



Interview Prep: 40 Artificial Intelligence Questions

www.springboard.com

20 mins read



Over the last decade, artificial intelligence (AI) has grown from a pipe dream into the driving force behind the fourth industrial revolution. If you browse through the world's leading job boards, you'll find that it's at the heart of some of the most [in-demand tech careers today](#).

"Everyone's trying to figure out ways to optimize their businesses and their practices, and how to automate and make their day-to-day lives a little bit easier, or a little bit more productive and functional,"

notes [Stephen Zafarino](#), vice president of national recruiting at Mondo.

So, whether your next job interview is related to data science, machine learning (ML), or deep learning (DL), you can bet that artificial intelligence questions will come up.

Artificial Intelligence Questions: Categories

Because it's a broad area of computer science, AI questions will keep popping up in various job interview scenarios. To make it easier for you to navigate through this space, we have curated a list of questions about artificial intelligence and divided them into multiple categories.

So whether you're hoping to move up the data science career ladder or looking to start your first [machine learning internship](#), make sure that you brush up on these AI interview questions and answers so you can walk into your next interview oozing confidence.

Artificial Intelligence Questions: Introduction to AI

If your AI interview is for an internship, there's a good chance that the interviewer will try to break the ice and make you feel more comfortable by asking some "simple" general interest questions.

These types of questions usually cover the basics, so even if they sound straightforward, you have to make sure that you don't get stumped (seemingly simple questions require that your answer be

delivered easily and flawlessly). However, it can quickly get more involved, so you have to be ready for whatever they throw at you.

Related: [The Most Common Machine Learning Terms, Explained](#)

1. What is artificial intelligence?

AI can be described as an [area of computer science that simulates human intelligence in machines](#). It's about smart algorithms making decisions based on the available data.

Whether it's Amazon's Alexa or a self-driving car, the goal is to mimic human intelligence at lightning speed (and with a reduced rate of error).

More reading: [What is AI? Everything you need to know about Artificial Intelligence](#)

2. What are intelligent agents?

An intelligent agent is an autonomous entity that leverages sensors to understand a situation and make decisions. It can also use actuators to perform both simple and complex tasks.

In the beginning, it might not be so great at performing a task, but it will improve over time. The [Roomba vacuum cleaner](#) is an excellent example of this.

More reading: [Intelligent agents defending for an IoT world](#)

3. What's the most popular programming language used in AI?

The open-source modular programming language Python leads the AI industry because of its simplicity and predictable coding behavior.

Its popularity can be attributed to open-source libraries like Matplotlib and NumPy, efficient frameworks such as Scikit-learn, and practical version libraries like Tensorflow and VTK.

There's a chance that the interviewer might keep the conversation going and ask you for more examples. If that happens, you can mention the following:

- Java
- Julia
- Haskell
- Lisp

More reading: [*The 5 Best Programming Languages for AI*](#)

4. What are AI neural networks?

[Neural networks](#) in AI mathematically model how the human brain works. This approach enables the machine to think and learn as humans do. This is how smart technology today recognizes speech, objects, and more.

More reading: [*Exploring Neural Networks with Activation Atlases*](#)

5. What's the difference between strong AI and weak AI?

The difference between the two is just like the terms sound. Strong AI can successfully imitate human intelligence and is at the core of advanced robotics.

Weak AI can only predict specific characteristics that resemble human intelligence. Alexa and Siri are excellent examples of weak AI.

Strong AI

- Can be applied widely
- Extensive scope

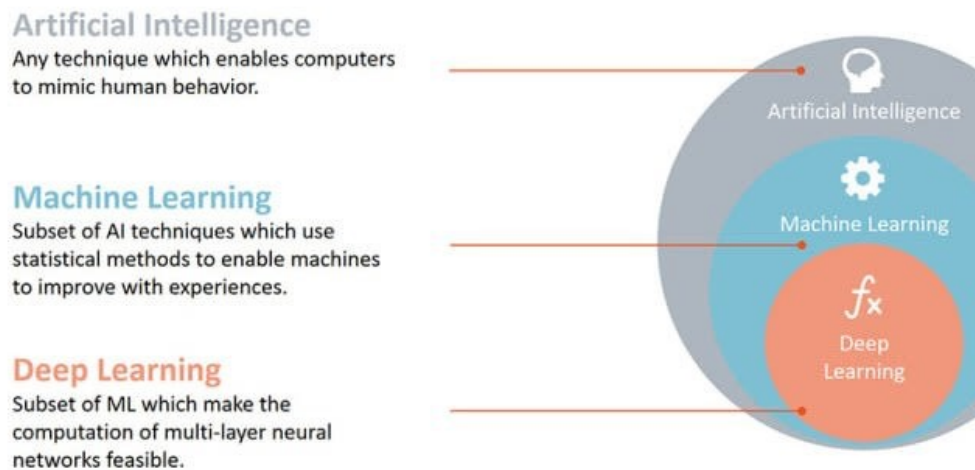
- Human-level intelligence
- Processes data by using clustering and association

Weak AI

- Can be great at performing some simple tasks
- Uses both supervised and unsupervised learning
- The scope can be minimal

More reading: [What's the Difference Between Weak and Strong AI?](#)

6. What's the difference between AI and ML?



([Source](#))

AI and ML are closely related, but these terms aren't interchangeable. ML actually falls under the umbrella of AI. It demands that machines carry out tasks in the same way that humans do.

The current application of ML in AI is based around the idea that we should enable access to data so machines can observe and learn for themselves.

More reading: [What's the Difference between Machine Learning and AI?](#)

7. How would you describe ML to a non-technical person?

ML is geared toward pattern recognition. A great example of this is your Facebook newsfeed and Netflix's recommendation engine.

In this scenario, ML algorithms observe patterns and learn from them. When you deploy an ML program, it will keep learning and improving with each attempt.

If the interviewer prods you to provide more real-world examples, you can list the following:

- Amazon product recommendations
- Fraud detection
- Search ranking
- Spam detection
- Spell correction

More reading: [How do you explain Machine Learning and Data Mining to Non-Computer Science people?](#)

8. What are some examples of AI in use?

Some compelling examples of AI applications are:

- [Chatbots](#)
- Facial recognition
- Image tagging
- Natural language processing
- Sales prediction
- Self-driving cars
- Sentiment analysis

More reading: [Ask the AI experts: What are the applications of AI?](#)

9. What's a Turing test?

The Turing test, named after [Alan Turing](#), is a method of testing a machine's human-level intelligence. For example, in a human-versus-machine scenario, a judge will be tasked with identifying which terminal was occupied by a human and which was occupied by a computer based on individual performance.

Whenever a computer can pass off as a human, it's deemed intelligent. The game has since evolved, but the premise remains the same.

More reading: [The Turing Test](#)

10. What's TensorFlow?

TensorFlow is an open-source framework dedicated to ML. It's a comprehensive and highly adaptable ecosystem of libraries, tools, and community resources that help developers build and deploy ML-powered applications. Both AlphaGo and Google Cloud Vision were built on the Tensorflow platform.

More reading: [TensorFlow Tutorial from Scratch: Building a Deep Learning Model on Fashion MNIST Dataset \(Part 1\)](#)

11. Why is game theory important to AI?

Game theory, developed by [American mathematician Josh Nash](#), is essential to AI because it plays an underlying role in how these smart algorithms improve over time.

At its most basic, AI is about algorithms that are deployed to find solutions to problems. Game theory is about players in opposition trying to achieve specific goals. As most aspects of life are about competition, game theory has many meaningful real-world applications.

These problems tend to be dynamic. Some game theory problems are natural candidates for AI algorithms. So, whenever game theory is

applied, multiple AI agents that interact with each other will only care about utility to itself.

Data scientists within this space should be [aware of the following games](#):

- Symmetric vs. asymmetric
- Perfect vs. imperfect information
- Cooperative vs. non-cooperative
- Simultaneous vs. sequential
- Zero-sum vs. non-zero-sum

More reading: [What is the connection between Game Theory and AI?](#)

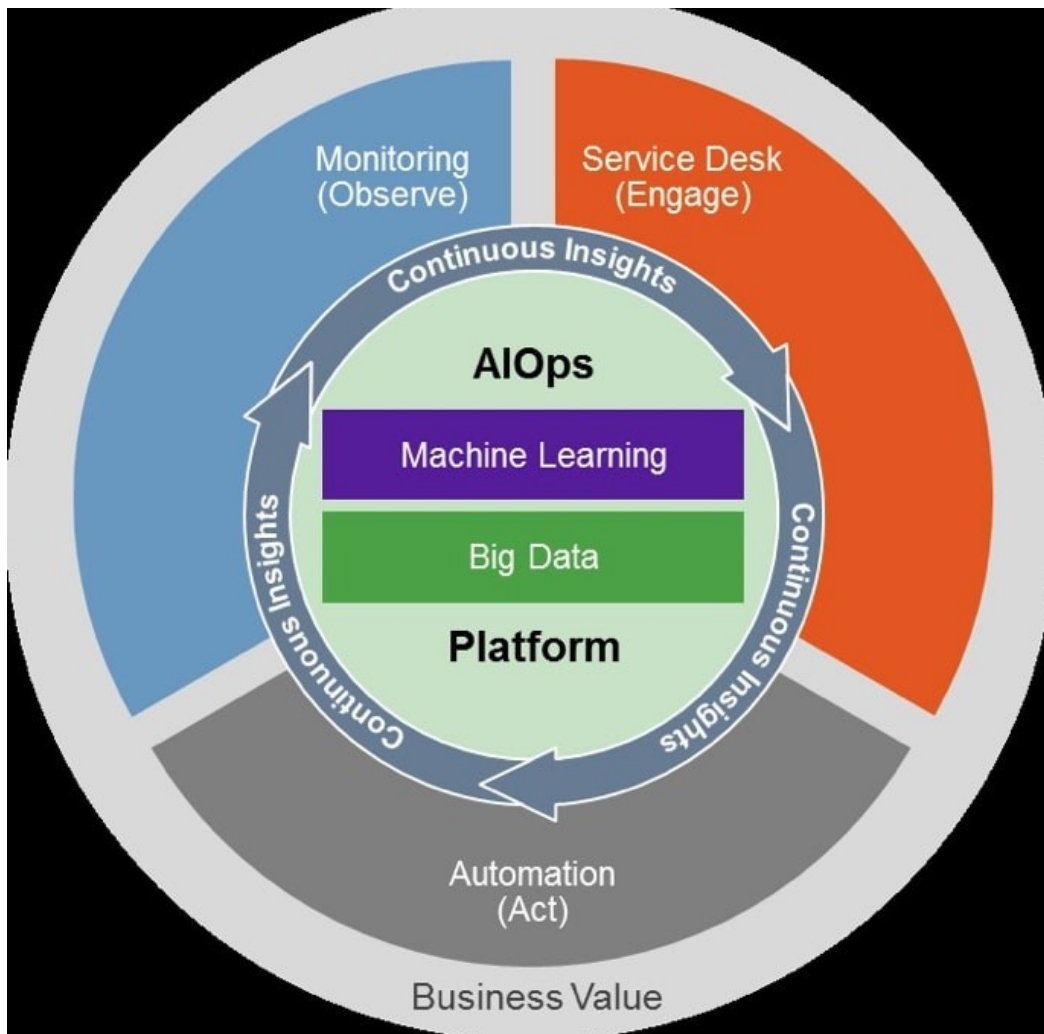
12. In your opinion, how will AI impact application development?

These types of questions help the interviewer ascertain your level of interest in the field. If you're naturally passionate about AI and everything related to it, you should have some knowledge about current industry trends.

So, if you have been actively following this space, you'll know all about AIOps. In the coming months, you can expect AI to be more involved in how we build applications. It has the potential to transform how we use and manage the infrastructure at a micro and macro level.

Some say that DevOps will be replaced by what they are calling AIOps because it allows developers to engage in accurate root cause analysis by combining big data, ML, and visualization.

AIOps can be described as a multilayered platform that can be used to automate and improve IT operations. In this scenario, developers can leverage analytics and ML to collect and process data from a variety of sources. This information can then be analyzed in real time to identify and rectify problems.



([Source](#))

More reading: [AIOps: Is DevOps Ready for an Infusion of Artificial Intelligence?](#)

13. What would you say are common misconceptions about AI?

Many AI-related misconceptions are making the rounds in the age of “fake news.” The most common ones are:

- AI will replace humans
- AI systems aren’t safe
- AI will lead to significant unemployment

While these types of stories are common, they're far from the truth. Even though some AI-based technology is able to complete some tasks—for example, analyzing zettabytes of data in less than a second—it still needs humans to gather the data and define the patterns for identification.

So we aren't near a reality where technology has the potential to replace us or our jobs.

More reading: [What's Behind the Hype About Artificial Intelligence?](#)

14. Can you name the properties of a good knowledge representation system?

From the perspective of [systems theory](#), a good knowledge representation system will have the following:

- **Acquisition efficiency** to acquire and incorporate new data
- **Inferential adequacy** to derive knowledge representation structures like symbols when new knowledge is learned from old knowledge
- **Inferential efficiency** to enable the addition of data into existing knowledge structures to help the inference process
- **Representation adequacy** to represent all the knowledge required in a specific domain

More reading: [Knowledge representation in AI](#)

15. What are the different types of keys in a relational database?

There are a variety of keys in a relational database, including:

- **Alternate keys** are candidate keys that exclude all primary keys.

- **Artificial keys** are created by assigning a unique number to each occurrence or record when there aren't any compound or standalone keys.
- **Compound keys** are made by combining multiple elements to develop a unique identifier for a construct when there isn't a single data element that uniquely identifies occurrences within a construct. Also known as a composite key or a concatenated key, compound keys consist of two or more attributes.
- **Foreign keys** are groups of fields in a database record that point to a key field or a group of fields that create a key of another database record that's usually in a different table. Often, foreign keys in one table refer to primary keys in another. As the referenced data can be linked together quite quickly, it can be critical to database normalization.
- **Natural keys** are data elements that are stored within constructs and utilized as primary keys.
- **Primary keys** are values that can be used to identify unique rows in a table and the attributes associated with them. For example, these can take the form of a Social Security number that's related to a specific person. In a relational model of data, the primary key is the candidate key. It's also the primary method used to identify a tuple in each possible relation.
- **Super keys** are defined in the relational model as a set of attributes of a relation variable. It holds that all relations assigned to that variable don't have any distinct tuples. They also don't have the same values for the attributes in the set. Super keys also are defined as a set of attributes of a relational variable upon which all of the functionality depends.

More reading: [What are the different types of keys in RDBMS?](#)

Artificial Intelligence Questions:

Statistics

AI, ML, and data science all have a great deal of overlap, so it's crucial to cover all bases before your AI interview. However, it's important to note that these fields aren't interchangeable. Although everything is relative, AI produces actions, ML produces predictions, and data science produces insights.

So what kind of potential data science-related AI questions should you be prepared for? Let's take a look.

16. In Python's standard library, what packages would you say are the most useful for data scientists?

Python wasn't built for data science. However, in recent years it has grown to become the go-to programming language for the following:

- Machine learning
- Predictive analytics
- Simple data analytics
- Statistics

For data science projects, the following packages in the Python standard library will make life easier and accelerate deliveries:

- **NumPy** (to process large multidimensional arrays, extensive collections of high-level mathematical functions, and matrices)
- **Pandas** (to leverage built-in methods for rapidly combining, filtering, and grouping data)
- **SciPy** (to extend NumPy's capabilities and solve tasks related to integral calculus, linear algebra, and probability theory)

More reading: [20 Python Interview Questions and Answers—Start Preparing for Your Ideal Job](#)

17. What is collaborative filtering?

Collaborative filtering can be described as a process of finding patterns from available information to build personalized recommendations. You can find collaborative filtering in action when you visit websites like Amazon and IMDB.

Also known as social filtering, this approach essentially makes suggestions based on the recommendations and preferences of other people who share similar interests.

More reading: [Collaborative Filtering](#)

18. Can you list some disadvantages related to linear models?

There are many disadvantages to using linear models, but the main ones are:

- Errors in linearity assumptions
- Lacks autocorrelation
- It can't solve overfitting problems
- You can't use it to calculate outcomes or binary outcomes

More reading: [What are the limitations of linear regression modeling in data analysis?](#)

19. What's a feature vector?

A feature vector is an n-dimensional vector that contains essential information that describes the characteristics of an object. For example, it can be an object's numerical features or a list of numbers taken from the output of a neural network layer.

In AI and data science, feature vectors can be used to represent numeric or symbolic characteristics of an object in mathematical terms for seamless analysis.

Let's break this down. A data set is usually organized into multiple examples where each example will have several features. However, a feature vector won't have the same feature for numerous examples. Instead, each example will correspond to one feature vector that will contain all the numerical values for that example object.

Feature vectors are often stacked into a design matrix. In this scenario, each row will be a feature vector for one example. Each column will feature all the examples that correspond to that particular feature. This means that it will be like a matrix, but with just one row and multiple columns (or a single column and multiple rows) like [1,2,3,5,6,3,2,0].

More reading: [Extract a feature vector for any image with PyTorch](#)

20. What are the typical characteristics of elements in a list and a dictionary?

In lists, elements maintain their order unless they are explicitly commanded to re-order. These can be made up of any data type that can be all the same or mixed. However, elements in lists can only be accessed via numeric, zero-based indices.

In a dictionary, the order isn't guaranteed. However, each entry will be assigned a key and a value. As a result, elements within a dictionary can be accessed by using their individual key.

So whenever you have a set of unique keys, you have to use a dictionary. Whenever a collection of items are in order, you can use a list.

It's difficult to predict how an AI interview will unfold, so if they follow up by asking you how to get a list of all the keys in a dictionary, respond with the following:

To obtain a list of keys in a dictionary, you'll have to use the following function `keys()`:

```
mydict={'a':1,'b':2,'c':3,'e':5}
mydict.keys()
dict_keys(['a', 'b', 'c', 'e'])
```

More reading: [How to Sort Python Dictionaries by Key or Value](#)

21. What's selection bias? What other types of biases could you encounter during sampling?

When you're dealing with a non-random sample, selection bias will occur due to flaws in the selection process. This happens when a subset of the data is consistently excluded because of a particular attribute. This exclusion will distort results and influence the statistical significance of the test.

Other types of biases include survivorship bias and undercoverage bias. It's important to always consider and reduce such biases because you'll want your smart algorithms to make accurate predictions based on the data.

More reading: [Mitigating Bias in AI Models](#)

22. What's a random forest? Could you explain its role in AI?

A random forest is a data construct that's applied to ML projects to develop a large number of random decision trees while analyzing variables.





([Source](#))

These algorithms can be leveraged to improve the way technologies analyze complex data sets. The basic premise here is that multiple weak learners can be combined to build one strong learner.

This is an excellent tool for AI and ML projects because it can work with large labeled and unlabeled data sets with a large number of attributes. It can also maintain accuracy when some data is missing. As it can model the importance of attributes, it can be used for dimensionality reduction.

More reading: [Machine Learning With Random Forest](#)

23. What's an eigenvalue? What about an eigenvector?

The directions along which a particular linear transformation compresses, flips, or stretches is called eigenvalue. Eigenvectors are used to understand these linear transformations.

For example, to make better sense of the covariance of the covariance matrix, the eigenvector will help identify the direction in which the covariances are going. The eigenvalues will express the importance of each feature.

Eigenvalues and eigenvectors are both critical to computer vision and ML applications. The most popular of these is known as [principal component analysis](#) for dimensionality reduction (e.g., eigenfaces for face recognition).

More reading: [What are eigenvectors and eigenvalues?](#)

24. Would you use batch normalization? If so, can you explain why?

The idea here is to standardize the data before sending it to another layer. This approach helps reduce the impact of previous layers by keeping the mean and variance constant. It also makes the layers independent of each other to achieve rapid convergence. For example, when we normalize features from 0 to 1 or from 1 to 100, it helps accelerate the learning cycle.

Check out this video: [Why Does Batch Normalization Work?](#)

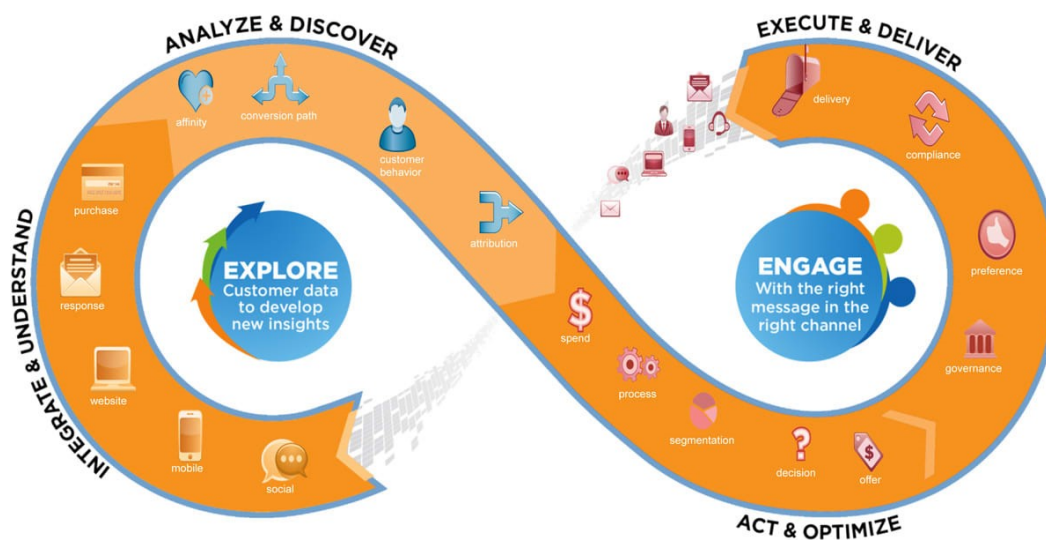
Artificial Intelligence Questions: Programming

AI interview questions are bound to enter the sphere of programming sooner rather than later. So let's dive right into it with the following AI questions and answers.

25. What's a hash table?

There are two parts to a hash table. The first is an array, or the actual table where the data is stored, and the other is a mapping function that's known as the hash function.

It's a data structure that implements an associative array abstract data type that can map key values. It can also compute an index into an array of slots or buckets where the desired value can be found.



([Source](#))

More reading: [Basics of Hash Tables](#)

26. What are the different algorithm techniques you can use in AI and ML?

Some algorithm techniques that can be leveraged are:

- Learning to learn
- Reinforcement learning (deep adversarial networks, q-learning, and temporal difference)
- Semi-supervised learning

- Supervised learning (decision trees, linear regression, naive bayes, nearest neighbor, neural networks, and support vector machines)
- Transduction
- Unsupervised learning (association rules and k-means clustering)

More reading: [Types of Machine Learning Algorithms You Should Know](#)

27. How would you go about choosing an algorithm to solve a business problem?

First, you have to develop a “problem statement” that’s based on the problem provided by the business. This step is essential because it’ll help ensure that you fully understand the type of problem and the input and the output of the problem you want to solve.

The problem statement should be simple and no more than a single sentence. For example, let’s consider enterprise spam that requires an algorithm to identify it.

The problem statement would be: **“Is the email fake/spam or not?”**
In this scenario, the identification of whether it’s fake/spam will be the output.

Once you have defined the problem statement, you have to identify the appropriate algorithm from the following:

- Any classification algorithm
- Any clustering algorithm
- Any regression algorithm
- Any recommendation algorithm

Which algorithm you use will depend on the specific problem you’re trying to solve. In this scenario, you can move forward with a cluster-

ing algorithm and choose a k-means algorithm to achieve your goal of filtering spam from the email system.

While examples aren't always necessary when answering questions about artificial intelligence, sometimes it will help make it easier for you to get your point across.

More reading: [How to Choose ML Algorithm: Machine Learning Questions & Answers Part – III](#)

28. When is it necessary to update an algorithm?

You should update an algorithm when the underlying data source has been changed or whenever there's a case of non-stationarity. The algorithm should also be updated when you want the model to evolve as data streams through the infrastructure.

More reading: ["The data has been changed" error when stepping from main form into subform](#)

29. What's regularization?

When you have underfitting or overfitting issues in a statistical model, you can use the regularization technique to resolve it. Regularization techniques like LASSO help penalize some model parameters if they are likely to lead to overfitting.

If the interviewer follows up with a question about other methods that can be used to avoid overfitting, you can mention cross-validation techniques such as [k-folds cross-validation](#).

Another approach is to keep the model simple by taking into account fewer variables and parameters. Doing this helps remove some of the noise in the training data.

More reading: [Machine Learning Explained: Regularization](#)

30. What's the difference between inductive, deductive, and abductive learning?

Inductive learning describes smart algorithms that learn from a set of instances to draw conclusions. In statistical ML, k-nearest neighbor and support vector machine are good examples of inductive learning.

There are three literals in (top-down) inductive learning:

- Arithmetic literals
- Equality and inequality
- Predicates

In **deductive learning**, the smart algorithms draw conclusions by following a truth-generating structure (major premise, minor premise, and conclusion) and then improve them based on previous decisions. In this scenario, the ML algorithm engages in deductive reasoning using a decision tree.

Abductive learning is a DL technique where conclusions are made based on various instances. With this approach, inductive reasoning is applied to causal relationships in deep neural networks.

More reading: [What's the difference between "inductive, "deductive" and "abductive" reasoning?](#)

31. What steps would you take to evaluate the effectiveness of your ML model?

You have to first split the data set into training and test sets. You also have the option of using a cross-validation technique to further segment the data set into a composite of training and test sets within the data.

Then you have to implement a choice selection of the performance metrics like the following:

- Confusion matrix
- Accuracy
- Precision
- Recall or sensitivity
- Specificity
- F1 score

For the most part, you can use measures such as accuracy, confusion matrix, or F1 score. However, it'll be critical for you to demonstrate that you understand the nuances of how each model can be measured by choosing the right performance measure to match the problem.

More reading: [Performance Metrics for Classification problems in Machine Learning](#)

32. What would you do if data in a data set were missing or corrupted?

Whenever data is missing or corrupted, you either replace it with another value or drop those rows and columns altogether. In Pandas, both [isNull\(\)](#) and [dropNA\(\)](#) are handy tools to find missing or corrupted data and drop those values. You can also use the [fillna\(\)](#) method to fill the invalid values in a placeholder—for example, “o.”

More reading: [5 Ways To Handle Missing Values In Machine Learning Datasets](#)

33. Do you know how to build a simple neural network? Can you demonstrate how you would do it with Python code?

These types of questions are designed to ascertain your programming skills. If you're a coding ninja, you should be able to achieve this with nine lines of Python code.

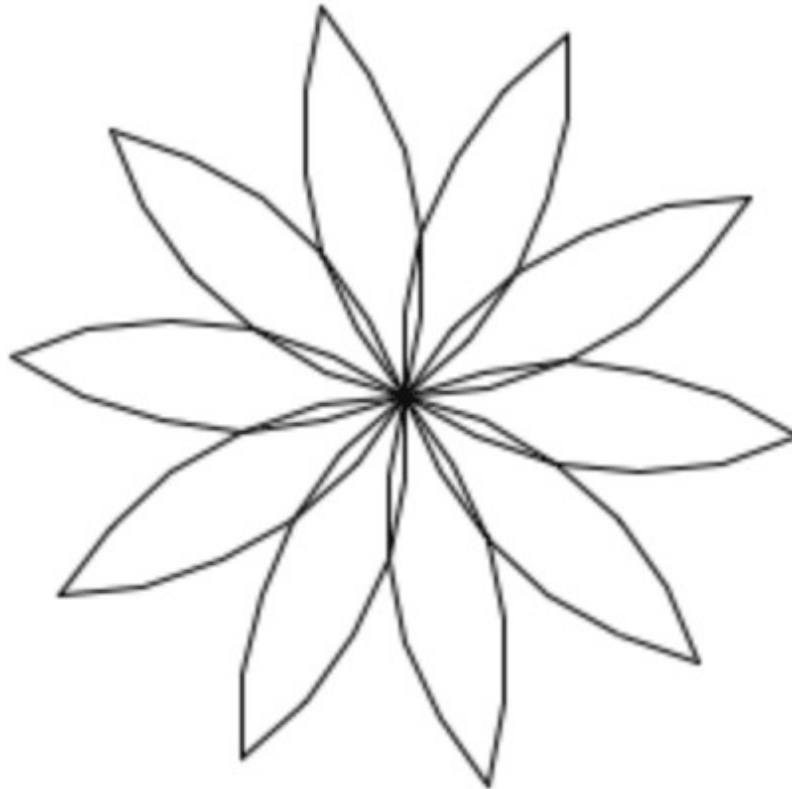
```
from numpy import exp, array, random, dot
training_set_inputs = array([[0, 0, 1], [1, 1, 1], [1, 0, 1], [0, 1, 1]])
training_set_outputs = array([[0, 1, 1, 0]]).T
random.seed(1)
synaptic_weights = 2 * random.random((2, 1)) - 1
```

```
synaptic_weights = 2 * random.random((3, 1)) - 1
for iteration in xrange(10000):
    output = 1 / (1 + exp(-(dot(training_set_inputs, synaptic_weights))))
    synaptic_weights += dot(training_set_inputs.T, (training_set_outputs - output) * output * (1 - output))
print 1 / (1 + exp(-(dot(array([1, 0, 0]), synaptic_weights))))
```

([Source](#))

More reading: [How to build a simple neural network in 9 lines of Python code](#)

34. Can you write a Python program to draw the flower shown below?



([Source](#))

Again, the interviewer is trying to test your coding skills. It's always good to perceive these types of questions as an opportunity to show a potential employer what you can do. So demonstrate your programming skills without hesitation and with confidence.

```
from turtle import Turtle, Screen

def draw_petal(turtle, radius):
    heading = turtle.heading()
    turtle.circle(radius, 60)
    turtle.left(120)
    turtle.circle(radius, 60)
    turtle.setheading(heading)

my_radius = int(input("What is the radius of the flower? "))
my_petals = int(input("How many petals do you want? "))

bob = Turtle()

for _ in range(my_petals):
    draw_petal(bob, my_radius)
    bob.left(360 / my_petals)

bob.hideturtle()

screen = Screen()
screen.exitonclick()
```

USAGE

```
> python3 test.py
What is the radius of the flower? 100
How many petals do you want? 10
```

([Source](#))

More reading: [Drawing a flower with python turtle](#)

Artificial Intelligence Questions: General AI Interest

The tech talent shortage has created fierce demand for your skills, but to land your “dream job” it’ll help if you can demonstrate your passion for the field. Whether your next scheduled AI interview is with a startup or an established tech giant, be ready for wide-ranging questions like the ones listed below.

35. Do you have research experience in AI?

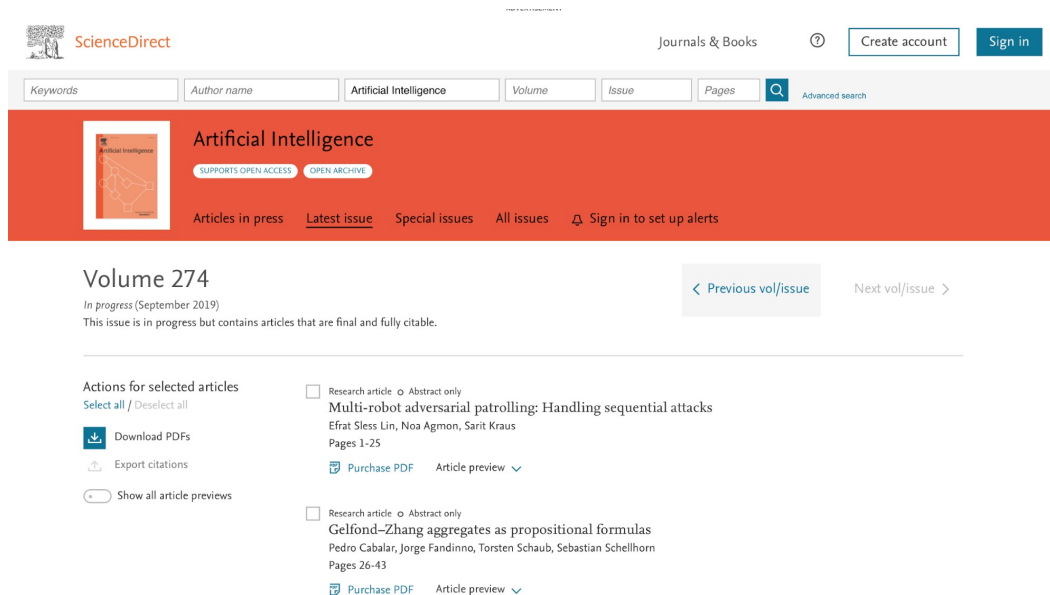
At present, a lot of work within the AI space is research-based. As a result, many organizations will be digging into your background to ascertain what kind of experience you have in this area. If you authored or co-authored research papers or have been supervised by industry leaders, make sure to share that information.

In fact, take it a step further and have a summary of your research experience along with your research papers ready to share with the interviewing panel.

However, if you don't have any formal research experience, have an explanation ready. For example, you can talk about how your AI journey started as a weekend hobby and grew into so much more within a space of two or three years.

36. What's the last AI-related research paper you read? What were your conclusions?

If you're passionate about AI, you have to keep up with scientific research within the field. An excellent place to start is by following [ScienceDirect](https://www.sciencedirect.com) to keep track of published research papers along with what's in the pipeline.



The screenshot shows the ScienceDirect website interface. At the top, there's a navigation bar with the ScienceDirect logo, "Journals & Books" link, and buttons for "Create account" and "Sign in". Below this is a search bar with fields for "Keywords", "Author name", "Artificial Intelligence", "Volume", "Issue", and "Pages", along with an "Advanced search" button. The main content area is a red banner for "Artificial Intelligence" with a journal cover image. It includes links for "Articles in press", "Latest issue", "Special issues", and "All issues", as well as a "Sign in to set up alerts" button. Below the banner, the "Volume 274" section is displayed, noting it is "In progress (September 2019)" and "This issue is in progress but contains articles that are final and fully citable." Navigation buttons for "Previous vol/issue" and "Next vol/issue" are present. A section titled "Actions for selected articles" lists two articles with checkboxes for "Research article" and "Abstract only". The first article is "Multi-robot adversarial patrolling: Handling sequential attacks" by Efrat Sless Lin, Noa Agmon, and Sarit Kraus, with options to "Purchase PDF" and view an "Article preview". The second article is "Gelfond-Zhang aggregates as propositional formulas" by Pedro Cabalar, Jorge Fandinno, Torsten Schaub, and Sebastian Schellhorn, also with "Purchase PDF" and "Article preview" options.

([Source](#))

37. What's your favorite use case?

Just like research, you should be up to date on what's going on in the industry. As such, if you're asked about use cases, make sure that you have a few examples in mind that you can share. Whenever possible, bring up your personal experiences.

You can also share what's happening in the industry. For example, if you're interested in the use of AI in medical images, [Health IT Analytics](#) has some interesting use cases:

- Detecting Fractures And Other Musculoskeletal Injuries
- Aiding In The Diagnosis Neurological Diseases
- Flagging Thoracic Complications And Conditions
- Screening For Common Cancers

38. What conferences are you hoping to attend this year? Any keynote speeches you're hoping to catch?

Conferences are great places to network, attend workshops, learn, and grow. So if you're planning to stick to a [career in artificial intelligence](#), you should be going to some of these. For example, [Deep Learning World](#) has a great one every summer.

This year's event in Las Vegas will feature keynote speakers like [Dr. Dyann Daley](#) (founder and CEO of Predict Align Prevent), [Siddha Ganju](#) (solutions architect at Nvidia), and [Dr. Alex Glushkovsky](#) (principal data scientist at BMO Financial Group, and others).

39. How is Google training data for self-driving cars?

If you're interested and heavily involved within this space, this question should be a no-brainer. If you know the answer, it'll demonstrate your knowledge about a variety of ML methods and how ML is applied

to autonomous vehicles. But even if you don't know the answer, take a stab at it as it will show your creativity and inventive nature.

Google has been using [reCAPTCHA](#) to source labeled data on storefronts and traffic signs for many years now. The company also has been using training data collected by [Sebastian Thrun](#), CEO of the Kitty Hawk Corporation and the co-founder (and former CEO) of Udacity.

Such information, although it might not seem significant, will show a potential employer that you're interested and excited about this field.

More reading: [Google X: Leveraging data and algorithms for self-driving cars](#)

40. Where do you usually source your data sets?

If you talk about AI projects that you've worked on in your free time, the interviewer will probably ask where you sourced your data sets. If you're genuinely passionate about the field, you would have worked on enough projects to know where you can find [free data sets](#).

For example, here are some freely available public data sets that you should know about (without conducting a Google search):

- [CelebFaces](#) (with 200,000 celebrity images along with 40 attribute annotations)
- [CIFAR](#) (with 60,000 images that map 10 different classes)
- [YouTube-8M](#) (with over 4,000 annotated entities taken from an enormous data set of YouTube videos)

Researchers have released hundreds of free resources like these along with the actual network architecture and weights used in their examples. So it will serve you well to explore some of these data sets and run some experiments before heading out for an AI interview.

More reading: [How to find datasets for Artificial Intelligence training](#)

Are you interested in more career-focused AI training? Find out more about [Springboard's Machine Learning Engineering Career Track](#), the first of its kind to come with a job guarantee.

