Top 20 Deep Learning
Interview Questions and Answers

Share on your Social Media

Want to know more about becoming an expert in IT?

Click Here to Get Started »

100% Placement Assurance

AUTHORISED CERTIFICATION PARTNER **IBM**

Top 20 Deep Learning Interview Questions and Answers

Published On: May 31, 2024

Nowadays, employers are searching for qualified candidates who can create models that closely resemble human behavior using deep learning and machine learning approaches. Therefore, the top 20 frequently asked **deep learning interview questions and answers** are covered in this article.

Download Deep Learning Interview Questions PDF

1. What is deep learning?

Deep learning uses large volumes of structured or unstructured data to train neural networks with

Related Courses at SLA

→ **Deep Learning Training in OMR**

→ **Deep Learning Training in Chennai**

Related Posts

Quick Enquiry
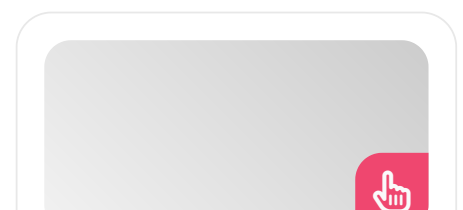
complex algorithms. It carries out intricate tasks to uncover hidden patterns and characteristics (like differentiating a cat from a dog in a photograph).

2. Explain neural networks.

Neural networks are a much simpler version of how people learn, mimicking the firing patterns of neurons in our brains. Three network layers make up the most popular types of neural networks:

- An input layer
- A hidden layer
- An output layer.

Neurons on each sheet, called "nodes," perform various tasks. CNN, RNN, GAN, and other deep learning techniques use neural networks.

3. How do activation functions fit into a neural network?

In its most basic form, an activation function determines whether or not a neuron fires. Any activation function can use the weighted sum of the inputs plus bias as an input. Activation functions include the step function, sigmoid, ReLU, Tanh, Softmax, and so on.

4. What is the Boltzmann machine?

A Boltzmann machine is a fundamental deep-learning model similar to a streamlined Multi-Layer perceptron. This two-layer neural net model makes stochastic decisions about turning on or off neurons. It consists of a visible input layer and a hidden layer. Nodes inside the same layer are not connected; however, they are connected across levels.

5. Define cost function

The cost function, often known as "loss" or "error," is a metric used to assess the performance level of

your model. It is employed in backpropagation to calculate the output layer's error. We use inaccuracy to drive the neural network backward through its many training processes.

6. Why is data normalization necessary, and what does it entail?

"Data normalization" is the process of normalizing and restructuring data. This step of pre-processing removes redundant data. When data is received, receiving the same information in many formats is common. To improve convergence, you should rescale the numbers to fit within a specific range.

7. Explain gradient descent

The best algorithm for minimizing an error or the cost function is known as gradient descent. The objective is to locate a function's local-global minima. This establishes the course that the model should follow to lower the error.

Download Deep Learning Syllabus PDF

8. Describe "Backpropagation"

One of the most popular questions in deep learning interviews is this one. One method to raise the network's performance is backpropagation. To lower the error, it changes the weights and backpropagates the error.

9. What are hyperparameters?

This is yet another question that comes up in deep learning interviews. Once the data is suitably prepared, hyperparameters are typically used when dealing with neural networks.

A hyperparameter is a parameter that is initialized before learning takes place. It establishes the training methodology and the network architecture (including the number of hidden units, learning rate, epochs, etc.).

10. How Does CNN Pooling Operate and What Does It Entail?

By pooling, a CNN's spatial dimensions can be decreased. Sliding a filter matrix over the input matrix reduces the dimensionality through downsampling processes and generates a pooled feature map.

Deep Learning Interview Questions and Answers for Experienced

11. How are LSTM networks operational?

A unique type of recurrent neural network called Long-Short-Term Memory (LSTM) can learn long-term dependencies and has long-term memory retention as its default function. Three steps comprise an LSTM network:

Step 1: The network selects which information to retain and which to discard.

Step 2: Cell state values are updated selectively.

Step 3: The network decides which part of the current state is conveyed to the output.

12. What are CNN's various layers?

CNN is composed of four layers:

Convolutional Layer: This layer divides the data into multiple smaller picture windows by performing a convolutional process.

ReLU Layer: it gives the network non-linearity and turns all of the negative pixels into zero. A rectified feature map is the result.

Pooling Layer: To decrease the dimensionality of the feature map, pooling is a downsampling technique.

The Fully Connected Layer is responsible for identifying and categorizing the things present in the image.

13. Why Would a Recurrent Neural Network Be Different from a Feedforward Neural Network?

Feedforward: Transmissions from an input to an output of a feedforward neural network are one-way. The network just takes the current input into account; feedback loops are absent. For example, CNN is unable to remember past inputs.

Recurrent: A looped network is produced when signals from a recurrent neural network travel in both directions. Its internal memory allows it to remember earlier data and generate a layer's output by combining the inputs it received before with the present input.

14. Explain multi-layer perception

MLPs feature three layers: an input layer, a hidden layer, and an output layer, just like neural networks. Its structure is identical to that of a single-layer perceptron with one or more concealed layers.

- A single-layer perceptron can only classify linear separable classes with binary output (0,1); MLP can classify nonlinear classes.
- Every node in the other layers—aside from the input layer—uses a nonlinear activation function.
- This indicates that all nodes and weights are joined together to produce the output, which is the basis for the input layers, the incoming data, and the activation function.
- MLP employs a supervised learning method called "backpropagation."
- Using the cost function, the neural network computes the error during backpropagation.
- This error is propagated backward from its source.

Deep Learning Salary

15. What does transfer learning mean to

you? List a few popular models for transfer learning.

The practice of transferring knowledge from one model to another without starting from scratch is called transfer learning. It employs key components of an already-learned model to solve novel but related machine-learning tasks.

Several widely used models for transfer learning include:

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

16. What distinguishes TensorFlow's 'SAME' from 'VALID' padding?

tf.nn.max_pool uses the Tensorflow library to perform the max-pooling procedure. The padding argument of *Tf.nn.max_pool* accepts two values: SAME and VALID.

By using padding == "SAME," it is ensured that every input element is filtered.

The filter and designated stride completely cover the input image. Since the output size (when stride = 1) is the same as the input size, the padding type is called SAME.

Padding == "VALID" indicates that the input image has no padding.

The input image is always where the filter window is located. For a filter to completely cover the input image and for you to define the stride, it is assumed that all of the dimensions are correct.

17. Describe the Adam optimization algorithm.

Adam optimization, or adaptive moment estimation, is an extension of stochastic gradient

descent. When dealing with complicated issues involving enormous volumes of data or parameters, this algorithm is helpful. It is effective and requires little memory.

The Adam optimization technique combines two approaches to gradient descent:

Momentum and Root Mean Square Propagation.

18. How can a neural network's hyperparameters be trained?

Four elements can be used to train a neural network's hyperparameters:

Batch size: This shows how big the input data is.

Epochs: Indicates how many times the neural network has access to the training data during a training session.

Momentum: A tool for determining what happens next when data is executed.

The learning rate denotes the amount of time needed for the network to adjust its settings and acquire new skills.

19. For an image classification problem, why is a convolutional neural network better than a dense neural network?

- Compared to a dense neural network, a convolutional neural network has a significantly smaller number of parameters. A CNN is, hence, less prone to overfitting.
- CNN lets you see the network's learning by visualizing the weights of a filter. Consequently, this improves our comprehension of the model.
- CNN uses a hierarchical training process to teach its models; in other words, it learns patterns by utilizing simpler ones to explain more complex ones.

20. Why is the mini-batch gradient such

a helpful tool?

- When compared to stochastic gradient descent, the mini-batch gradient is far more efficient.
- Locating the flat minima enables you to achieve generalization.
- To enable gradient approximation for the entire dataset, a mini-batch gradient aids in avoiding local minima.

21. What are TensorFlow programming elements?

Constants: Constants are variables with constant values. The tf.constant() command is used to define a constant. For example:

> *a = tf.constant(2.0,tf.float32)*
>
> *b = tf.constant(3.0)*
>
> *Print(a, b)*

Variables: We can add new trainable parameters to the graph using variables. When executing the graph in a session, we utilize the tf.Variable() command to define and initialize variables. For example:

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

Placeholders: They let us incorporate outside data into a TensorFlow model. It allows one to assign a value at a later time. The tf.placeholder() command is used to define a placeholder. For 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 conducted to assess the nodes. We refer to this as 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))
```

Conclusion

You will feel more confident to ace machine learning interviews with the help of this set of deep learning interview questions and answers. Join us for the Best **Deep Learning Training in Chennai.**

Deep Learning Training

Share on your Social Media

Navigation

About Us

Blog Posts

Careers

Contact

Placement Training

Corporate Training

Hire With Us

Job Seekers

SLA's Recently Placed Students

Reviews

Sitemap

Softlogic Academy

Softlogic Systems

KK Nagar [Corporate Office]

No.10, PT Rajan Salai, K.K. Nagar, Chennai – 600 078.
Landmark: Karnataka Bank Building
Phone: +91 86818 84318

Email: enquiry@softlogicsys.in
Map: Google Maps Link

OMR

No. E1-A10, RTS Food Street
92, Rajiv Gandhi Salai (OMR),
Navalur, Chennai - 600 130.
Landmark: Adj. to AGS Cinemas
Phone: +91 89256 88858
Email: info@softlogicsys.in
Map: Google Maps Link

Courses

- Python
- Software Testing
- Full Stack Developer
- Java
- Power BI
- Clinical SAS
- Data Science
- Embedded
- Cloud Computing
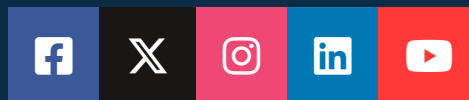- Hardware and Networking
- VBA Macros
- Mobile App Development
- DevOps

Important Links

- Disclaimer
- Privacy Policy
- Terms and Conditions

Social Media Links

Review Sources

- Google
- Trustpilot
- Glassdoor
- Mouthshut
- Sulekha
- Justdial
- Ambitionbox
- Indeed
- Software Suggest
- Sitejabber