ML / Deep learning:

1. **What is the difference between supervised and unsupervised learning? Can you provide examples of each?**
   Supervised: Regression, Classification.
   unsupervised: Clustering.

2. **Can you explain the bias-variance tradeoff in machine learning? How do you address overfitting and underfitting?**

   High-bias: an underfitting, too simple model.

   Bias: error → linear regression model→ when trying to fit nonlinear relationship

   High-variance: overfitting → complex model.

   Regularization→ adds a penalty term to the loss function of the model to discourage overfitting.

   Ensemble models: like bagging and boosting, which combine multiple models to reduce the variance of the prediction.

3. **What is regularization in machine learning? How does it help prevent overfitting?**

   Adds penalty term to the model's function,

L1(lasso): add a penalty term to the sum of absolute values of the model's parameters → zero → remove outliers

L2(Ridge): add a penalty term to the sum of squared values of the model parameters

4. **What is the difference between classification and regression problems in machine learning?**

   Classification: Discrete categorical label.

   Regression: Continuous numerical values.

5. **Can you explain the concept of deep learning? How is it different from traditional machine learning?**

   Extract features.

6. **Can you describe the structure of a typical neural network used in deep learning? How does it work?**
   Input, layers, output.

   Training: Adjust the weights of neurons.

   Activation function: Introduce non-linearity, allowing ot to learn more complex relationships between the input and the output.

7. **Activation functions:**
    - **Sigmoid function:** maps input to 0 and 1→ with vanishing gradient problem.
    - **Tanh function:** maps input to -1 and 1, which can be used for binary classification. → with vanishing gradient problem.
    - **Relu function:** [input if positive otherwise 0], dying Relu problem.
    - **Leaky Relu.**
    - **Softmax function:** outputs probability distribution.

8. **What is backpropagation in neural networks? How does it help improve the accuracy of a model?**
    1. Forward pass: output,
    2. Calculate the difference between input and output.
    3. Propagated back through the network to update the weights and biases of the neurons.
    4. Backpropagation: compute the gradient to update the weights and biases through optimization.

    Backpropagation problems:

    - Vanishing Gradients: sigmoid, Tanh
    - Exploding Gradients: Relu.
    - Overfitting: memorizing the noise or outliers.
    - Local minima: where the gradient is close to zero, use momentum and adaptive learning rates.
    - Computational cost: batch normalization, weight sharing, and parallelization.

9. **Can you explain the difference between convolutional neural networks (CNNs) and recurrent neural networks (RNNs)? When would you use one over the other?**

   CNN: convolutional layers to extract features from input data, the weights in the CL are shared across all the neurons, allowing the network to detect the same features in different parts of the input. Then the output is fed to one or more fully connected layers to make the final predictions.

   RNN: using for sequential data.

10. **Can you describe the process of training a deep learning model? What are some common challenges in this process?**

11. Can you discuss some applications of deep learning in the industry? How are companies using it to solve real-world problems?

12. What is transfer learning in deep learning? Can you provide an example of how it is used in practice?

13. What is the vanishing gradient problem in deep learning? How can it be addressed?

14. Can you explain the concept of generative adversarial networks (GANs)? How do they work?

15. What is the difference between a convolutional layer and a pooling layer in a CNN?

16. Can you explain the concept of dropout in deep learning? How does it help prevent overfitting?

17. Can you discuss some common techniques for preprocessing data in machine learning? Why are they important?

18. Can you explain the difference between precision and recall in classification problems? How are they related to the concept of a confusion matrix?

19. Can you describe the process of hyperparameter tuning? How do you decide which hyperparameters to adjust and by how much?

20. Can you discuss some common evaluation metrics used in machine learning? How do you choose which metric to use for a particular problem?

21. Can you explain the concept of ensemble learning in machine learning? How does it help improve the accuracy of a model?

21. What is reinforcement learning? Can you provide an example of how it is used in practice?

22. Can you explain the difference between online and batch learning in machine learning? When would you use one over the other?

23. Can you describe the process of feature engineering in machine learning? Why is it important?

24. Can you explain the difference between a decision tree and a random forest in machine learning? When would you use one over the other?

25. Can you discuss some common challenges in working with unstructured data in machine learning? How can they be addressed?

26. Can you describe the process of data augmentation in deep learning? How does it help improve the accuracy of a model?

27. Can you explain the concept of transfer learning in machine learning? How does it work?

28. Can you discuss some common techniques for reducing the dimensionality of data in machine learning? Why are they important?

29. Can you explain the difference between a linear and a nonlinear model in machine learning? When would you use one over the other?

30. Can you describe the process of clustering in unsupervised learning? Can you provide an example of how it is used in practice?

31. Can you explain the difference between L1 and L2 regularization in machine learning? How do they affect the model?

32. Can you discuss some common techniques for handling missing data in machine learning? How do they work?

33. Can you explain the concept of transfer learning in computer vision? Can you provide an example of how it is used in practice?

34. Can you describe the process of word embedding in natural language processing (NLP)? How does it work?

35. Can you explain the concept of attention in deep learning? How does it help improve the accuracy of a model?

36. Can you discuss some common techniques for dealing with imbalanced datasets in machine learning? How do they work?

37. Can you explain the concept of autoencoders in deep learning? How do they work?

38. Can you describe the process of cross-validation in machine learning? Why is it important?

39. Can you explain the difference between a local and a global minimum in optimization problems? How do you address the issue of getting stuck in a local minimum?

40. Can you discuss some challenges in scaling machine learning models to large datasets or distributed systems? How can they be addressed?

OCR:

1. Can you explain what OCR is and how it works? What are some common challenges in OCR?

2. Can you discuss the difference between handwriting recognition and printed text recognition in OCR? How are they different?

3. Can you describe the process of training an OCR model? What are some common techniques used in training?

4. Can you explain the difference between character-level and word-level recognition in OCR? When would you use one over the other?

5. Can you discuss some common techniques for preprocessing images in OCR? Why are they important?

6. Can you explain the concept of feature extraction in OCR? What are some common techniques used for feature extraction?

7. Can you describe the process of segmentation in OCR? How does it help improve the accuracy of a model?

8. Can you explain the difference between thresholding and binarization in OCR? How are they used in practice?

9. Can you discuss some common techniques for handling noise in OCR? How do they work?

10. Can you explain the concept of post-processing in OCR? What are some common techniques used for post-processing?

11. Can you describe the process of evaluating an OCR model? What are some common metrics used for evaluation?

12. Can you discuss some common challenges in working with OCR for non-Latin scripts, such as Chinese or Arabic? How can they be addressed?

13. Can you explain the difference between rule-based and machine learning-based approaches in OCR? When would you use one over the other?

14. Can you describe the process of building an end-to-end OCR system? What are some common components of such a system?

15. Can you discuss some common techniques for improving the accuracy of an OCR model over time? How are they used in practice?

16. Can you explain the concept of deep learning-based OCR? How is it different from traditional OCR approaches?

17. Can you describe the process of training a deep learning-based OCR model? What are some common challenges in this process?

18. Can you discuss some common techniques for handling multi-language OCR? How do they work?

19. Can you explain the difference between offline and online OCR? When would you use one over the other?

20. Can you discuss some common applications of OCR in industry? How are companies using it to solve real-world problems?

YOLO:

YOLO (You Only Look Once) is a state-of-the-art object detection system that uses a single neural network to predict bounding boxes and class probabilities directly from full images in real-time.

Here are some interview questions about YOLO:

1. Can you explain how YOLO works? What are some advantages of using YOLO for object detection?

2. Can you discuss the difference between YOLOv1, YOLOv2, YOLOv3, and YOLOv4? What are some improvements in each version?

3. Can you explain the concept of anchor boxes in YOLO? How do they help improve the accuracy of a model?

4. Can you describe the process of training a YOLO model? What are some common challenges in this process?

5. Can you discuss some common techniques for improving the accuracy of a YOLO model? How are they used in practice?

6. Can you explain the difference between two-stage and one-stage object detection systems? How is YOLO different from other one-stage systems?

7. Can you describe the process of implementing YOLO on embedded devices or mobile phones? What are some common challenges in this process?

8. Can you discuss some common techniques for handling occlusion in YOLO? How do they work?

9. Can you explain the concept of transfer learning in YOLO? How is it used in practice?

10. Can you discuss some common applications of YOLO in industry? How are companies using it to solve real-world problems?

11. Can you explain the difference between YOLO and other popular object detection systems such as Faster R-CNN and SSD? When would you use one over the other?

12. Can you describe the process of fine-tuning a pre-trained YOLO model? What are some common techniques used in fine-tuning?

13. Can you discuss some common challenges in working with YOLO for small object detection? How can they be addressed?

14. Can you explain the concept of non-maximum suppression (NMS) in YOLO? How does it help improve the accuracy of a model?

15. Can you describe the process of evaluating a YOLO model? What are some common metrics used for evaluation?

Transformers:

1. **Can you explain what a transformer is and how it works? What are some advantages of using transformers over other types of neural networks?**

- NN architecture, and is used for NLP and CV, such as machine translation, language modeling, and text classification.
- Uses self-attention mechanisms, and can process variable-length sequences of tokens.
- Handle long-large dependencies.
- Ability to learn from context.

2. **Can you discuss the difference between self-attention and multi-head attention in a transformer? How are they used in practice?**

3. **Can you describe the process of training a transformer model? What are some common challenges in this process?**
    - Initialize model weights
    - Feeding input data
    - Computing loss function
    - Updating the weights using backpropagation.

4. **Can you explain the concept of positional encoding in a transformer? How does it help the model handle variable-length sequences?**
    - It encodes the position of each token in the input sequence.
    - Helps handle the variable length sequences by providing information about the relative position of each token.

5. **Can you discuss some common techniques for improving the accuracy of a transformer model? How are they used in practice?**

6. **Can you explain the difference between encoder-only and encoder-decoder transformer architectures? When would you use one over the other?**

   encoder -only: the output is fixed-length vector representing the input sequence. → BERT

   Cannot generate text (only understand the text)

   Encoder-decoder: the output is a sequence of tokens generated based on an input sequence.

7. **Can you describe the process of fine-tuning a pre-trained transformer model? What are some common techniques used in fine-tuning?**
8. **Can you discuss some common applications of transformers in the industry? How are companies using them to solve real-world problems?**
9. **Can you explain the concept of transfer learning in transformers? How is it used in practice?**
10. **Can you describe the process of implementing a transformer model on embedded devices or mobile phones? What are some common challenges in this process?**
11. **Can you discuss some common challenges in working with transformers for non-Latin scripts, such as Chinese or Arabic? How can they be addressed?**

    Complex morphology

12. **Can you explain the difference between BERT and GPT-2? When would you use one over the other?**

    BERT is an encoder-only mechanism, that cannot generate text.

13. **Can you describe the process of building an end-to-end transformer-based NLP system? What are some common components of such a system?**
14. **Can you discuss some common techniques for handling long sequences in transformers? How do they work?**
15. **Can you explain the concept of hierarchical transformers? How are they used in practice?**
16. **Can you describe the process of evaluating a transformer model? What are some common metrics used for evaluation?**
17. **Can you explain the difference between a transformer and a recurrent neural network (RNN)? When would you use one over the other?**

Transformer model components:

1. **Input Embeddings:** this is done using a trainable lookup table.
2. **Positional Encoding**.
3. **Encoder:** multi-head self-attention mechanism + position-wise fully connected feed-forward network.
4. **Multi-Head Attention:** type of self-attention allows the model to attend to different parts of the input sequence simultaneously.
5. Feed-Forward Network.
6. Layer Normalization.
7. Decoder.
8. Masked Multi-Head Attention.

optimizers

1. What is an optimizer in machine learning?
2. What are the most commonly used optimization algorithms in deep learning?
3. What is the difference between gradient descent and stochastic gradient descent?
4. What is momentum in optimization algorithms?
5. How does the learning rate affect the optimization process?
6. How do you choose the learning rate for an optimizer?
7. What is the role of regularization in optimization algorithms?
8. What is the difference between L1 and L2 regularization?
9. What is the Adam optimizer and how does it work?
10. How do you choose the hyperparameters for an optimizer?

11. What are the advantages and disadvantages of using different optimization algorithms?
12. How do you handle the problem of vanishing gradients in deep learning?
13. What is the role of batch normalization in optimization?
14. What is the difference between batch gradient descent and mini-batch gradient descent?
15. What is the concept of adaptive learning rates in optimization algorithms, and how does it work?

Gradient Descent

1. What is gradient descent, and how does it work?
2. What are the types of gradient descent algorithms?
3. What is the difference between batch gradient descent, stochastic gradient descent, and mini-batch gradient descent?
4. What is the role of learning rate in gradient descent, and how do you choose the appropriate learning rate?
5. What is the impact of the learning rate on the convergence of the gradient descent algorithm?
6. How do you handle the problem of overshooting and undershooting in gradient descent?
7. What is the role of momentum in gradient descent, and how does it help in convergence?
8. What is the difference between gradient descent and Newton's method?
9. What is the role of regularization in gradient descent?
10. How do you handle the problem of local minima in gradient descent?
11. What is the difference between convex and non-convex optimization problems in gradient descent?
12. What is the impact of the batch size on the convergence of the gradient descent algorithm?
13. How do you choose the appropriate batch size for the gradient descent algorithm?
14. How do you handle the problem of vanishing gradients in deep learning?
15. What is the impact of the initialization of the weights on the convergence of the gradient descent algorithm?