**Basic Interview Questions**

**1. What is the difference between Machine Learning and Deep Learning?**

[**Machine Learning**](https://intellipaat.com/blog/what-is-machine-learning/) forms a subset of Artificial Intelligence, where we use statistics and algorithms to train machines with data, thereby, helping them improve with experience.

[**Deep Learning**](https://intellipaat.com/blog/what-is-deep-learning/) is a part of Machine Learning, which involves mimicking the human brain in terms of structures called neurons, thereby, forming [**neural networks**](https://intellipaat.com/blog/tutorial/machine-learning-tutorial/neural-network-tutorial/).

**2. What is a perceptron?**

A perceptron is similar to the actual neuron in the human brain. It receives inputs from various entities and applies functions to these inputs, which transform them to be the output.

A perceptron is mainly used to perform binary classification where it sees an input, computes functions based on the weights of the input, and outputs the required transformation.

**3. How is Deep Learning better than Machine Learning?**

Machine Learning is powerful in a way that it is sufficient to solve most of the problems. However, [**Deep Learning**](https://intellipaat.com/blog/deep-learning-tutorial-deep-learning/) gets an upper hand when it comes to working with data that has a large number of dimensions. With data that is large in size, a Deep Learning model can easily work with it as it is built to handle this.

**4. What are some of the most used applications of Deep Learning?**

Deep Learning is used in a variety of fields today. The most used ones are as follows:

* Sentiment Analysis
* Computer Vision
* Automatic Text Generation
* Object Detection
* [Natural Language Processing](https://intellipaat.com/blog/what-is-natural-language-processing/)
* Image Recognition

**5. What is the meaning of overfitting?**

Overfitting is a very common issue when working with Deep Learning. It is a scenario where the Deep Learning algorithm vigorously hunts through the data to obtain some valid information.

This makes the Deep Learning model pick up noise rather than useful data, causing very high variance and low bias. This makes the model less accurate, and this is an undesirable effect that can be prevented.

**6. What are activation functions?**

Activation functions are entities in Deep Learning that are used to translate inputs into a usable output parameter. It is a function that decides if a neuron needs activation or not by calculating the weighted sum on it with the bias.

Using an activation function makes the model output to be non-linear. There are many types of activation functions:

* ReLU
* Softmax
* Sigmoid
* Linear
* Tanh

Bottom of Form

**7. Why is Fourier transform used in Deep Learning?**

Fourier transform is an effective package used for analyzing and managing large amounts of data present in a database. It can take in real-time array data and process it quickly. This ensures that high efficiency is maintained and also makes the model more open to processing a variety of signals.

**8. What are the steps involved in training a perception in Deep Learning?**

There are five main steps that determine the learning of a perceptron:

1. Initialize thresholds and weights
2. Provide inputs
3. Calculate outputs
4. Update weights in each step
5. Repeat steps 2 to 4

**9. What is the use of the loss function?**

The loss function is used as a measure of accuracy to see if a neural network has learned accurately from the training data or not. This is done by comparing the training dataset to the testing dataset.

The loss function is a primary measure of the performance of the neural network. In Deep Learning, a good performing network will have a low loss function at all times when training.

**10. What are some of the Deep Learning frameworks or tools that you have used?**

This question is quite common in a Deep Learning interview. Make sure to answer based on the experience you have with the tools.

However, some of the top Deep Learning frameworks out there today are:

* TensorFlow
* Keras
* PyTorch
* Caffe2
* CNTK
* MXNet
* Theano

**11. What is the use of the swish function?**

The swish function is a self-gated activation function developed by Google. It is now a popular activation function used by many as Google claims that it outperforms all of the other activation functions in terms of computational efficiency.

**12. What are autoencoders?**

Autoencoders are artificial neural networks that learn without any supervision. Here, these networks have the ability to automatically learn by mapping the inputs to the corresponding outputs.

Autoencoders, as the name suggests, consist of two entities:

* Encoder: Used to fit the input into an internal computation state
* Decoder: Used to convert the computational state back into the output

**13. What are the steps to be followed to use the gradient descent algorithm?**

There are five main steps that are used to initialize and use the gradient descent algorithm:

* Initialize biases and weights for the network
* Send input data through the network (the input layer)
* Calculate the difference (the error) between expected and predicted values
* Change values in neurons to minimize the loss function
* Multiple iterations to determine the best weights for efficient working

**14. Differentiate between a single-layer perceptron and a multi-layer perceptron.**

|  |  |
| --- | --- |
| **Single-layer Perceptron** | **Multi-layer Perceptron** |
| Cannot classify non-linear data points | Can classify non-linear data |
| Takes in a limited amount of parameters | Withstands a lot of parameters |
| Less efficient with large data | Highly efficient with large datasets |

**15. What is data normalization in Deep Learning?**

Data normalization is a preprocessing step that is used to refit the data into a specific range. This ensures that the network can learn effectively as it has better convergence when performing backpropagation.

**16. What is forward propagation?**

Forward propagation is the scenario where inputs are passed to the hidden layer with weights. In every single hidden layer, the output of the activation function is calculated until the next layer can be processed. It is called forward propagation as the process begins from the input layer and moves toward the final output layer.

**17. What is backpropagation?**

[**Backpropagation**](https://intellipaat.com/blog/tutorial/artificial-intelligence-tutorial/back-propagation-algorithm/) is used to minimize the cost function by first seeing how the value changes when weights and biases are tweaked in the neural network. This change is easily calculated by understanding the gradient at every hidden layer. It is called backpropagation as the process begins from the output layer, moving backward to the input layers.

**18. What are hyperparameters in Deep Learning?**

Hyperparameters are variables used to determine the structure of a neural network. They are also used to understand parameters, such as the learning rate and the number of hidden layers, and more, present in the neural network.

**19. How can hyperparameters be trained in neural networks?**

Hyperparameters can be trained using four components as shown below:

* Batch size: This is used to denote the size of the input chunk. Batch sizes can be varied and cut into sub-batches based on the requirement.
* Epochs: An epoch denotes the number of times the training data is visible to the neural network so that it can train. Since the process is iterative, the number of epochs will vary based on the data.
* Momentum: Momentum is used to understand the next consecutive steps that occur with the current data being executed at hand. It is used to avoid oscillations when training.
* Learning rate: Learning rate is used as a parameter to denote the time required for the network to update the parameters and learn.

**Intermediate Interview Questions**

**20. What is the meaning of dropout in Deep Learning?**

Dropout is a technique that is used to avoid overfitting a model in Deep Learning. If the dropout value is too low, then it will have minimal effect on learning. If it is too high, then the model can under-learn, thereby, causing lower efficiency.

**21. What are tensors?**

Tensors are multidimensional arrays in Deep Learning that are used to represent data. They represent the data with higher dimensions. Due to the high-level nature of the programming languages, the syntax of tensors is easily understood and broadly used.

**22. What is the meaning of model capacity in Deep Learning?**

In Deep Learning, model capacity refers to the capacity of the model to take in a variety of mapping functions. Higher model capacity means a large amount of information can be stored in the network.

**23. What is a Boltzmann machine?**

A Boltzmann machine is a type of recurrent neural network that uses binary decisions, alongside biases, to function. These neural networks can be hooked up together to create deep belief networks, which are very sophisticated and used to solve the most complex problems out there.

**24. What are some of the advantages of using TensorFlow?**

TensorFlow has numerous advantages, and some of them are as follows:

* High amount of flexibility and platform independence
* Trains using CPU and GPU
* Supports auto differentiation and its features
* Handles threads and asynchronous computation easily
* Open-source
* Has a large community

**25. What is a computational graph in Deep Learning?**

A computation graph is a series of operations that are performed to take inputs and arrange them as nodes in a graph structure. It can be considered as a way of implementing mathematical calculations into a graph. This helps in parallel processing and provides high performance in terms of computational capability.

**26. What is a CNN?**

CNNs are [convolutional neural networks](https://intellipaat.com/blog/tutorial/artificial-intelligence-tutorial/convolution-neural-network/) that are used to perform analysis on [image annotation](https://intellipaat.com/blog/image-annotation/) and visuals. These classes of neural networks can input a multi-channel image and work on it easily.

**27. What are the various layers present in a CNN?**

There are four main layers that form a convolutional neural network:

* Convolution: These are layers consisting of entities called filters that are used as parameters to train the network.
* ReLu: It is used as the activation function and is always used with the convolution layer.
* Pooling: Pooling is the concept of shrinking the complex data entities that form after convolution and is primarily used to maintain the size of an image after shrinkage.
* Connectedness: This is used to ensure that all of the layers in the neural network are fully connected and activation can be computed using the bias easily.

**28. What is an RNN in Deep Learning?**

RNNs stand for [recurrent neural networks](https://intellipaat.com/blog/tutorial/artificial-intelligence-tutorial/recurrent-neural-network/), which form to be a popular type of artificial neural network. They are used to process sequences of data, text, genomes, handwriting, and more. RNNs make use of backpropagation for the training requirements.

**29. What is a vanishing gradient when using RNNs?**

Vanishing gradient is a scenario that occurs when we use RNNs. Since RNNs make use of backpropagation, gradients at every step of the way will tend to get smaller as the network traverses through backward iterations. This equates to the model learning very slowly, thereby, causing efficiency problems in the network.

**30. What is exploding gradient descent in Deep Learning?**

Exploding gradients are an issue causing a scenario that clumps up the gradients. This creates a large number of updates of the weights in the model when training.

The working of gradient descent is based on the condition that the updates are small and controlled. Controlling the updates will directly affect the efficiency of the model.

**31. What is the use of LSTM?**

LSTM stands for long short-term memory. It is a type of RNN that is used to sequence a string of data. It consists of feedback chains that give it the ability to perform like a general-purpose computational entity.

**32. Where are autoencoders used?**

Autoencoders have a wide variety of usage in the real world. The following are some of the popular ones:

* Adding color to black–white images
* Removing noise from images
* Dimensionality reduction
* Feature removal and variation

**33. What are the types of autoencoders?**

There are four main types of autoencoders:

* Deep autoencoders
* Convolutional autoencoders
* Sparse autoencoders
* Contractive autoencoders

**34. What is a Restricted Boltzmann Machine?**

A Restricted Boltzmann Machine, or RBM for short, is an undirected graphical model that is popularly used in Deep Learning today. It is an algorithm that is used to perform:

* Dimensionality reduction
* Regression
* Classification
* Collaborative filtering
* Topic modeling

Next up on this top Deep Learning interview questions and answers blog, let us take a look at the advanced questions.

**Advanced Interview Questions**

**35. What are some of the limitations of Deep Learning?**

There are a few disadvantages of Deep Learning as mentioned below:

* Networks in Deep Learning require a huge amount of data to train well.
* Deep Learning concepts can be complex to implement sometimes.
* Achieving a high amount of model efficiency is difficult in many cases.

**36. What are the variants of gradient descent?**

There are three variants of gradient descent as shown below:

* Stochastic gradient descent: A single training example is used for the calculation of gradient and for updating parameters.
* Batch gradient descent: Gradient is calculated for the entire dataset, and parameters are updated at every iteration.
* Mini-batch gradient descent: Samples are broken down into smaller-sized batches and then worked on as in the case of stochastic gradient descent.

**37. Why is mini-batch gradient descent so popular?**

Mini-batch gradient descent is popular as:

* It is more efficient when compared to stochastic gradient descent.
* Generalization is done by finding the flat minima.
* It helps avoid the local minima by allowing the approximation of the gradient for the entire dataset.

**38. What are deep autoencoders?**

Deep autoencoders are an extension of the regular autoencoders. Here, the first layer is responsible for the first-order function execution of the input. The second layer will take care of the second-order functions, and it goes on.

Usually, a deep autoencoder is a combination of two or more symmetrical deep-belief networks where:

* The first five shallow layers consist of the encoding part
* The other layers take care of the decoding part

On the next set of Deep Learning questions, let us look further into the topic.

**39. Why is the Leaky ReLU function used in Deep Learning?**

Leaky ReLU, also called LReL, is used to manage a function to allow the passing of small-sized negative values if the input value to the network is less than zero.

**40. What are some of the examples of supervised learning algorithms in Deep Learning?**

There are three main supervised learning algorithms in Deep Learning:

* Artificial neural networks
* Convolutional neural networks
* Recurrent neural networks

**41. What are some of the examples of unsupervised learning algorithms in Deep Learning?**

There are three main unsupervised learning algorithms in Deep Learning:

* Autoencoders
* Boltzmann machines
* Self-organizing maps

Next up, let us look at  more neural network interview questions that will help you ace the interviews.

**42. Can we initialize the weights of a network to start from zero?**

Yes, it is possible to begin with zero initialization. However, it is not recommended to use because setting up the weights to zero initially will cause all of the neurons to produce the same output and the same gradients when performing backpropagation. This means that the network will not have the ability to learn at all due to the absence of asymmetry between each of the neurons.

**43. What is the meaning of valid padding and same padding in CNN?**

* Valid padding: It is used when there is no requirement for padding. The output matrix will have the dimensions (n – f + 1) X (n – f + 1) after convolution.
* Same padding: Here, padding elements are added all around the output matrix. It will have the same dimensions as the input matrix.

**44. What are some of the applications of transfer learning in Deep Learning?**

Transfer learning is a scenario where a large model is trained on a dataset with a large amount of data and this model is used on simpler datasets, thereby resulting in extremely efficient and accurate neural networks.

The popular examples of transfer learning are in the case of:

* BERT
* ResNet
* GPT-2
* VGG-16

**45. How is the transformer architecture better than RNNs in Deep Learning?**

With the use of sequential processing, programmers were up against:

* The usage of high processing power
* The difficulty of parallel execution

This caused the rise of the transformer architecture. Here, there is a mechanism called attention mechanism, which is used to map all of the dependencies between sentences, thereby making huge progress in the case of NLP models.

**46. What are the steps involved in the working of an LSTM network?**

There are three main steps involved in the working of an LSTM network:

* The network picks up the information that it has to remember and identifies what to forget.
* Cell state values are updated based on Step 1.
* The network calculates and analyzes which part of the current state should make it to the output.

**47. What are the elements in TensorFlow that are programmable?**

In TensorFlow, users can program three elements:

* Constants
* Variables
* Placeholders

**48. What is the meaning of bagging and boosting in Deep Learning?**

Bagging is the concept of splitting a dataset and randomly placing it into bags for training the model.

Boosting is the scenario where incorrect data points are used to force the model to produce the wrong output. This is used to retrain the model and increase accuracy.

**49. What are generative adversarial networks (GANs)?**

Generative adversarial networks are used to achieve generative modeling in Deep Learning. It is an unsupervised task that involves the discovery of patterns in the input data to generate the output.

The generator is used to generate new examples, while the discriminator is used to classify the examples generated by the generator.

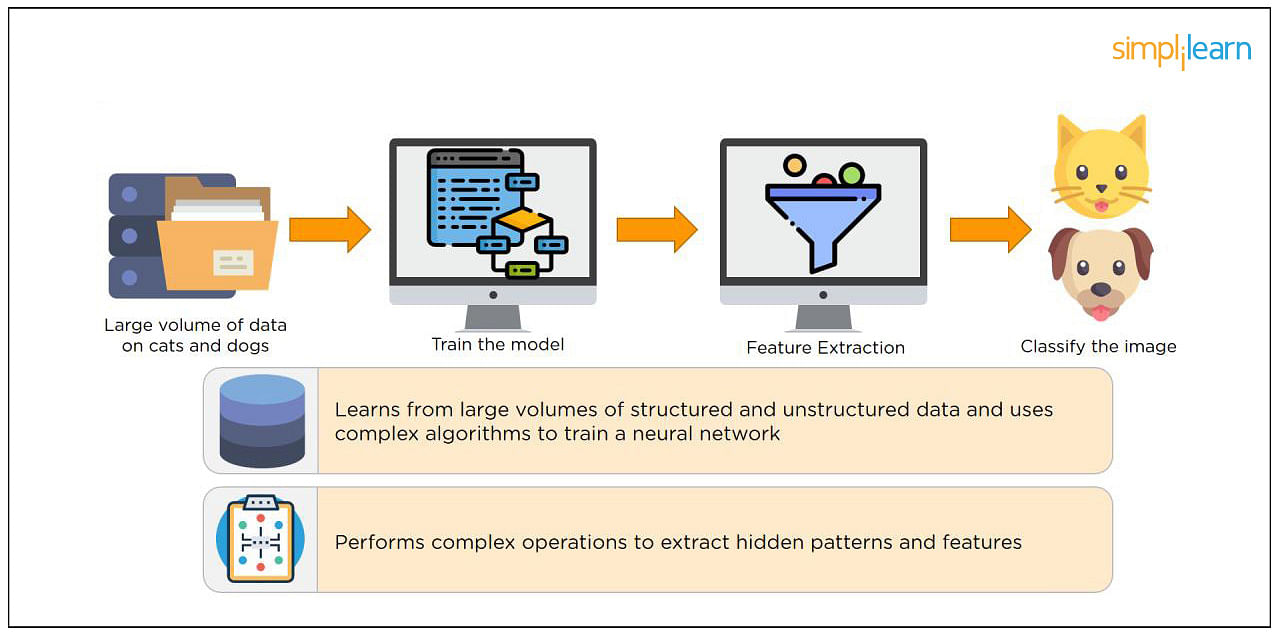
**50. Why are generative adversarial networks (GANs) so popular?**

Generative adversarial networks are used for a variety of purposes. In the case of working with images, they have a high amount of traction and efficient working.

* Creation of art: GANs are used to create artistic images, sketches, and paintings.
* Image enhancement: They are used to greatly enhance the resolution of the input images.
* Image translation: They are also used to change certain aspects, such as day to night and summer to winter, in images easily.

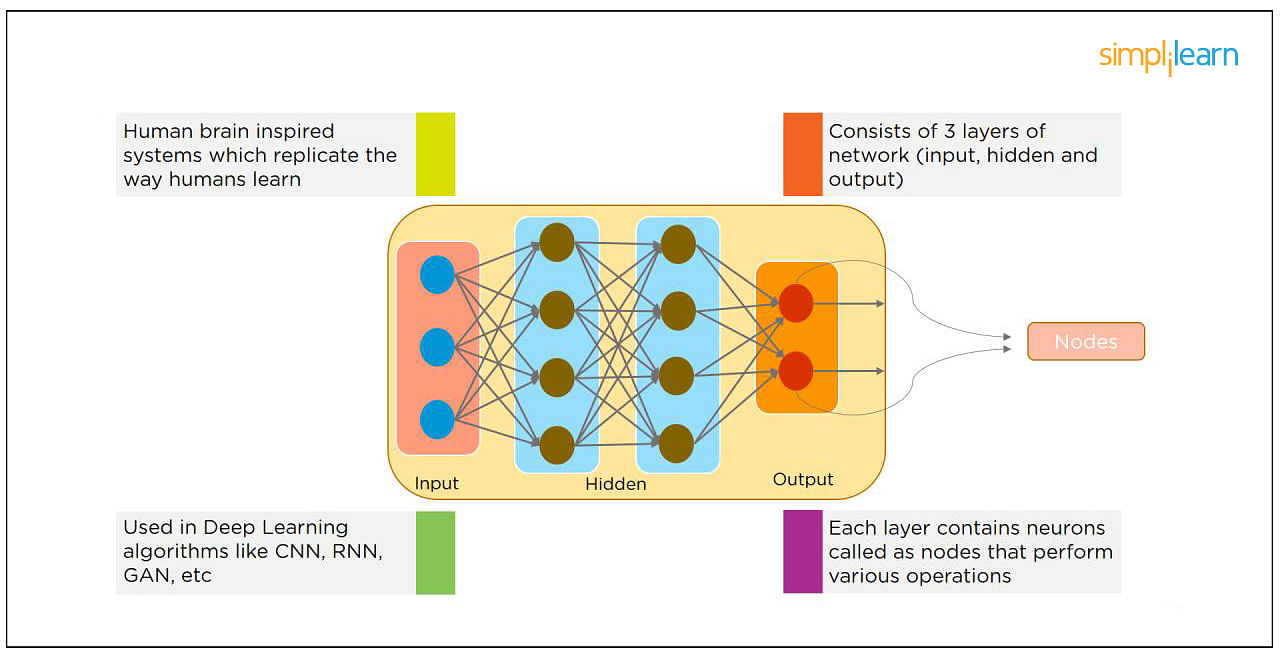
### 1. What is Deep Learning?

If you are going for a deep learning interview, you definitely know what exactly deep learning is. However, with this question the interviewee expects you to give an in-detail answer, with an example. [Deep Learning](https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-deep-learning) involves taking large volumes of structured or unstructured data and using complex algorithms to train neural networks. It performs complex operations to extract hidden patterns and features (for instance, distinguishing the image of a cat from that of a dog).



### 2. What is a Neural Network?

[Neural Networks](https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-neural-network) replicate the way humans learn, inspired by how the neurons in our brains fire, only much simpler.



The most common Neural Networks consist of three network layers:

1. An input layer
2. A hidden layer (this is the most important layer where feature extraction takes place, and adjustments are made to train faster and function better)
3. An output layer

Each sheet contains neurons called “nodes,” performing various operations. Neural Networks are used in [deep learning algorithms](https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm) like CNN, RNN, GAN, etc.

### 3. What Is a Multi-layer Perceptron(MLP)?

As in Neural Networks, [MLPs](https://www.simplilearn.com/tutorials/deep-learning-tutorial/multilayer-perceptron) have an input layer, a hidden layer, and an output layer. It has the same structure as a single layer [perceptron](https://www.simplilearn.com/tutorials/deep-learning-tutorial/perceptron) with one or more hidden layers. A single layer perceptron can classify only linear separable classes with binary output (0,1), but MLP can classify nonlinear classes.

Except for the input layer, each node in the other layers uses a nonlinear activation function. This means the input layers, the data coming in, and the activation function is based upon all nodes and weights being added together, producing the output. MLP uses a[supervised learning](https://www.simplilearn.com/tutorials/machine-learning-tutorial/supervised-and-unsupervised-learning) method called “backpropagation.” In backpropagation, the neural network calculates the error with the help of cost function. It propagates this error backward from where it came (adjusts the weights to train the model more accurately).

### 4. What Is Data Normalization, and Why Do We Need It?

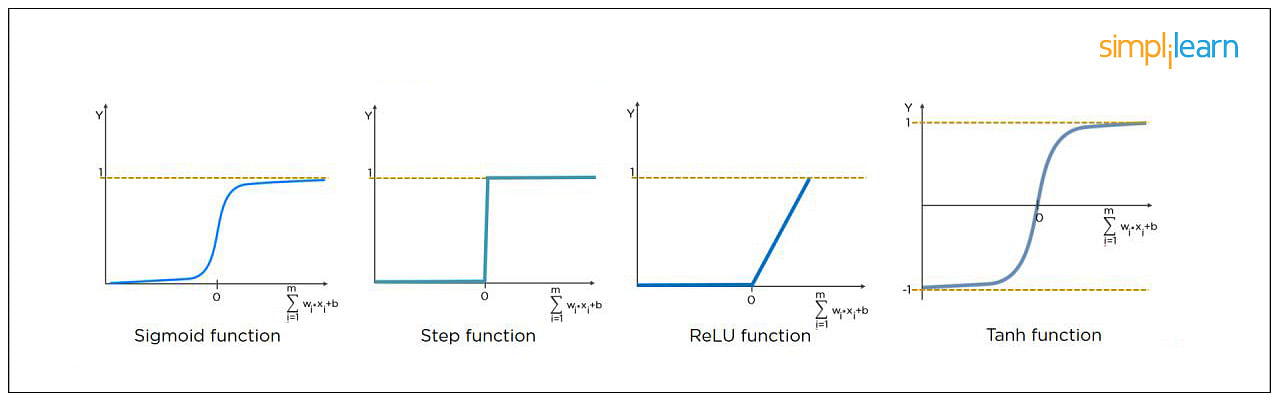
The process of standardizing and reforming data is called “Data Normalization.” It’s a pre-processing step to eliminate data redundancy. Often, data comes in, and you get the same information in different formats. In these cases, you should rescale values to fit into a particular range, achieving better convergence.

### 5. What is the Boltzmann Machine?

One of the most basic Deep Learning models is a Boltzmann Machine, resembling a simplified version of the Multi-Layer Perceptron. This model features a visible input layer and a hidden layer -- just a two-layer neural net that makes stochastic decisions as to whether a neuron should be on or off. Nodes are connected across layers, but no two nodes of the same layer are connected.

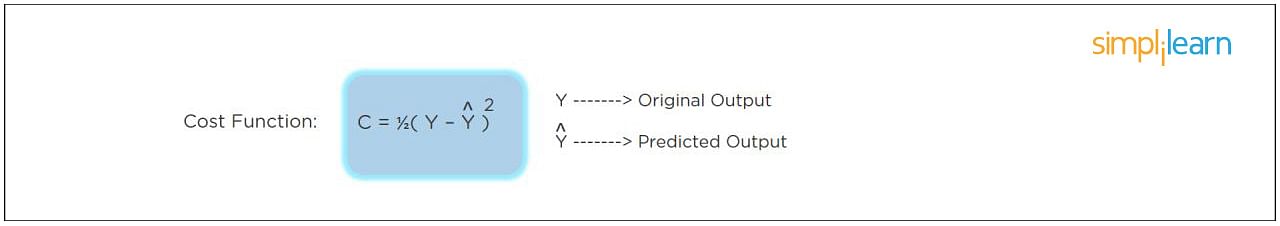
### 6. What Is the Role of Activation Functions in a Neural Network?

At the most basic level, an activation function decides whether a neuron should be fired or not. It accepts the weighted sum of the inputs and bias as input to any activation function. Step function, Sigmoid, ReLU, Tanh, and Softmax are examples of activation functions.



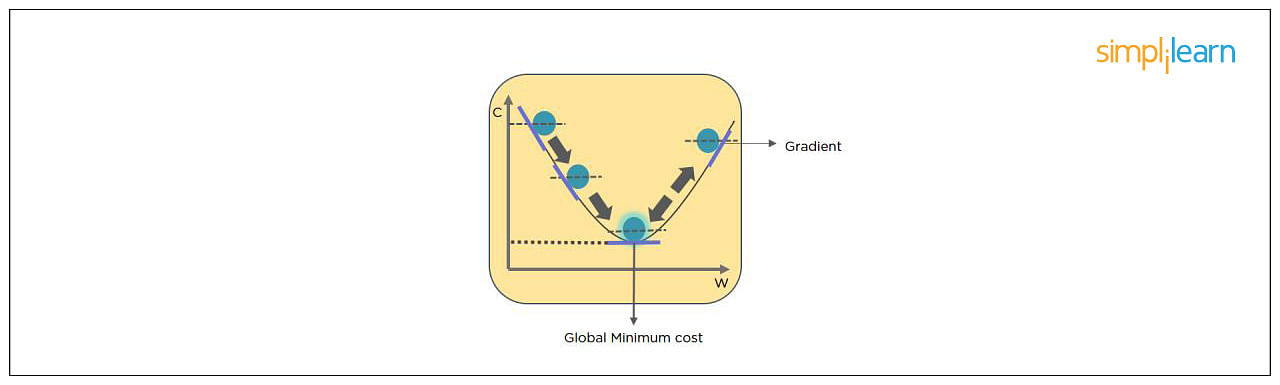
### 7. What Is the Cost Function?

Also referred to as “loss” or “error,” cost function is a measure to evaluate how good your model’s performance is. It’s used to compute the error of the output layer during backpropagation. We push that error backward through the neural network and use that during the different training functions.



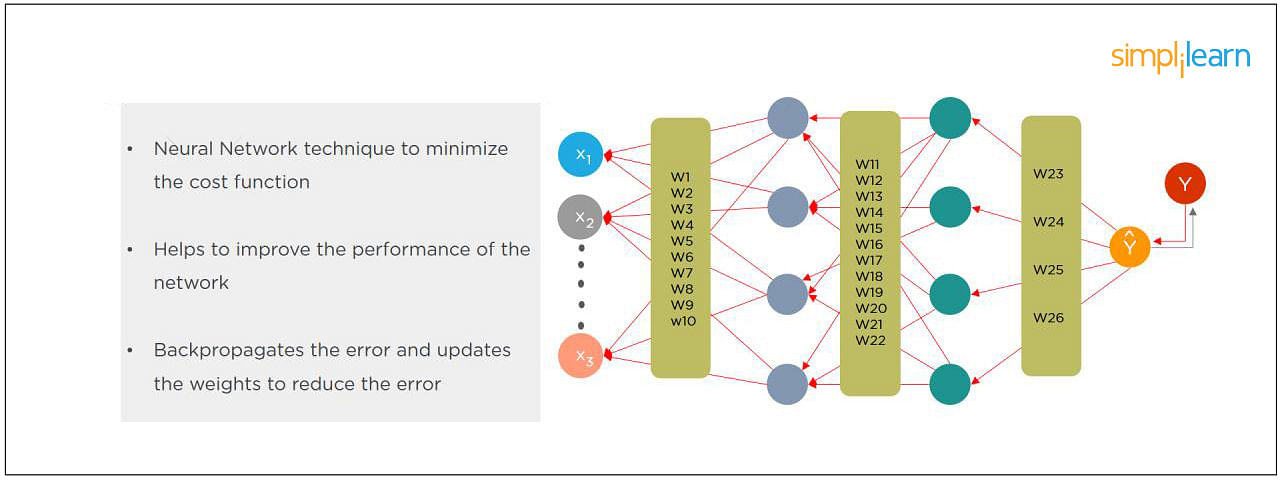
### 8. What Is Gradient Descent?

Gradient Descent is an optimal algorithm to minimize the cost function or to minimize an error. The aim is to find the local-global minima of a function. This determines the direction the model should take to reduce the error.



### 9. What Do You Understand by Backpropagation?

This is one of the most frequently asked deep learning interview questions. Backpropagation is a technique to improve the performance of the network. It backpropagates the error and updates the weights to reduce the error.

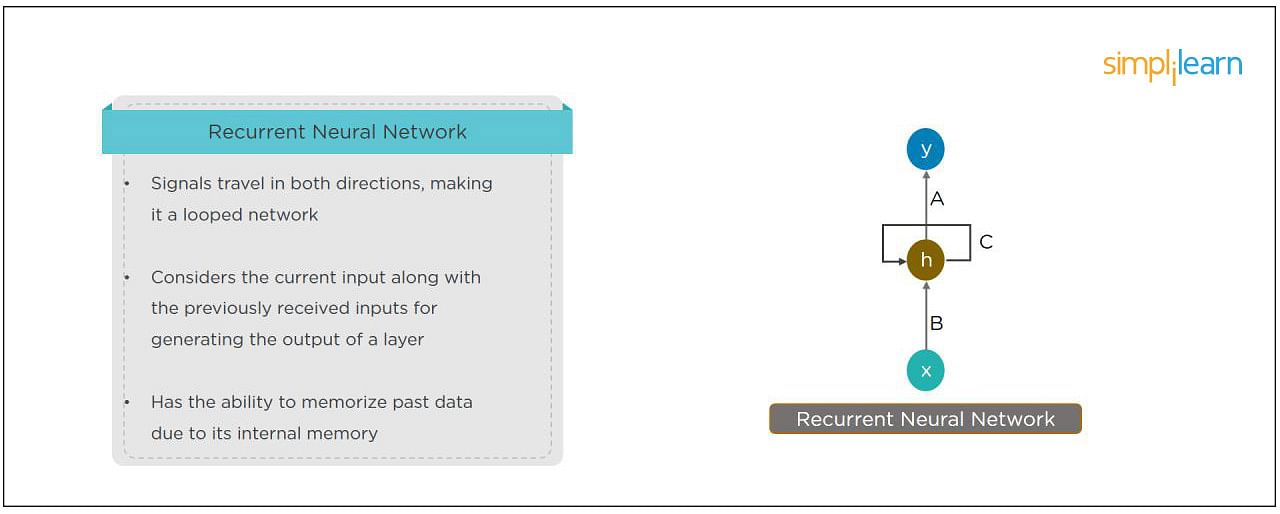


### 10. What Is the Difference Between a Feedforward Neural Network and Recurrent Neural Network?

In this deep learning interview question, the interviewee expects you to give a detailed answer.

A Feedforward Neural Network signals travel in one direction from input to output. There are no feedback loops; the network considers only the current input. It cannot memorize previous inputs (e.g., [CNN](https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network)).

A Recurrent Neural Network’s signals travel in both directions, creating a looped network. It considers the current input with the previously received inputs for generating the output of a layer and can memorize past data due to its internal memory.

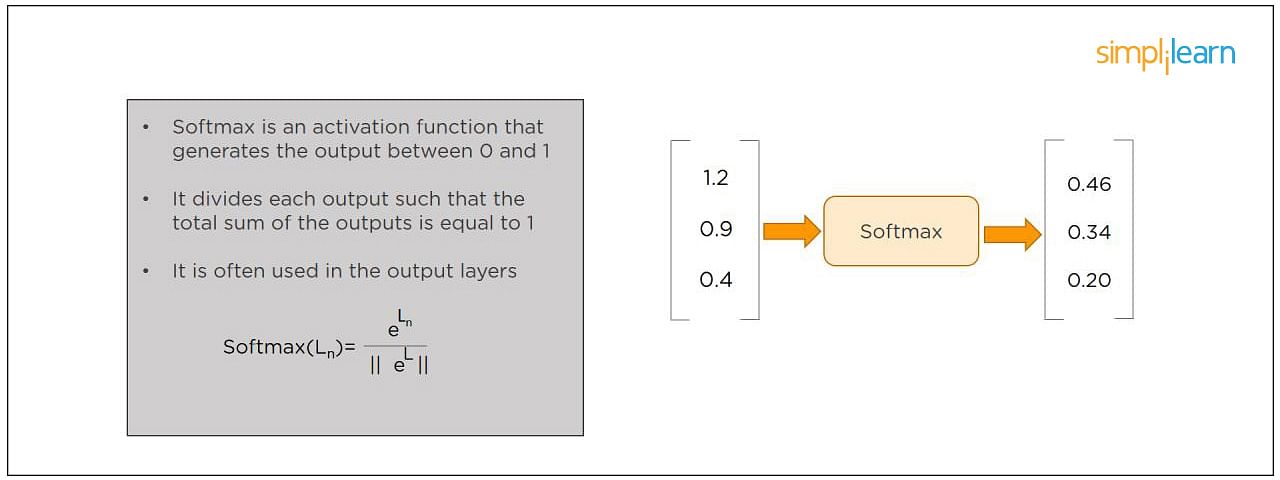


### 11. What Are the Applications of a Recurrent Neural Network (RNN)?

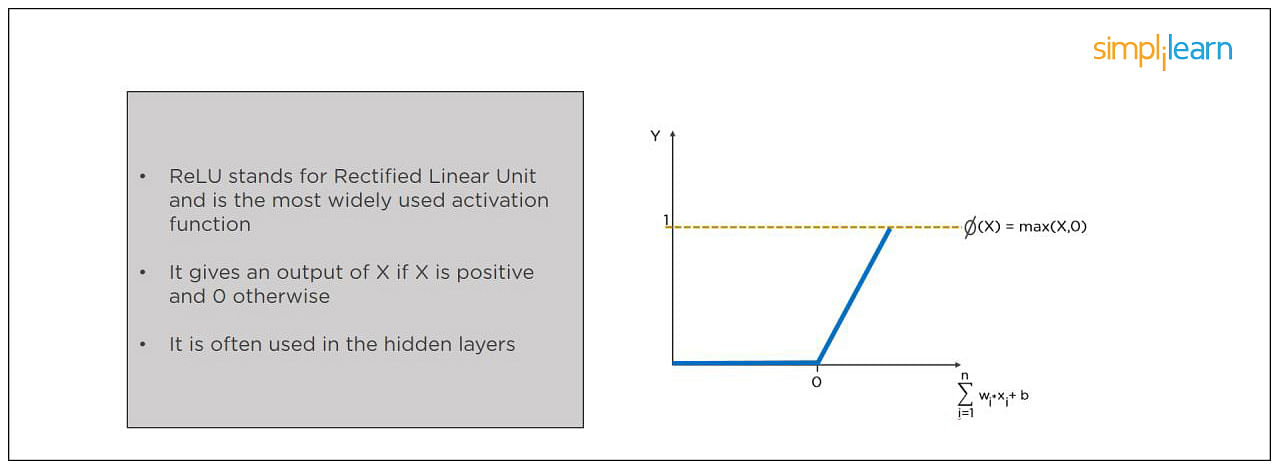
The [RNN](https://www.simplilearn.com/tutorials/deep-learning-tutorial/rnn) can be used for sentiment analysis, text mining, and image captioning. Recurrent Neural Networks can also address time series problems such as predicting the prices of stocks in a month or quarter.

### 12. What Are the Softmax and ReLU Functions?

Softmax is an activation function that generates the output between zero and one. It divides each output, such that the total sum of the outputs is equal to one. Softmax is often used for output layers.

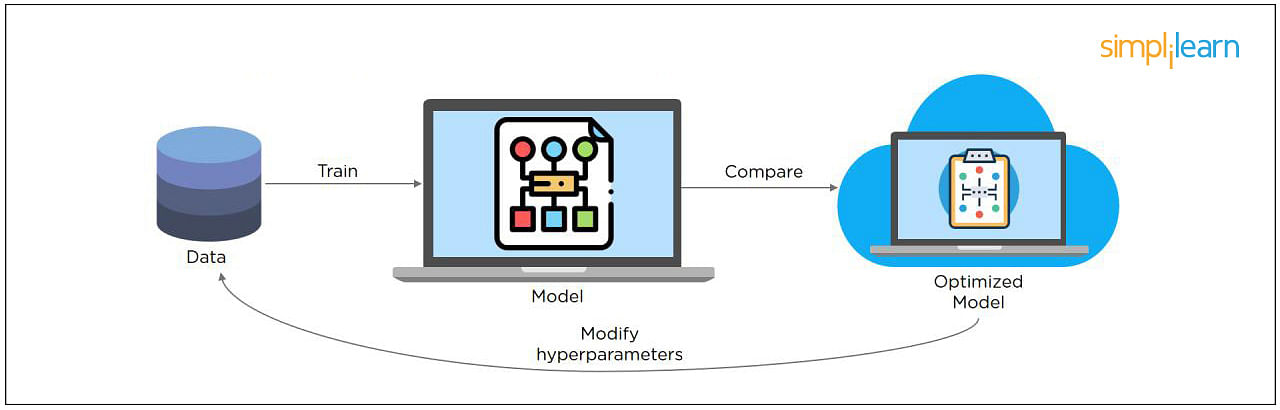


ReLU (or Rectified Linear Unit) is the most widely used activation function. It gives an output of X if X is positive and zeros otherwise. ReLU is often used for hidden layers.



### 13. What Are Hyperparameters?

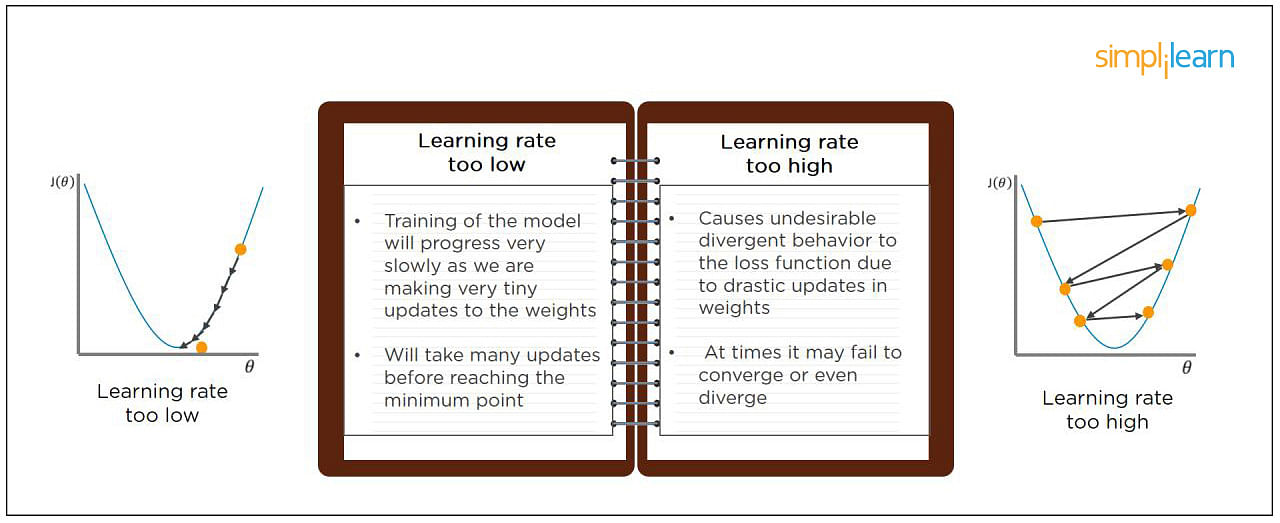
This is another frequently asked deep learning interview question. With neural networks, you’re usually working with hyperparameters once the data is formatted correctly. A hyperparameter is a parameter whose value is set before the learning process begins. It determines how a network is trained and the structure of the network (such as the number of hidden units, the learning rate, epochs, etc.).



### 14. What Will Happen If the Learning Rate Is Set Too Low or Too High?

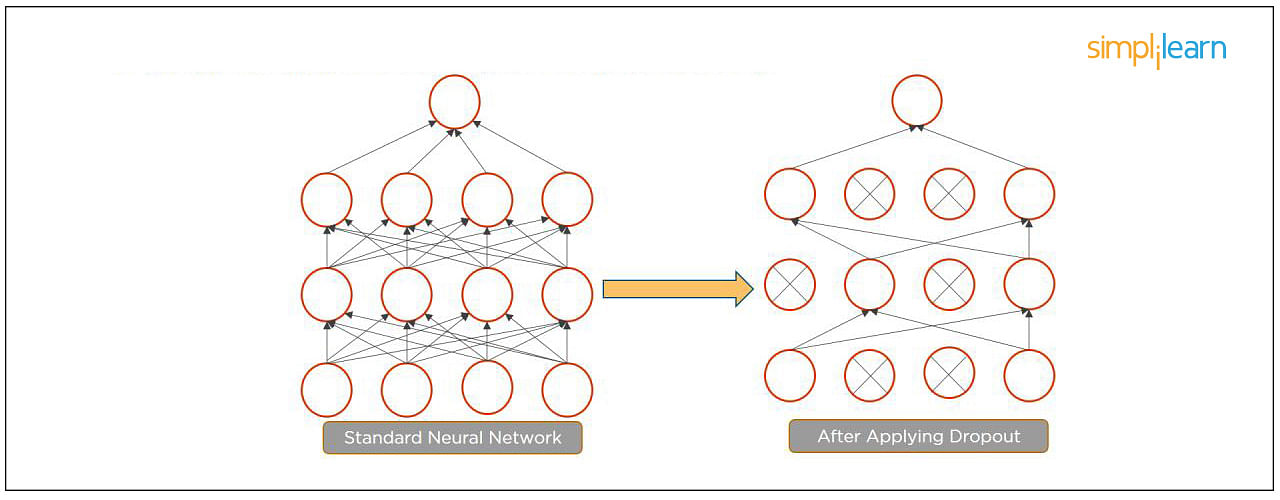
When your learning rate is too low, training of the model will progress very slowly as we are making minimal updates to the weights. It will take many updates before reaching the minimum point.

If the learning rate is set too high, this causes undesirable divergent behavior to the loss function due to drastic updates in weights. It may fail to converge (model can give a good output) or even diverge (data is too chaotic for the network to train).



### 15. What Is Dropout and Batch Normalization?

Dropout is a technique of dropping out hidden and visible units of a network randomly to prevent overfitting of data (typically dropping 20 percent of the nodes). It doubles the number of iterations needed to converge the network.



Batch normalization is the technique to improve the performance and stability of neural networks by normalizing the inputs in every layer so that they have mean output activation of zero and standard deviation of one.

The next step on this top Deep Learning interview questions and answers blog will be to discuss intermediate questions.

### 16. What Is the Difference Between Batch Gradient Descent and Stochastic Gradient Descent?

|  |  |
| --- | --- |
| Batch Gradient Descent | Stochastic Gradient Descent |
| The batch gradient computes the gradient using the entire dataset.  It takes time to converge because the volume of data is huge, and weights update slowly. | The stochastic gradient computes the gradient using a single sample.  It converges much faster than the batch gradient because it updates weight more frequently. |

### 17. What is Overfitting and Underfitting, and How to Combat Them?

Overfitting occurs when the model learns the details and noise in the training data to the degree that it adversely impacts the execution of the model on new information. It is more likely to occur with nonlinear models that have more flexibility when learning a target function. An example would be if a model is looking at cars and trucks, but only recognizes trucks that have a specific box shape. It might not be able to notice a flatbed truck because there's only a particular kind of truck it saw in training. The model performs well on training data, but not in the real world.

Underfitting alludes to a model that is neither well-trained on data nor can generalize to new information. This usually happens when there is less and incorrect data to train a model. Underfitting has both poor performance and accuracy.

To combat overfitting and underfitting, you can resample the data to estimate the model accuracy (k-fold cross-validation) and by having a validation dataset to evaluate the model.

### 18. How Are Weights Initialized in a Network?

There are two methods here: we can either initialize the weights to zero or assign them randomly.

Initializing all weights to 0: This makes your model similar to a linear model. All the neurons and every layer perform the same operation, giving the same output and making the deep net useless.

Initializing all weights randomly: Here, the weights are assigned randomly by initializing them very close to 0. It gives better accuracy to the model since every neuron performs different computations. This is the most commonly used method.

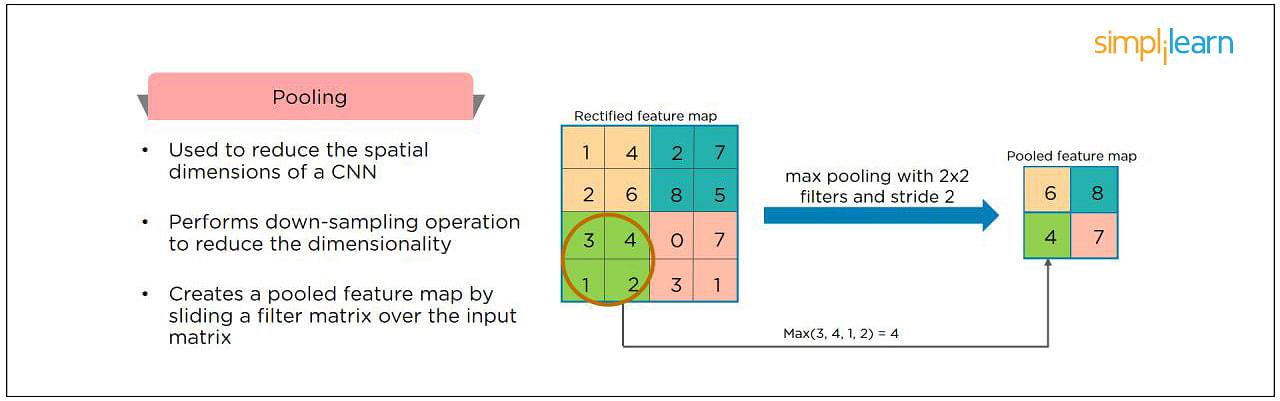
### 19. What Are the Different Layers on CNN?

There are four layers in CNN:

1. Convolutional Layer -  the layer that performs a convolutional operation, creating several smaller picture windows to go over the data.
2. ReLU Layer - it brings non-linearity to the network and converts all the negative pixels to zero. The output is a rectified feature map.
3. Pooling Layer - pooling is a down-sampling operation that reduces the dimensionality of the feature map.
4. Fully Connected Layer - this layer recognizes and classifies the objects in the image.

### 20. What is Pooling on CNN, and How Does It Work?

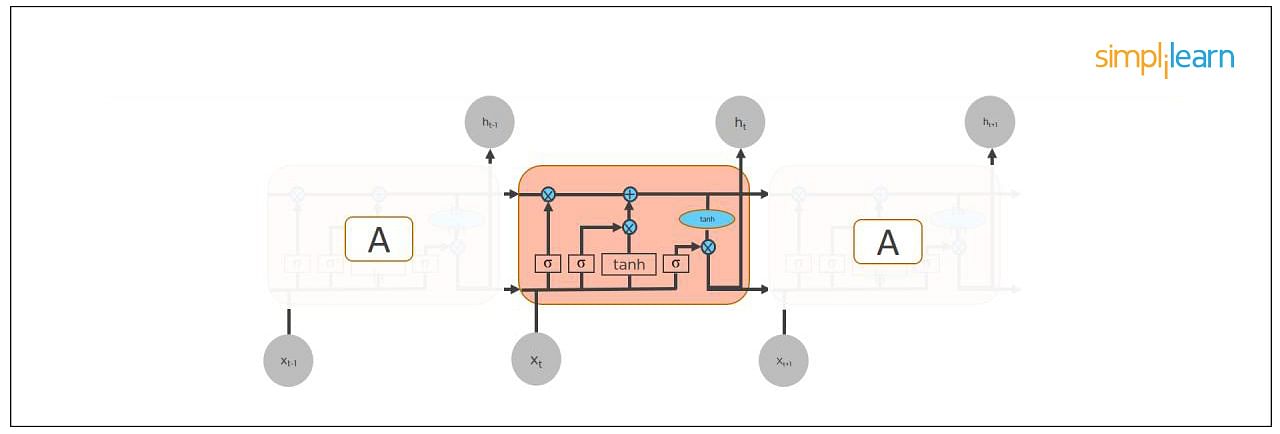
Pooling is used to reduce the spatial dimensions of a CNN. It performs down-sampling operations to reduce the dimensionality and creates a pooled feature map by sliding a filter matrix over the input matrix.



### 21. How Does an LSTM Network Work?

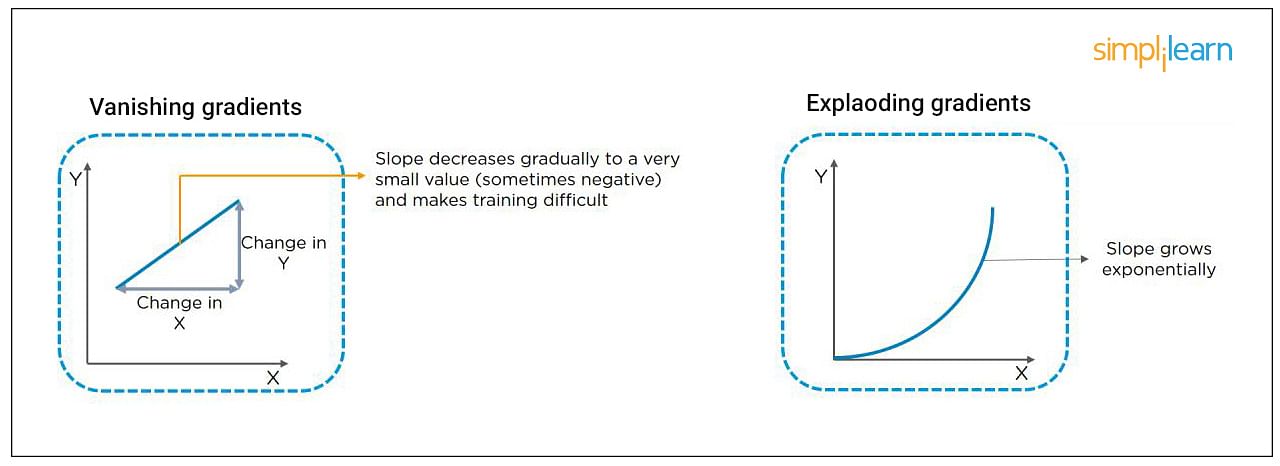
Long-Short-Term Memory (LSTM) is a special kind of recurrent neural network capable of learning long-term dependencies, remembering information for long periods as its default behavior. There are three steps in an LSTM network:

* Step 1: The network decides what to forget and what to remember.
* Step 2: It selectively updates cell state values.
* Step 3: The network decides what part of the current state makes it to the output.



### 22. What Are Vanishing and Exploding Gradients?

While training an RNN, your slope can become either too small or too large; this makes the training difficult. When the slope is too small, the problem is known as a “Vanishing Gradient.” When the slope tends to grow exponentially instead of decaying, it’s referred to as an “Exploding Gradient.” Gradient problems lead to long training times, poor performance, and low accuracy.



### 23. What Is the Difference Between Epoch, Batch, and Iteration in Deep Learning?

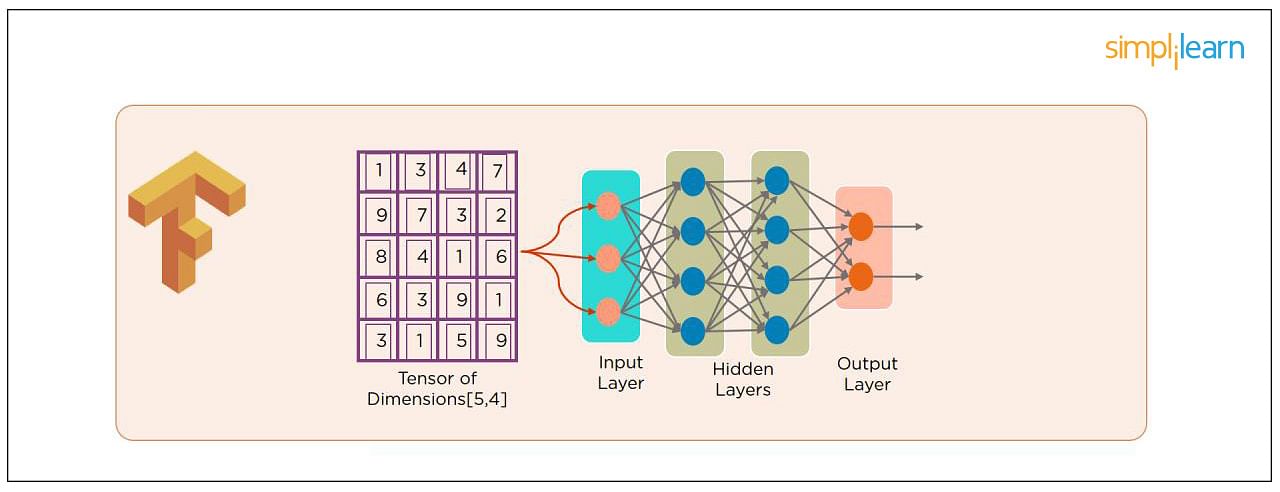
* Epoch - Represents one iteration over the entire dataset (everything put into the training model).
* Batch - Refers to when we cannot pass the entire dataset into the neural network at once, so we divide the dataset into several batches.
* Iteration - if we have 10,000 images as data and a batch size of 200. then an epoch should run 50 iterations (10,000 divided by 50).

### 24. Why is Tensorflow the Most Preferred Library in Deep Learning?

[Tensorflow](https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-tensorflow) provides both [C++](https://www.simplilearn.com/tutorials/cpp-tutorial/learn-cpp-basics) and Python APIs, making it easier to work on and has a faster compilation time compared to other Deep Learning libraries like [Keras and Torch](https://www.simplilearn.com/keras-vs-tensorflow-vs-pytorch-article" \o "Keras and Torch" \t "_blank). Tensorflow supports both CPU and GPU computing devices.

### 25. What Do You Mean by Tensor in Tensorflow?

This is another most frequently asked deep learning interview question. A tensor is a mathematical object represented as arrays of higher dimensions. These arrays of data with different dimensions and ranks fed as input to the neural network are called “Tensors.”



### 26. What Are the Programming Elements in Tensorflow?

Constants - Constants are parameters whose value does not change. To define a constant we use  tf.constant() command. For example:

a = tf.constant(2.0,tf.float32)

b = tf.constant(3.0)

Print(a, b)

Variables - Variables allow us to add new trainable parameters to graph. To define a variable, we use the tf.Variable() command and initialize them before running the graph in a session. An example:

W = tf.Variable([.3].dtype=tf.float32)

b = tf.Variable([-.3].dtype=tf.float32)

Placeholders - these allow us to feed data to a tensorflow model from outside a model. It permits a value to be assigned later. To define a placeholder, we use the tf.placeholder() command. An example:

a = tf.placeholder (tf.float32)

b = a\*2

with tf.Session() as sess:

result = sess.run(b,feed\_dict={a:3.0})

print result

Sessions - a session is run to evaluate the nodes. This is called the “Tensorflow runtime.” For example:

a = tf.constant(2.0)

b = tf.constant(4.0)

c = a+b

# Launch Session

Sess = tf.Session()

# Evaluate the tensor c

print(sess.run(c))

### 27. Explain a Computational Graph.

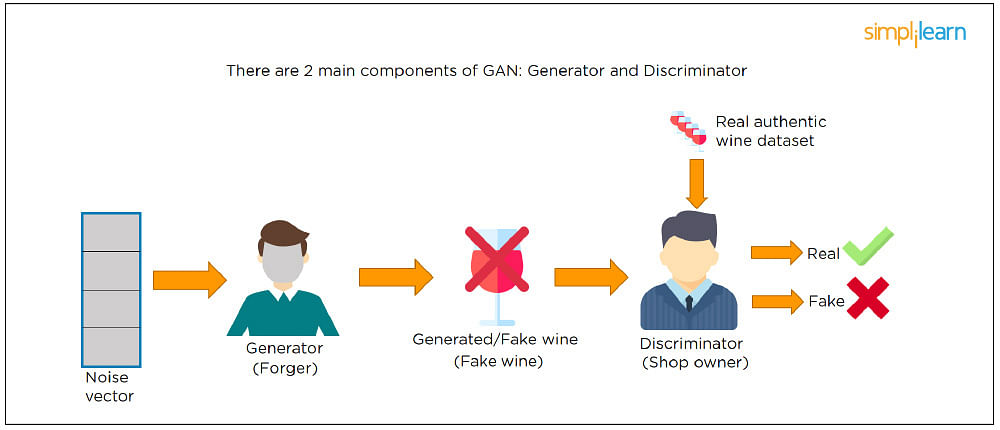
Everything in a [tensorflow](https://www.simplilearn.com/tutorials/deep-learning-tutorial/tensorflow" \o "tensorflow" \t "_blank) is based on creating a computational graph. It has a network of nodes where each node operates, Nodes represent mathematical operations, and edges represent tensors. Since data flows in the form of a graph, it is also called a “DataFlow Graph.”

### 28. Explain [Generative Adversarial Network.](https://www.simplilearn.com/tutorials/deep-learning-tutorial/generative-adversarial-networks-gans)

Suppose there is a wine shop purchasing wine from dealers, which they resell later. But some dealers sell fake wine. In this case, the shop owner should be able to distinguish between fake and authentic wine.

The forger will try different techniques to sell fake wine and make sure specific techniques go past the shop owner’s check. The shop owner would probably get some feedback from wine experts that some of the wine is not original. The owner would have to improve how he determines whether a wine is fake or authentic.

The forger’s goal is to create wines that are indistinguishable from the authentic ones while the shop owner intends to tell if the wine is real or not accurately.



Let us understand this example with the help of an image shown above.

There is a noise vector coming into the forger who is generating fake wine.

Here the forger acts as a Generator.

The shop owner acts as a Discriminator.

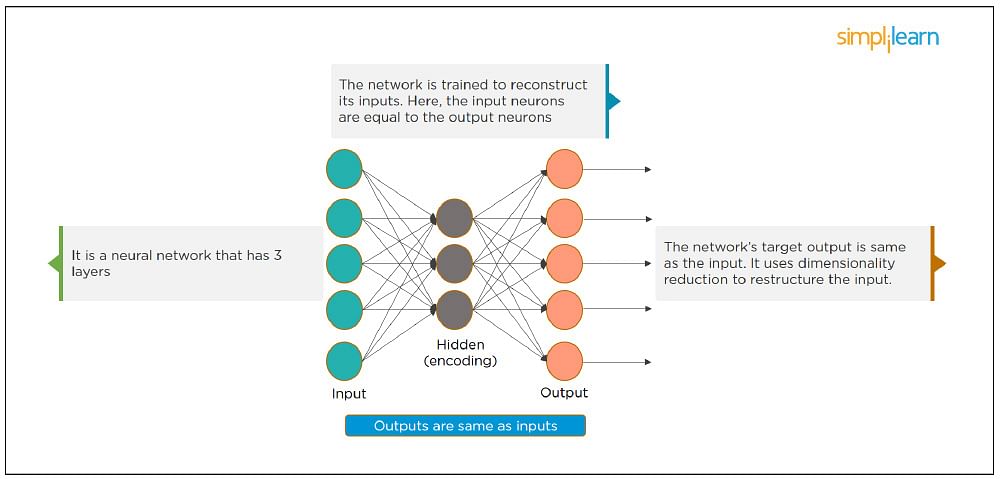
The Discriminator gets two inputs; one is the fake wine, while the other is the real authentic wine. The shop owner has to figure out whether it is real or fake.

So, there are two primary components of Generative Adversarial Network (GAN) named:

1. Generator
2. Discriminator

The generator is a CNN that keeps keys producing images and is closer in appearance to the real images while the discriminator tries to determine the difference between real and fake images The ultimate aim is to make the discriminator learn to identify real and fake images.

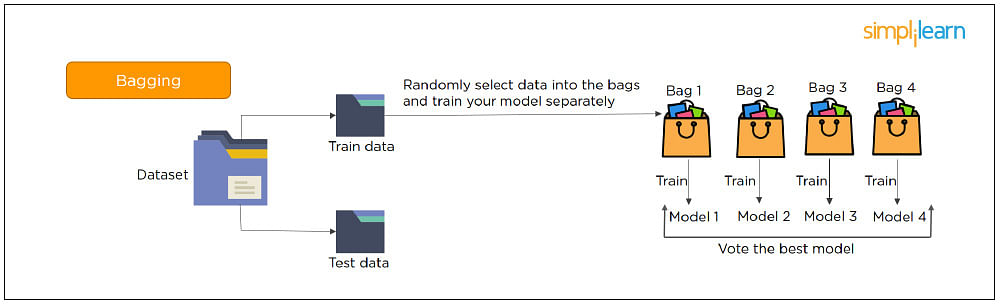
### 29. What Is an Auto-encoder?



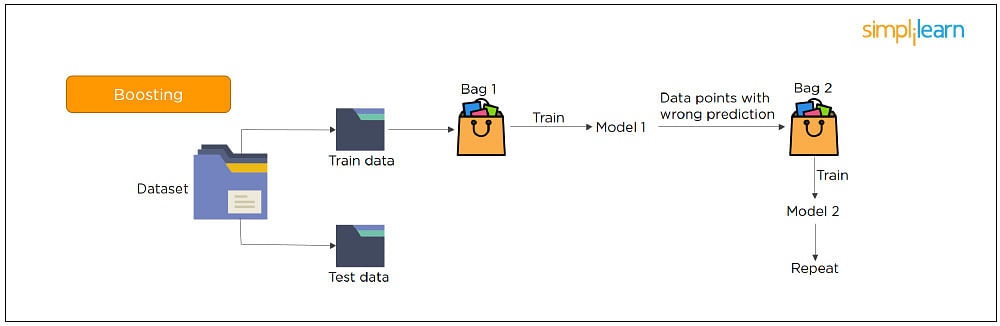
This Neural Network has three layers in which the input neurons are equal to the output neurons. The network's target outside is the same as the input. It uses dimensionality reduction to restructure the input. It works by compressing the image input to a latent space representation then reconstructing the output from this representation.

### 30. What Is Bagging and Boosting?

Bagging and Boosting are ensemble techniques to train multiple models using the same learning algorithm and then taking a call.



With Bagging, we take a dataset and split it into training data and test data. Then we randomly select data to place into the bags and train the model separately.



With Boosting, the emphasis is on selecting data points which give wrong output to improve the accuracy.

The following are some of the most important advanced deep learning interview questions that you should know!

### 31. What is the significance of using the Fourier transform in Deep Learning tasks?

The Fourier transform function efficiently analyzes, maintains, and manages large datasets. You can use it to generate real-time array data that is helpful for processing multiple signals.

### 32. What do you understand by transfer learning? Name a few commonly used transfer learning models.

Transfer learning is the process of transferring the learning from a model to another model without having to train it from scratch. It takes critical parts of a pre-trained model and applies them to solve new but similar machine learning problems.

Some of the popular transfer learning models are:

* VGG-16
* BERT
* GTP-3
* Inception V3
* XCeption

### 33. What is the difference between SAME and VALID padding in Tensorflow?

Using the Tensorflow library, tf.nn.max\_pool performs the max-pooling operation. Tf.nn.max\_pool has a padding argument that takes 2 values - SAME or VALID.

With padding == “SAME” ensures that the filter is applied to all the elements of the input.

The input image gets fully covered by the filter and specified stride. The padding type is named SAME as the output size is the same as the input size (when stride=1).

With padding == “VALID” implies there is no padding in the input image. The filter window always stays inside the input image. It assumes that all the dimensions are valid so that the input image gets fully covered by a filter and the stride defined by you.

### 34. What are some of the uses of Autoencoders in Deep Learning?

* Autoencoders are used to convert black and white images into colored images.
* Autoencoder helps to extract features and hidden patterns in the data.
* It is also used to reduce the dimensionality of data.
* It can also be used to remove noises from images.

### 35. What is the Swish Function?

Swish is an activation function proposed by Google which is an alternative to the ReLU activation function.

It is represented as: f(x) = x \* sigmoid(x).

The Swish function works better than ReLU for a variety of deeper models.

The derivative of Swist can be written as: y’ = y + sigmoid(x) \* (1 - y)

### 36. What are the reasons for mini-batch gradient being so useful?

* Mini-batch gradient is highly efficient compared to stochastic gradient descent.
* It lets you attain generalization by finding the flat minima.
* Mini-batch gradient helps avoid local minima to allow gradient approximation for the whole dataset.

### 37. What do you understand by Leaky ReLU activation function?

Leaky ReLU is an advanced version of the ReLU activation function. In general, the ReLU function defines the gradient to be 0 when all the values of inputs are less than zero. This deactivates the neurons. To overcome this problem, Leaky ReLU activation functions are used. It has a very small slope for negative values instead of a flat slope.

### 38. What is Data Augmentation in Deep Learning?

Data Augmentation is the process of creating new data by enhancing the size and quality of training datasets to ensure better models can be built using them. There are different techniques to augment data such as numerical data augmentation, image augmentation, GAN-based augmentation, and text augmentation.

### 39. Explain the Adam optimization algorithm.

Adaptive Moment Estimation or Adam optimization is an extension to the stochastic gradient descent. This algorithm is useful when working with complex problems involving vast amounts of data or parameters. It needs less memory and is efficient.

Adam optimization algorithm is a combination of two gradient descent methodologies -

Momentum and Root Mean Square Propagation.

### 40. Why is a convolutional neural network preferred over a dense neural network for an image classification task?

* The number of parameters in a convolutional neural network is much more diminutive than that of a Dense Neural Network. Hence, a CNN is less likely to overfit.
* CNN allows you to look at the weights of a filter and visualize what the network learned. So, this gives a better understanding of the model.
* CNN trains models in a hierarchical way, i.e., it learns the patterns by explaining complex patterns using simpler ones.

### 41. Which strategy does not prevent a model from over-fitting to the training data?

1. Dropout
2. Pooling
3. Data augmentation
4. Early stopping

Answer: b) Pooling - It’s a layer in CNN that performs a downsampling operation.

### 42. Explain two ways to deal with the vanishing gradient problem in a deep neural network.

* Use the ReLU activation function instead of the sigmoid function
* Initialize neural networks using Xavier initialization that works with tanh activation.

### 43. Why is a deep neural network better than a shallow neural network?

Both deep and shallow neural networks can approximate the values of a function. But the deep neural network is more efficient as it learns something new in every layer. A shallow neural network has only one hidden layer. But a deep neural network has several hidden layers that create a deeper representation and computation capability.

### 44. What is the need to add randomness in the weight initialization process?

If you set the weights to zero, then every neuron at each layer will produce the same result and the same gradient value during backpropagation. So, the neural network won’t be able to learn the function as there is no asymmetry between the neurons. Hence, randomness to the weight initialization process is crucial.

### 45. How can you train hyperparameters in a neural network?

Hyperparameters in a neural network can be trained using four components:

Batch size: Indicates the size of the input data.

Epochs: Denotes the number of times the training data is visible to the neural network to train.

Momentum: Used to get an idea of the next steps that occur with the data being executed.

Learning rate: Represents the time required for the network to update the parameters and learn.