNNs (convolutional neural networks) essentially have three parts, convolution layers, pooling layers, and fully-connected layers. It usually takes a 2D (sometimes more dimensions) matrix of it pixel values and outputs a result.

Convolution starts at the top left and takes a small window with a certain width and height and performs an operation on that, the operation is usually a matrix multiplication where the matrix to multiply by is decided via gradient descent to get the best final results. It then moves according to a stride parameter and does the same. It does this all the way across the image and outputs a new image.

Pooling is similar in the sense that it breaks the image down using small windows; however, the operation it runs on this small window is usually (average, max, or min) to combine the small window into a single pixel.

After a set amount of convolutions and pooling, the final output is put through a fully connected layer, which is a conventional feed forward neural network to output a result.



source:[towardsdatascience.com](https://towardsdatascience.com/)

Why did we not use standard neural networks(feedforward neural network) instead of CNN?

CNN contracts an input image into a smaller and meaningful channels of data,hence reducing the number of parameters to train and the training is better with CNNs as it considers every pixel value more than once.However,we use the standard neural networks towards the end whose behavior is best shown in this [video](https://www.youtube.com/watch?v=aircAruvnKk&t=494s).

The parameters in CNN are the weight matrices used as filters during convolution.The parameters in these matrices are tuned during backpropagation via gradient descent or any other optimization algorithm.

Intuitively,subsequent layers in a convolutional neural networks matches the different features of the input image to form a meaningful blocks of data.