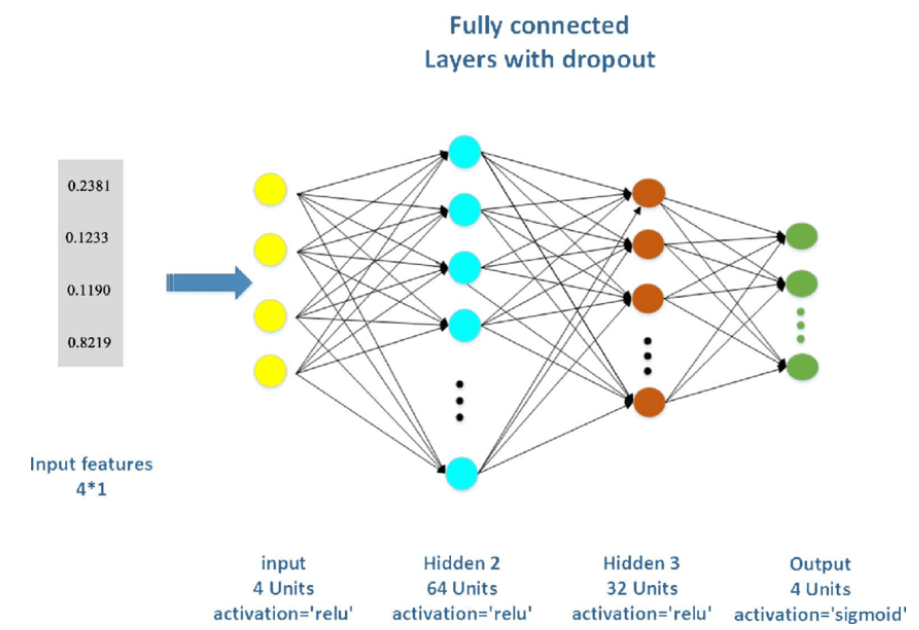
Different Types of Hidden layer

Hidden layers in Artificial Neural Networks (ANNs) are the intermediate layers between the input and output layers. They play a crucial role in learning and extracting features from the input data. Here are some common types of hidden layers used in ANNs:

**1. Dense (Fully Connected) Layer**

Each neuron in this layer is connected to every neuron in the previous layer. This is the most basic type of layer and is used extensively in feedforward neural networks.

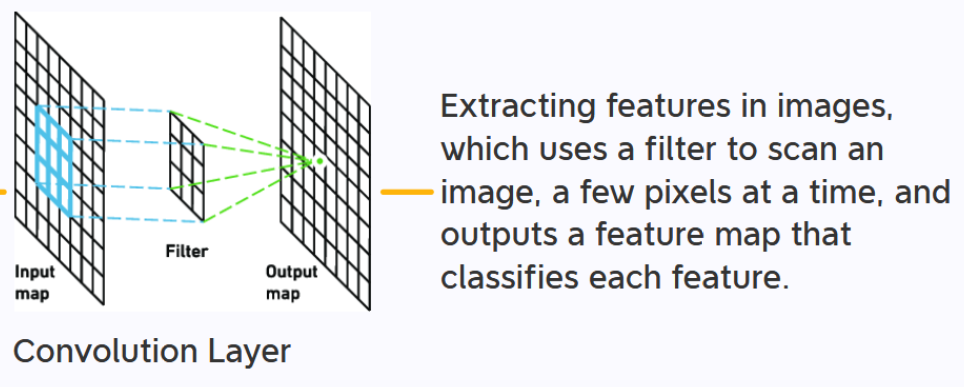
Input Layer -> Dense Layer -> Output Layer



### 2. Convolutional Layer

Commonly used in Convolutional Neural Networks (CNNs) for image and video processing. It applies convolution operations to the input, capturing spatial hierarchies in the data.

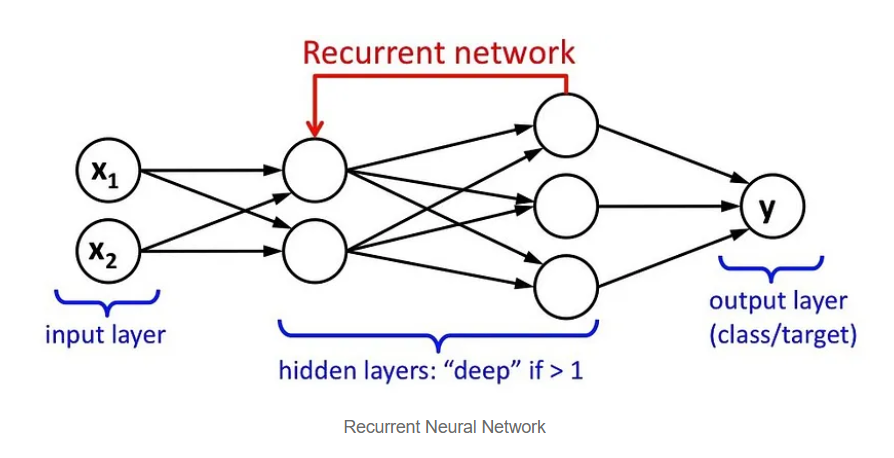
Input Layer -> Convolutional Layer -> Pooling Layer -> Dense Layer -> Output Layer



### 3. Recurrent Layer

Used in Recurrent Neural Networks (RNNs) to process sequential data like time series or natural language. The layer maintains a 'memory' by looping over time steps.

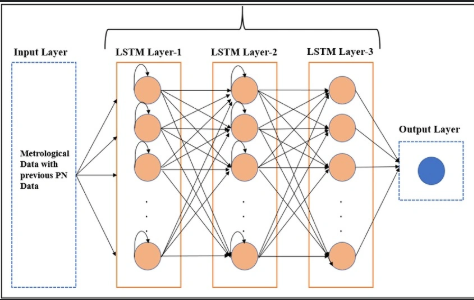
Input Layer -> Recurrent Layer (with time steps) -> Dense Layer -> Output Layer



### 4. Long Short-Term Memory (LSTM) Layer

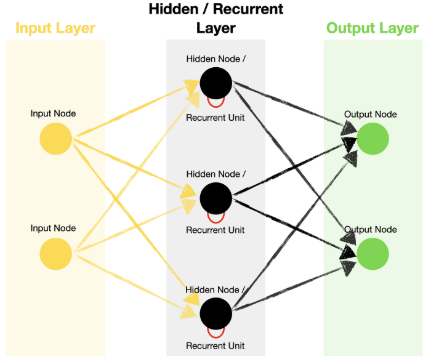
A special kind of recurrent layer that can capture long-term dependencies in sequential data. It’s designed to avoid the vanishing gradient problem in RNNs.

Input Layer -> LSTM Layer -> Dense Layer -> Output Layer



### 5. Gated Recurrent Unit (GRU) Layer

Similar to LSTM but with a simpler architecture and fewer parameters, making it computationally more efficient.

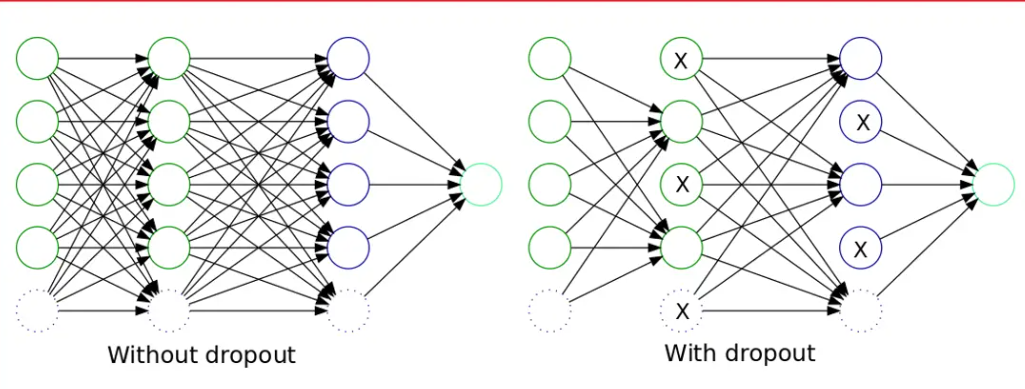


Input Layer -> GRU Layer -> Dense Layer -> Output Layer

### 6. Dropout Layer

Used to prevent overfitting by randomly setting a fraction of input units to zero at each update during training time.

Input Layer -> Dropout Layer -> Dense Layer -> Output Layer



### 7. Batch Normalization Layer

Normalizes the input of each mini-batch, helping to speed up training and make the model more stable.

Input Layer -> Batch Normalization Layer -> Dense Layer -> Output Layer

