**1) What is deep learning?**

Deep learning is a part of machine learning with an algorithm inspired by the structure and function of the brain, which is called an **artificial neural network**. In the mid-1960s, **Alexey Grigorevich Ivakhnenko** published the first general, while working on deep learning network. Deep learning is suited over a range of fields such as computer vision, speech recognition, natural language processing, etc.

**2) What are the main differences between AI, Machine Learning, and Deep Learning?**

* AI stands for Artificial Intelligence. It is a technique which enables machines to mimic human behavior.
* Machine Learning is a subset of AI which uses statistical methods to enable machines to improve with experiences.
* Deep learning is a part of Machine learning, which makes the computation of multi-layer neural networks feasible. It takes advantage of neural networks to simulate human-like decision making.

**3) Differentiate supervised and unsupervised deep learning procedures.**

* Supervised learning is a system in which both input and desired output data are provided. Input and output data are labeled to provide a learning basis for future data processing.
* Unsupervised procedure does not need labeling information explicitly, and the operations can be carried out without the same. The common unsupervised learning method is **cluster analysis**. It is used for exploratory data analysis to find hidden patterns or grouping in data.

**4) What are the applications of deep learning?**

There are various applications of deep learning:

* Computer vision
* Natural language processing and pattern recognition
* Image recognition and processing
* Machine translation
* Sentiment analysis
* Question Answering system
* Object Classification and Detection
* Automatic Handwriting Generation
* Automatic Text Generation.

**5) Do you think that deep network is better than a shallow one?**

Both shallow and deep networks are good enough and capable of approximating any function. But for the same level of accuracy, deeper networks can be much more efficient in terms of computation and number of parameters. Deeper networks can create deep representations. At every layer, the network learns a new, more abstract representation of the input.

**6) What do you mean by "overfitting"?**

Overfitting is the most common issue which occurs in deep learning. It usually occurs when a deep learning algorithm apprehends the sound of specific data. It also appears when the particular algorithm is well suitable for the data and shows up when the algorithm or model represents high variance and low bias.

**7) What is Backpropagation?**

Backpropagation is a training algorithm which is used for multilayer neural networks. It transfers the error information from the end of the network to all the weights inside the network. It allows the efficient computation of the gradient.

Backpropagation can be divided into the following steps:

* It can forward propagation of training data through the network to generate output.
* It uses target value and output value to compute error derivative concerning output activations.
* It can backpropagate to compute the derivative of the error concerning output activations in the previous layer and continue for all hidden layers.
* It uses the previously calculated derivatives for output and all hidden layers to calculate the error derivative concerning weights.
* It updates the weights.

**8) What is the function of the Fourier Transform in Deep Learning?**

Fourier transform package is highly efficient for analyzing, maintaining, and managing a large databases. The software is created with a high-quality feature known as the **special portrayal**. One can effectively utilize it to generate real-time array data, which is extremely helpful for processing all categories of signals.

**9) Describe the theory of autonomous form of deep learning in a few words.**

There are several forms and categories available for the particular subject, but the autonomous pattern represents independent or unspecified mathematical bases which are free from any specific categorizer or formula.

**10) What is the use of Deep learning in today's age, and how is it adding data scientists?**

Deep learning has brought significant changes or revolution in the field of machine learning and data science. The concept of a **complex neural network** (CNN) is the main center of attention for data scientists. It is widely taken because of its advantages in performing next-level machine learning operations. The advantages of deep learning also include the process of clarifying and simplifying issues based on an algorithm due to its utmost flexible and adaptable nature. It is one of the rare procedures which allow the movement of data in independent pathways. Most of the data scientists are viewing this particular medium as an advanced additive and extended way to the existing process of machine learning and utilizing the same for solving complex day to day issues.

**11) What are the deep learning frameworks or tools?**

Deep learning frameworks or tools are:

Tensorflow, Keras, Chainer, Pytorch, Theano & Ecosystem, Caffe2, CNTK, DyNetGensim, DSSTNE, Gluon, Paddle, Mxnet, BigDL

**12) What are the disadvantages of deep learning?**

There are some disadvantages of deep learning, which are:

* Deep learning model takes longer time to execute the model. In some cases, it even takes several days to execute a single model depends on complexity.
* The deep learning model is not good for small data sets, and it fails here.

**13) What is the meaning of term weight initialization in neural networks?**

In neural networking, weight initialization is one of the essential factors. A bad weight initialization prevents a network from learning. On the other side, a good weight initialization helps in giving a quicker convergence and a better overall error. Biases can be initialized to zero. The standard rule for setting the weights is to be close to zero without being too small.

**14) Explain Data Normalization.**

Data normalization is an essential preprocessing step, which is used to rescale values to fit in a specific range. It assures better convergence during backpropagation. In general, data normalization boils down to subtracting the mean of each data point and dividing by its standard deviation.

**15) Why is zero initialization not a good weight initialization process**?

If the set of weights in the network is put to a zero, then all the neurons at each layer will start producing the same output and the same gradients during backpropagation.

As a result, the network cannot learn at all because there is no source of asymmetry between neurons. That is the reason why we need to add randomness to the weight initialization process.

16) What are the prerequisites for starting in Deep Learning?

There are some basic requirements for starting in Deep Learning, which are:

* Machine Learning
* Mathematics
* Python Programming

17) What are the supervised learning algorithms in Deep learning?

* Artificial neural network
* Convolution neural network
* Recurrent neural network

18) What are the unsupervised learning algorithms in Deep learning?

* Self Organizing Maps
* Deep belief networks (Boltzmann Machine)
* Auto Encoders

19) How many layers in the neural network?

* **Input Layer**  
  The input layer contains input neurons which send information to the hidden layer.
* **Hidden Layer**  
  The hidden layer is used to send data to the output layer.
* **Output Layer**  
  The data is made available at the output layer.

20) What is the use of the Activation function?

The activation function is used to introduce nonlinearity into the neural network so that it can learn more complex function. Without the Activation function, the neural network would be only able to learn function, which is a linear combination of its input data.

Activation function translates the inputs into outputs. The activation function is responsible for deciding whether a neuron should be activated or not. It makes the decision by calculating the weighted sum and further adding bias with it. The basic purpose of the activation function is to introduce non-linearity into the output of a neuron.

21) How many types of activation function are available?

* Binary Step
* Sigmoid
* Tanh
* ReLU
* Leaky ReLU
* Softmax
* Swish

22) What is a binary step function?

The binary step function is an activation function, which is usually based on a threshold. If the input value is above or below a particular threshold limit, the neuron is activated, then it sends the same signal to the next layer. This function does not allow multi-value outputs.

23) What is the sigmoid function?

The sigmoid activation function is also called the logistic function. It is traditionally a trendy activation function for neural networks. The input data to the function is transformed into a value between **0.0** and **1.0**. Input values that are much larger than 1.0 are transformed to the value 1.0. Similarly, values that are much smaller than 0.0 are transformed into 0.0. The shape of the function for all possible inputs is an S-shape from zero up through 0.5 to 1.0. It was the default activation used on neural networks, in the early 1990s.

24) What is Tanh function?

The hyperbolic tangent function, also known as tanh for short, is a similar shaped nonlinear activation function. It provides output values between **-1.0** and **1.0**. Later in the 1990s and through the 2000s, this function was preferred over the sigmoid activation function as models. It was easier to train and often had better predictive performance.

25) What is ReLU function?

A node or unit which implements the activation function is referred to as a **rectified linear activation unit** or ReLU for short. Generally, networks that use the rectifier function for the hidden layers are referred to as **rectified networks**.

Adoption of ReLU may easily be considered one of the few milestones in the deep learning revolution.

26) What is the use of leaky ReLU function?

The Leaky ReLU (LReLU or LReL) manages the function to allow small negative values when the input is less than zero.

27) What is the softmax function?

The softmax function is used to calculate the probability distribution of the event over 'n' different events. One of the main advantages of using softmax is the output probabilities range. The range will be between 0 to 1, and the sum of all the probabilities will be equal to one. When the softmax function is used for multi-classification model, it returns the probabilities of each class, and the target class will have a high probability.

28) What is a Swish function?

Swish is a new, self-gated activation function. Researchers at Google discovered the Swish function. According to their paper, it performs better than ReLU with a similar level of computational efficiency.

29) What is the most used activation function?

**Relu function** is the most used activation function. It helps us to solve vanishing gradient problems.

30) Can Relu function be used in output layer?

No, Relu function has to be used in hidden layers.

31) In which layer softmax activation function used?

Softmax activation function has to be used in the output layer.

32) What do you understand by Autoencoder?

Autoencoder is an artificial neural network. It can learn representation for a set of data without any supervision. The network automatically learns by copying its input to the output; typically,internet representation consists of smaller dimensions than the input vector. As a result, they can learn efficient ways of representing the data. Autoencoder consists of two parts; an encoder tries to fit the inputs to the internal representation, and a decoder converts the internal state to the outputs.

33) What do you mean by Dropout?

Dropout is a cheap regulation technique used for reducing overfitting in neural networks. We randomly drop out a set of nodes at each training step. As a result, we create a different model for each training case, and all of these models share weights. It's a form of model averaging.

34) What do you understand by Tensors?

Tensors are nothing but a de facto for representing the data in deep learning. They are just multidimensional arrays, which allows us to represent the data having higher dimensions. In general, we deal with high dimensional data sets where dimensions refer to different features present in the data set.

35) What do you understand by Boltzmann Machine?

A Boltzmann machine (also known as stochastic Hopfield network with hidden units) is a type of recurrent neural network. In a Boltzmann machine, nodes make binary decisions with some bias. Boltzmann machines can be strung together to create more sophisticated systems such as deep belief networks. Boltzmann Machines can be used to optimize the solution to a problem.

Some important points about Boltzmann Machine-

* It uses a recurrent structure.
* It consists of stochastic neurons, which include one of the two possible states, either 1 or 0.
* The neurons present in this are either in an adaptive state (free state) or clamped state (frozen state).
* If we apply simulated annealing or discrete Hopfield network, then it would become a Boltzmann Machine.

36) What is Model Capacity?

The capacity of a deep learning neural network controls the scope of the types of mapping functions that it can learn. Model capacity can approximate any given function. When there is a higher model capacity, it means that the larger amount of information can be stored in the network.

37) What is the cost function?

A cost function describes us how well the neural network is performing with respect to its given training sample and the expected output. It may depend on variables such as weights and biases.It provides the performance of a neural network as a whole. In deep learning, our priority is to minimize the cost function. That's why we prefer to use the concept of gradient descent.

38) Explain gradient descent?

An optimization algorithm that is used to minimize some function by repeatedly moving in the direction of steepest descent as specified by the negative of the gradient is known as gradient descent. It's an iteration algorithm, in every iteration algorithm, we compute the gradient of a cost function, concerning each parameter and update the parameter of the function via the following formula:

Where,

**Θ - is the parameter vector,**

**α - learning rate,**

**J(Θ) - is a cost function**

In machine learning, it is used to update the parameters of our model. Parameters represent the coefficients in linear regression and weights in neural networks.

39) Explain the following variant of Gradient Descent: Stochastic, Batch, and Mini-batch?

* **Stochastic Gradient Descent**  
  Stochastic gradient descent is used to calculate the gradient and update the parameters by using only a single training example.
* **Batch Gradient Descent**  
  Batch gradient descent is used to calculate the gradients for the whole dataset and perform just one update at each iteration.
* **Mini-batch Gradient Descent**  
  Mini-batch gradient descent is a variation of stochastic gradient descent. Instead of a single training example, mini-batch of samples is used. Mini-batch gradient descent is one of the most popular optimization algorithms.

40) What are the main benefits of Mini-batch Gradient Descent?

* It is computationally efficient compared to stochastic gradient descent.
* It improves generalization by finding flat minima.
* It improves convergence by using mini-batches. We can approximate the gradient of the entire training set, which might help to avoid local minima.

41) What is matrix element-wise multiplication? Explain with an example.

Element-wise matrix multiplication is used to take two matrices of the same dimensions. It further produces another combined matrix with the elements that are a product of corresponding elements of matrix a and b.

42) What do you understand by a convolutional neural network?

A convolutional neural network, often called CNN, is a feedforward neural network. It uses convolution in at least one of its layers. The convolutional layer contains a set of filter (kernels). This filter is sliding across the entire input image, computing the dot product between the weights of the filter and the input image. As a result of training, the network automatically learns filters that can detect specific features.

43) Explain the different layers of CNN.

There are four layered concepts that we should understand in CNN (Convolutional Neural Network):

* **Convolution**  
  This layer comprises of a set of independent filters. All these filters are initialized randomly. These filters then become our parameters which will be learned by the network subsequently.
* **ReLU**  
  The ReLu layer is used with the convolutional layer.
* **Pooling**  
  It reduces the spatial size of the representation to lower the number of parameters and computation in the network. This layer operates on each feature map independently.
* **Full Collectedness**  
  Neurons in a completely connected layer have complete connections to all activations in the previous layer, as seen in regular Neural Networks. Their activations can be easily computed with a matrix multiplication followed by a bias offset.

44) What is an RNN?

RNN stands for Recurrent Neural Networks. These are the artificial neural networks which are designed to recognize patterns in sequences of data such as handwriting, text, the spoken word, genomes, and numerical time series data. RNN use backpropagation algorithm for training because of their internal memory. RNN can remember important things about the input they received, which enables them to be very precise in predicting what's coming next.

45) What are the issues faced while training in Recurrent Networks?

Recurrent Neural Network uses backpropagation algorithm for training, but it is applied on every timestamp. It is usually known **as Back-propagation Through Time** (BTT).

There are two significant issues with Back-propagation, such as:

* **Vanishing Gradient**  
  When we perform Back-propagation, the gradients tend to get smaller and smaller because we keep on moving backward in the Network. As a result, the neurons in the earlier layer learn very slowly if we compare it with the neurons in the later layers.Earlier layers are more valuable because they are responsible for learning and detecting simple patterns. They are the building blocks of the network.  
  If they provide improper or inaccurate results, then how can we expect the next layers and complete network to perform nicely and provide accurate results. The training procedure tales long, and the prediction accuracy of the model decreases.
* **Exploding Gradient**  
  Exploding gradients are the main problem when large error gradients accumulate. They provide result in very large updates to neural network model weights during training.  
  Gradient Descent process works best when updates are small and controlled. When the magnitudes of the gradient accumulate, an unstable network is likely to occur. It can cause poor prediction of results or even a model that reports nothing useful.

46) Explain the importance of LSTM.

LSTM stands for **Long short-term memory**. It is an artificial RNN (Recurrent Neural Network) architecture, which is used in the field of deep learning. LSTM has feedback connections which makes it a "general purpose computer." It can process not only a single data point but also entire sequences of data.

They are a special kind of RNN which are capable of learning long-term dependencies.

47) What are the different layers of Autoencoders? Explain briefly.

An autoencoder contains three layers:

* **Encoder**  
  The encoder is used to compress the input into a latent space representation. It encodes the input images as a compressed representation in a reduced dimension. The compressed images are the distorted version of the original image.
* **Code**  
  The code layer is used to represent the compressed input which is fed to the decoder.
* **Decoder**  
  The decoder layer decodes the encoded image back to its original dimension. The decoded image is a reduced reconstruction of the original image. It is automatically reconstructed from the latent space representation.

48) What do you understand by Deep Autoencoders?

Deep Autoencoder is the extension of the simple Autoencoder. The first layer present in DeepAutoencoder is responsible for first-order functions in the raw input. The second layer is responsible for second-order functions corresponding to patterns in the appearance of first-order functions. Deeper layers which are available in the Deep Autoencoder tend to learn even high-order features.

A deep autoencoder is the combination of two, symmetrical deep-belief networks:

* First four or five shallow layers represent the encoding half.
* The other combination of four or five layers makes up the decoding half.

49) What are the three steps to developing the necessary assumption structure in Deep learning?

The procedure of developing an assumption structure involves three specific actions.

* The first step contains algorithm development. This particular process is lengthy.
* The second step contains algorithm analyzing, which represents the in-process methodology.
* The third step is about implementing the general algorithm in the final procedure. The entire framework is interlinked and required for throughout the process.

50) What do you understand by Perceptron? Also, explain its type.

A perceptron is a neural network unit (an artificial neuron) that does certain computations to detect features. It is an algorithm for supervised learning of binary classifiers. This algorithm is used to enable neurons to learn and processes elements in the training set one at a time.

There are two types of perceptrons:

* **Single-Layer Perceptron**  
  Single layer perceptrons can learn only linearly separable patterns.
* **Multilayer Perceptrons**  
  Multilayer perceptrons or feedforward neural networks with two or more layers have the higher processing power.

**51. Explain Convolutional Neural Network**

The AlexNet, was first able to beat more traditional geometrical approaches on the most popular object recognition contest - the ILSVRC. Ever since, DCNNs have been achieving state-of-the-art results on object recognition tasks.

**# Why CNN**

If you would use Fully connected ANN then you would have to train a lot of parameters and network has to optimize all of these parameters,the training process could then become very time and storage intensive. In order to solve this computational problem, a different kind of network architecture is used, called \*\*Convolutional Neural Network (CNN).

CNNs introduce two new types of hidden layers in which each neuron is only connected to a small subset of the neurons in the previous layer, to prevent the

aforementioned problem.</br>

**1. Convolutional Layer:** What separates a convolutional layer from a fully-connected one is that each neuron is only connected to a small, local subset of the neurons in the previous layer, which is a square sized region across the height and width dimensions. The size of this square is a hyperparameter named Receptive Field.

Neurons of the convolution operator can recognize certain local patterns of the previous layer’s output.In order to now reconize multiple different features within one layer, it is required to have several Filters. How many convolutions are being conducted is defined by another hyperparameter, called Stride, which determines how big the gap between two scanned regions is.

Why Padding ?? (Having Certain Width & Height + Cover the edges of images much better )

**2. Pooling Layer:** The third kind of layer, which has the purpose of decreasing the complexity of CNNs, is the Pooling Layer. Similarly to the convolutional layer, neurons in the pooling layer are connected to a square sized region across the width and height dimensions of the previous layer. The main difference between convolution and pooling is that a pooling layer is not Parametrized.

Max Pooling & Average Pooling

**##52. APPLICATIONS OF DCNN FOR OBJECT RECOGNITION TASKS**

Three object recognition tasks - classification, localization and detection - and how each of them can be tackled with DCNNs.

**### Classification:**

The task of Image Classification describes the challenge of categorizing a given image into one of several classes. A possible application of this could be the recognition of hand-written digits, where the input image is classified as one of the ten classes.

**### Localization:**

For Localization, the information about which category an image belongs to is already available and the task is to instead figure out where exactly the object is located in the image. This location is typically specified by a two-dimensional bounding box, which consists of four values that describe the location of two opposite couples of corners. Finding these four values is the main challenge of localization and is commonly referred to as \*\*Bounding Box Regression\*\*.

To perform a localization task, we can use a similar architecture as the one we defined for classification. The only thing that has to be modified is the final output layer, which can simply be replaced by another output layer that performs the bounding box regression instead.

**### Detection:**

In contrast to multi-class localization, the number of objects in a given image is not known prior to the execution, when performing Object Detection. In order to use DCNNs for such object detection tasks, the architecture needs to be extended to handle the flexible amount of detections.

**53 -R-CNN:** Region-based Convolutional Neural Networks (R-CNNs) are using Region Proposal Networks to only extract potentially interesting regions. These regions are called Regions of Interest (ROIs). They are obtained by running a quick segmentation to spot blob-like structures.

**54-R-FCN:** These networks are structured very similarly to the faster R-CNN architecture, but instead of using fully-connected modules to predict classes and bounding boxes for each ROI, R-FCNs use Position-Sensitive Convolutional Modules.

**55 -YOLO(You Only Look Once):** In contrast with region proposal based techniques, Single-Shot detection architectures do not predict any regions of interests, but instead, a fixed amount of detections on the image directly, which are then filtered to contain only the actual detections. These networks do therefore have much faster execution times than region-based architectures but are found to also have a lower detection accuracy. Note : Suitable for real time applications.

**56 - Single Shot MultiBox Detectors (SSD) :** In order to handle the problems of YOLO that arise due to the fixed amount of predictions and cell sizes,Single Shot MultiBox Detectors (SSD) have been developed, which predict detections of different scales and also make predictions for multiple different aspect ratios. Therefore, SSD detectors can make finer predictions, leading to significantly better results.

**## 57- DCNN ARCHITECTURES :**

ImageNet Large Scale Visual Recognition Competition (ILSVRC) - Models in chronological order.

LeNet-5 (1998)\*\* [3 CL + 2 Pooling + 1 FC]

Most of the DCNNs that are being used for object recognition today are based on this basic architecture. As can be seen, the network architecture is relatively simple, as it only consists of an input layer of size 32 × 32, an output layer of size 10, as well as three 5 × 5 convolutional, two 2×2 pooling and one fully-connected layer in between, making it a total of six hidden layers.

AlexNet (2012)\*\* [5 CL + 3 Pooling + 2 FC]

It was the first DCNN that managed to beat more traditional object recognition approaches in the ILSVRC.

ZFNet and OverFeat (2013) [Same as AlexNet but with a slight modification]

VGGNet and GoogLeNet (2014)

VGGNet, scored second place in the ILSVRC and influenced the deep learning scene in an important way, as they showed that using a deeper architecture does generally lead to better results, which was not obvious at that time.

Google won the ILSVRC with a different very deep network that had 22 parametrized layers - even more than the VGGNet - and was named GoogLeNet. This network was an improvement of the AlexNet that was not only much deeper but also reduced the number of parameters. The latter was achieved by replacing the first fully-connected layer, which is typically accountable for the highest number of parameters, by another convolutional layer. In addition to that, they also implemented the so-called Inception Modules, which enable a network to recognize patterns of different sizes within the same layer.

ResNet (2015)

Their network, called Deep Residual Network, or ResNet for short, is able to perform much better with very deep architectures. In ResNets, convolutional layers are divided into Residual Blocks and for each block a Residual Connections is added, which is bypassing the corresponding block. By adding these residual connections, the result of a training step can be backpropagated to the earlier layers directly, without any interference from subsequent layers. Therefore, residual connections enable the training of even deeper networks.

\* Inception-v4, Inception-ResNet-v1 and ResNeXt (2016)

\* Densenet, DPN and MobileNets (2017)

**## 58 : Activation Function :**

* + Sigmoid (not used because they suffer from vanishing and exploding gradient problem, except in LSTM/RNN)
  + Tanh
  + ReLu (Dead Neurons Problem)
  + Leaky ReLu (Solve the Dead Neurons Problem)
  + ELU (Exponential Linear Unit)

**## 59 : Dropout**

A form of regularization useful in training neural networks. Dropout regularization works by removing a random selection of a fixed number of the units in a network layer for a single gradient step. The more units dropped out, the stronger the regularization. This is analogous to training the network to emulate an exponentially large ensemble of smaller networks.

## 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 is a part of Machine Learning, which involves mimicking the human brain in terms of structures called neurons, thereby forming neural networks.

### ****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
* 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

### ****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?****

Backprobation 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.

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

## 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 are 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.

We will check out neural network interview questions alongside as it is also a vital part of Deep Learning.

### ****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 in 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.

***If you are looking forward to becoming an expert in Deep Learning, make sure to check out Intellipaat’s***[***AI Engineer Course***](https://intellipaat.com/artificial-intelligence-masters-training-course/)***.***

### ****26. What is a CNN?****

CNNs are convolutional neural networks that are used to perform analysis on images and visuals. These classes of neural networks can input a multi-channel image and work on it easily.

These Deep Learning questions must be answered in a concise way. So make sure to understand them and revisit them if necessary.

### ****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 used always 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, 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.

These are some of the vital advanced deep learning interview questions that you have to know about!

### ****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.** Why is it necessary to introduce non-linearities in a neural network?

**Solution**: otherwise, we would have a composition of linear functions, which is also a linear function, giving a linear model. A linear model has a much smaller number of parameters, and is therefore limited in the complexity it can model.

**2.**Describe two ways of dealing with the vanishing gradient problem in a neural network.

**Solution**:

* Using ReLU activation instead of sigmoid.
* Using Xavier initialization.

**3.**What are some advantages in using a CNN (convolutional neural network) rather than a DNN (dense neural network) in an image classification task?

**Solution**: while both models can capture the relationship between close pixels, CNNs have the following properties:

* It is translation invariant — the exact location of the pixel is irrelevant for the filter.
* It is less likely to overfit — the typical number of parameters in a CNN is much smaller than that of a DNN.
* Gives us a better understanding of the model — we can look at the filters’ weights and visualize what the network “learned”.
* Hierarchical nature — learns patterns in by describing complex patterns using simpler ones.

**4.** Describe two ways to visualize features of a CNN in an image classification task.

Solution:

* Input occlusion — cover a part of the input image and see which part affect the classification the most. For instance, given a trained image classification model, give the images below as input. If, for instance, we see that the 3rd image is classified with 98% probability as a dog, while the 2nd image only with 65% accuracy, it means that the part covered in the 2nd image is more important.
* Activation Maximization — the idea is to create an artificial input image that maximize the target response (gradient ascent).

**5.**Is trying the following learning rates: 0.1,0.2,…,0.5 a good strategy to optimize the learning rate?

**Solution:**No, it is recommended to try a logarithmic scale to optimize the learning rate.

**6.**Suppose you have a NN with 3 layers and ReLU activations. What will happen if we initialize all the weights with the same value? what if we only had 1 layer (i.e linear/logistic regression?)

**Solution:**If we initialize all the weights to be the same we would not be able to break the symmetry; i.e, all gradients will be updated the same and the network will not be able to learn. In the 1-layers scenario, however, the cost function is convex (linear/sigmoid) and thus the weights will always converge to the optimal point, regardless of the initial value (convergence may be slower).

**7.**Explain the idea behind the Adam optimizer.

**Solution:**Adam, or adaptive momentum, combines two ideas to improve convergence: per-parameter updates which give faster convergence, and momentum which helps to avoid getting stuck in saddle point.

**8.**Compare batch, mini-batch and stochastic gradient descent.

Solution: batch refers to estimating the data by taking the entire data, mini-batch by sampling a few datapoints, and SGD refers to update the gradient one datapoint at each epoch. The tradeoff here is between how precise the calculation of the gradient is versus what size of batch we can keep in memory. Moreover, taking mini-batch rather than the entire batch has a regularizing effect by adding random noise at each epoch.

**9.**What is data augmentation? Give examples.

**Solution:**Data augmentation is a technique to increase the input data by performing manipulations on the original data. For instance in images, one can: rotate the image, reflect (flip) the image, add Gaussian blur.

**10.**What is the idea behind GANs?

**Solution:**GANs, or generative adversarial networks, consist of two networks (D,G) where D is the “discriminator” network and G is the “generative” network. The goal is to create data — images, for instance, which are undistinguishable from real images. Suppose we want to create an adversarial example of a cat. The network G will generate images. The network D will classify images according to whether they are a cat or not. The cost function of G will be constructed such that it tries to “fool” D — to classify its output always as cat.

**11.**What are the advantages of using Batchnorm?

**Solution:** Batchnorm accelerates the training process. It also (as a byproduct of including some noise) has a regularizing effect.

**12. W**hat is multi-task learning? When should it be used?

**Solution:**Multi-tasking is useful when we have a small amount of data for some task, and we would benefit from training a model on a large dataset of another task. Parameters of the models are shared — either in a “hard” way (i.e the same parameters) or a “soft” way (i.e regularization/penalty to the cost function).

**13.** What is end-to-end learning? Give a few of its advantages.

**Solution:**End-to-end learning is usually a model which gets the raw data and outputs directly the desired outcome, with no intermediate tasks or feature engineering. It has several advantages, among which: there is no need to handcraft features, and it generally leads to lower bias.

**14.**What happens if we use a ReLU activation and then a sigmoid as the final layer?

**Solution:** Since ReLU always outputs a non-negative result, the network will constantly predict one class for all the inputs!

**15.**How to solve the exploding gradient problem?

**Solution:**A simple solution to the exploding gradient problem is gradient clipping — taking the gradient to be ±M when its absolute value is bigger than M, where M is some large number.

**16.** Is it necessary to shuffle the training data when using batch gradient descent?

**Solution:**No, because the gradient is calculated at each epoch using the entire training data, so shuffling does not make a difference.

**17.**When using mini batch gradient descent, why is it important to shuffle the data?

**Solution:** otherwise, suppose we train a NN classifier and have two classes — A and B, and that all samples of one class come before the other class. Not shuffling the data will make the weights converge to a wrong value.

**18.** Describe some hyperparameters for transfer learning.

**Solution:** How many layers to keep, how many layers to add, how many to freeze.

**19**. Is dropout used on the test set?

**Solution**: No! only in the train set. Dropout is a regularization technique that is applied in the training process.

**20**. Explain why dropout in a neural network acts as a regularizer.

**Solution**: There are several (related) explanations to why dropout works. It can be seen as a form of model averaging — at each step we “turn off” a part of the model and average the models we get. It also adds noise, which naturally has a regularizing effect. It also leads to more sparsity of the weights and essentially prevents co-adaptation of neurons in the network.

**21.** Give examples in which a many-to-one RNN architecture is appropriate.

**Solution**: A few examples are: sentiment analysis, gender recognition from speech, .

**22**. When can’t we use BiLSTM? Explain what assumption has to be made.

**Solution**: in any bi-directional model, we assume that we have access to the next elements of the sequence in a given “time”. This is the case for text data (i.e sentiment analysis, translation etc.), but not the case for time-series data.

**23**. True/false: adding L2 regularization to a RNN can help with the vanishing gradient problem.

**Solution**: false! Adding L2 regularization will shrink the weights towards zero, which can actually make the vanishing gradients worse in some cases.

**24**. Suppose the training error/cost is high and that the validation cost/error is almost equal to it. What does it mean? What should be done?

**Solution**: this indicates underfitting. One can add more parameters, increase the complexity of the model, or lower the regularization.

**25**. Describe how L2 regularization can be explained as a sort of a weight decay.

**Solution**: Suppose our cost function is C(w), and that we add a penalization c|w|2 . When using gradient descent, the iterations will look like

w = w -grad(C)(w) — 2cw = (1–2c)w — grad(C)(w)

In this equation, the weight is multiplied by a factor < 1.