# Lecture\_6b

Hello and welcome to the second lecture in this week. In this video, we're going to look at a recap of neural networks. We are going to look at the common variants of neural networks.

Again, neural networks are very popular for implementing machine learning and conventionally, a neural network can be defined as a model of reasoning based on the human brain. A typical neural network will comprise of a group of nodes or processors, also called artificial neurons, which are analogous to the biological neurons in the brain and the neurons is an a typical neural network are connected by weighted links which pass signals from one neuron to another in a way that makes a typical neural network a universal function approximate. The perceptron is the simplest form of a neural network and it primarily consists of a single neuron with adjustable synaptic weights

and a hard limiter. A multiplayer perceptron is a feed-forward neural network with one or more hidden layers as illustrated. Typically it consists of an input layer, at least one hidden layer and an output layer. In a feed-forward neural network, particularly the multilayer perceptron, the input signals are propagated in a forward direction on a layer-by-layer basis because the errors in the output layer in ta multilayer perceptron or multilayer neural network are always propagated back to the input layer. They are also called back-propagation neural network and the learning algorithm in a typical back-propagation neural network has two phases.

First, a training input pattern is presented to the network input layer and the network propagates the input pattern from a layer to layer until the output pattern is generated by the output layer.

And if this pattern is different from the desired output, an error is calculated and then propagated backwards through the network from the output layer to the input layer as I mentioned earlier on. The weights are then modified as the error has been propagated from the output layer to the hidden layer and then on to the input layer. Variants of neural networks. There are several ways to depict the different types of neural networks that are available today. Some of the possible ways include, but definitely are not limited to: the mode of operation or the modus operandi of the neural network, the training algorithm employed by the neural network, the architecture or the structure of the neural network and several other ways. And as a result, we have many variants or types of neural networks available today as illustrated here. For our intents and purposes, we are going to be focussing on the popular variants of neural networks, which include the feed-forward neural network, already introduced in the previous lecture video.

Recurrent neural network and the convolutional neural network. Looking at a feed-forward neural network in more details. In a typical feed-forward neural network, all nodes are fully connected and the information flows from the input layer to the output layer without back loops as illustrated here, and it's also good to know that the connections between the nodes do not form a cycle. The information flow, at the risk of repetition is unidirectional, that is it is just in one direction and there is at least one layer between the input and output. In other words, there is at least one hidden layer. Feed-forward neuron networks are mostly trained using backpropagation method as discussed in the previous lecture video. To implement supervised learning, that is classification and/or regression using the feed-forward neural networks, suitable activation functions must be selected or decided by the data analyst. It's good to know that if the activation function of the last unit in the feed forward neural network is a linear function, then the feed-forward neural network is a regression model. However, if the activation function of the last unit in the feed-forward neural network is a logistic function, then the feed-forward network is a classification model. Other choices of activation functions for feed-forward neural networks may include the hyperbolic tangent and the rectified linear unit. The hyperbolic tangent activation function is quite similar to the logistic sigmoid function because of the sigmoidal shape, that is s shaped and the range is from negative one to positive one as illustrated. Negative inputs to the function are mapped strongly negative. Zero inputs to the function are mapped near zero and the function is monotonic. However, the derivative is not monotonic. By being monotonic, it indicates that the function is either entirely non-increasing, as you can see here, or entirely non-decreasing, as seen here and it is mainly used for binary classification. Rectified linear unit activation function arguably is the most widely used activation function in neural networks. If it's input is negative, then it's output is equal to zero. Otherwise, the output is equal to the input. The ranges from zero to up until infinity.

The function and its derivative are both monotonic. Recurrent neural networks are derived from feed-forward neural network. However, it's good to know that recurrent neural networks are not feed-forward neural networks because recurrent neural networks have loops. In typical recurrent neural networks, the connections between the nodes form a directed graph along temporal sequence as seen here on a directive graph where we have a set of vertices connected my edges where the edges have a direction associated with them. Conventionally, recurring neural networks allow previous outputs to be used as inputs while having hidden state. Recurrent neural networks are mainly used to classify or generate sequences. By definition, a sequences is a data array or matrix in which each row is a feature vector. Recurrent neural networks label sequences by predicting a class or category for the entire sequence. Recurrent neural networks work by generating a sequence by outputting another sequence which is related

(the length may be different) to the input sequence.

Recurrent neural networks are mainly employed in the fields of natural language processing and speech recognition. Intuitively, because sentences, text and speeches are naturally occurring sequences. each unit in each recurrent layer in a typical recurrent neural network takes two inputs: the first outputs from the previous layer and the second stage from previous time step from the same layer where the unit resides. To train recurrent neural network models, The backpropagation through time (BPTT) algorithm is employed. Conceptually, backpropagation through time algorithm works by unrolling all input timesteps of the recurrent neural network. Summarily, the backpropagation through time algorithm as follows: In the first step, a sequence or pattern of timesteps of input-output are presented to the recurrent neural network. In the second step, the network is unrolled and errors are evaluated by comparing predicted output to the expected output across each timestep. In the third step the network is rolled-up and weights are updated to minimise errors.

The fourth step, the algorithm iterates by going back to step two and the entire process is repeated until a certain criteria is met. Training the recurrent neural network using the backpropagation through time algorithm can become computationally expensive as the number of timesteps increases. This is so because the number of derivatives required per single weight update is equal to the number of timesteps per input sequence or pattern. Large number of timesteps can cause weights to vanish or explode, that is go to zero or overflow and cause learning to be very slow

and the model to become noisy. In practise, gated units are used in recurrent neural networks to store information in the units of the recurrent neural network for future use. The stored information is usually taken from an early timestep in the input sequence and it is then employed to process features near the end of the input sequence. A very common gated recurrent neural network is the long short term memory (LSTM). Convolutional neural networks are a class of deep learning neural networks. By definition, deep learning simply refers to training neural networks with more than two non-output layers.

As previously explained, when neural networks grow larger due to an increased number of layers, they tend to experience vanishing gradient or exploding gradient. Given the context to vanishing gradient or exploding gradient,

we need to look at how backpropagation walks again. Backpropagation is quite efficient in updating the values of parameters in neural networks. It works by deriving the partial derivatives of a complex cost function. The partial derivatives are then used to proportionally update the neural network's parameters in each iteration. In some cases during gradient descent which is a first-order iterative optimisation used to find the optimal parameters, the gradients become so small (vanishingly) that some parameters do not change their values or even worse, the neural network may be prevented from training any further. This phenomena is generally referred to as varnishing gradient. Modern algorithmic frameworks, mathematical tools and activation functions such as the rectified linear unit work by circumvent the problem of varnishing gradient. As deep learning generally refers to training

neural networks using modern algorithmic frameworks and mathematical tools regardless of how deep the neural network case that is, regardless of how many layers the neural network has.

Convolutional neural networks are a special type of feed-forward neural networks that significantly reduce the number of parameters in a deep network without diminishing the quality of the model. A typical convolutional neural network will have the following layers: an input layer, the convolutional layers, the rectified linear unit layers, the pooling layers, the fully connected layer and the output layer. Convolutional rectified linear unit, pooling and fully connected layers in a typical convolutional neural network usually constitute the hidden layers. Taking a closer look at the convolutional neural network and what exactly happens in each layer. In the convolutional layers, convolutional operation is applied to the input and the information is passed on to the next layer. In the pooling layers, we have combination of the outputs of clusters of neurons into a single neuron in the next layer.

In the fully connected layers, every neuron in one layer is connected to every neuron in the next layer. And the main distinction between the convolutional layers and the fully connected layers

is this: in a convolutional layer, neurons only receive input from a subarea of the previous layer, whereas in a fully connected layer each neuron receives inputs from every element of the previous layer. Convolutional neural networks are mainly used for image processing and classification. For any given image processing or classification problem, a typical convolutional neural network works by extracting features from its inputs. As a result, there is no need for manual feature extraction. The features are also not trained. Rather, they are learnt while the network trains on a set of input images. And this makes deep learning models extremely accurate for computer vision tasks. In convolutional neural networks, feature detection is learnt through several hidden layers and each layer increases the complexity of the learnt features.

Summarily, a conventional convolutional neural network works as follows for an image classification or processing problem: it takes an image as an input. It applies many different filters to the image to generate a feature map. It then applies a rectified linear unit to activation function to enhance non-linearity. It then applies a pooling layer to each feature map. It flattens the pooled images to a single long vector. It inputs the vector into a fully connected artificial neural network. Processes the features through the artificial neural network and the final fully connected layer categorises or classifies the image. Typically, the training convolutional neural networks goes through forward propagation and backpropagation over several epochs. That is, it iterates until there is a well-defined neural network which trained weights and the feature detectors.

In this video we've looked at a recap of neural networks and we've also discussed some common variants of the neural networks, particularly feed-forward neural networks, recurrent neural networks and convolutional neural networks.