|  |  |
| --- | --- |
| 1 | Why don't we start all of the weights with zeros? |
| Ans | Starting all of the weights with zeros does not work well in practice because the same weights are applied to all inputs. This can cause the model to become stuck in a local minimum, meaning the model fails to find the optimal solution. Additionally, the weights need to be initialized randomly in order for the model to explore a wide range of solutions in order to find the optimal solution. |
| 2 | Why is it beneficial to start weights with a mean zero distribution? |
| Ans | Starting weights with a mean zero distribution helps to avoid any unintentional bias caused by the weights. If the weights are initialized with a mean that is not zero, then the model could be unintentionally biased towards certain outputs, meaning it could be less accurate than it should be. Additionally, starting weights with a mean zero distribution helps to ensure that changes in the weights during training are symmetrical. This helps to avoid any ‘dead-ends’ where the weights are locked in a non-optimal configuration. |
| 3 | What is dilated convolution, and how does it work? |
| Ans | Dilated convolution is a type of convolution in which the filter used has a higher stride than usual. This allows the filter to cover a larger area of the input image and capture more information. The filter is also given a “dilation factor” which increases the size of the filter so that it can capture more information from the input image. The dilation factor is usually defined as the ratio between the stride and the filter size. This means that the filter will “see” more of the input image and thus be able to capture more information. This is especially useful for image segmentation and object detection tasks. |
| 4 | What is TRANSPOSED CONVOLUTION, and how does it work? |
| Ans | Transposed convolution, also known as deconvolution, is a type of convolution operation that reverses the forward convolution process. It is the same as a regular convolution operation, but with the weights and inputs swapped. The output of a transposed convolution is larger than the input, and it is used to upsample an image. Transposed convolution works by first calculating the output size of the operation and then computing the weights of the operation. The output is then computed by taking the dot product of the weights and the input. By reversing the forward convolution process, transposed convolution can be used to increase the size of an image. |
| 5 | Explain Separable convolution |
| Ans | Separable convolutions are a type of convolution where the input is broken down into two separate convolutional operations. The first operation is a standard 2D convolution where the input is convolved with a 2D filter. The output of this convolution is then convolved with a 1D filter. This approach is used to reduce the computational cost of convolutions. Separable convolutions can also be used to reduce the number of parameters in a model, resulting in a simpler and more efficient model. Separable convolutions are especially useful for deep learning models where the number of parameters can quickly become unmanageable. |
| 6 | What is depthwise convolution, and how does it work? |
| Ans | Depthwise convolution is a type of convolutional neural network (CNN) operation where a single filter is applied to each input channel separately. This can be used to reduce the model complexity as well as to increase the performance of the model. It works by applying the same filter across all of the channels in an input image, meaning that instead of having one kernel applied across all channels in a single convolutional layer, each channel has its own kernel. This allows the model to learn different patterns in each channel and can lead to improved performance. |
| 7 | What is Depthwise separable convolution, and how does it work? |
| Ans | Depthwise separable convolution is a type of convolution where the input is first split into different channels and then a convolution operation is applied to each channel separately. This is different from regular convolution where the same convolution operation is applied to the entire input at once. The benefit of depthwise separable convolution is that it requires fewer parameters, which helps reduce computation time and requires less memory. Additionally, since each channel is processed separately, it helps to reduce the risk of overfitting. |
| 8 | Capsule networks are what they sound like. |
| Ans | They are convolutional neural networks (CNNs) with capsules instead of neurons. The capsules are specialized units that are designed to capture and represent spatial relationships between input data. They enable the network to learn more complex patterns and representations than a traditional CNN. CNNs are great for recognizing patterns, but they struggle with spatial relationships. Capsule networks are better at capturing those relationships and are better at recognizing objects in a scene. |
| 9 | Why is POOLING such an important operation in CNNs? |
| Ans | Pooling is an important operation in CNNs because it reduces the size of the data while extracting the most meaningful features from the data. Pooling also helps to reduce the number of parameters and computations in the network, which in turn reduces the training time and allows for faster inference. Pooling also helps to reduce overfitting, as it reduces the complexity of the model without sacrificing accuracy. |
| 10 | What are receptive fields and how do they work? |
| Ans | Receptive fields are regions in an image that a neuron can detect. They are the areas of the image that the neuron is sensitive to and will respond to when presented with an input. Receptive fields are determined by the size and shape of a filter used in a convolutional neural network (CNN). Each neuron in a CNN has a specific receptive field, and the neurons are arranged in a grid pattern to detect different parts of an image. The size and shape of the receptive fields determine the type of features that the neural network can detect. |