**Assignment - 02**

1. Explain convolutional neural network, and how does it work?

Ans: Explanation of Convolutional Neural Network (CNN) and how it works:

A Convolutional Neural Network (CNN) is a type of deep learning model designed for processing structured grid-like data, such as images.

CNNs consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers.

In a CNN, convolutional layers apply filters (kernels) to input images to extract features hierarchically. These filters slide over the input image, performing element-wise multiplication and summing up the results to produce feature maps.

Pooling layers downsample the feature maps, reducing their spatial dimensions while preserving important information. Common pooling operations include max pooling and average pooling.

Fully connected layers at the end of the network perform classification based on the features extracted by previous layers.

CNNs leverage parameter sharing and local connectivity to efficiently learn spatial hierarchies of features in images, making them highly effective for tasks such as image classification, object detection, and image segmentation.

1. How does refactoring parts of your neural network definition favor you?

Ans: Refactoring parts of a neural network definition improves code readability, modularity, and maintainability.

It allows for better organization of the network architecture, making it easier to debug and modify.

Refactoring facilitates code reuse and abstraction, enabling faster development and experimentation with different architectures and configurations.

3. What does it mean to flatten? Is it necessary to include it in the MNIST CNN? What is the reason for this?

Ans: Flattening refers to converting multi-dimensional input data into a one-dimensional vector.

In the MNIST CNN, flattening is necessary to transform the output of the convolutional and pooling layers into a format that can be fed into the fully connected layers for classification.

Since the fully connected layers expect one-dimensional input, flattening is essential to maintain the sequential structure of the data and ensure compatibility with the subsequent layers.

4. What exactly does NCHW stand for?

Ans: NCHW stands for "Number of samples (N), Channels (C), Height (H), and Width (W)."

It is a common data format used to represent multi-dimensional arrays or tensors in deep learning frameworks, such as PyTorch and TensorFlow.

In this format, the dimensions of the tensor are ordered as follows: batch size (N), number of channels (C), height (H), and width (W).

5. Why are there 7\*7\*(1168-16) multiplications in the MNIST CNN's third layer?

Ans: Calculation of multiplications in the MNIST CNN's third layer:

The calculation involves the number of multiplications required to compute the output feature map of the convolutional layer.

In the MNIST CNN's third layer, there are 7x7 filters applied to feature maps with a depth of (1168-16) channels.

The number of multiplications is calculated as the product of the filter size, input depth, and output depth, considering the parameters of the convolutional operation.

6.Explain definition of receptive field?

Ans: Receptive field refers to the region of input space that a particular feature in a neural network is sensitive to.

It represents the area of the input data that contributes to the activation of a specific neuron in the network.

Receptive fields can vary in size and shape across different layers of the network, depending on the architecture and parameters of the convolutional and pooling operations.

7. What is the scale of an activation's receptive field after two stride-2 convolutions? What is the reason for this?

Ans: After two stride-2 convolutions, the receptive field of an activation increases exponentially.

Each stride-2 convolution effectively doubles the receptive field by skipping alternate pixels in both dimensions.

As a result, the receptive field grows much larger compared to the original input size, allowing the network to capture global spatial dependencies and contextual information.

8. What is the tensor representation of a color image?

Ans: A color image is represented as a 3-dimensional tensor, typically in the format (Height x Width x Channels).

Each channel represents a color component (Red, Green, and Blue), and the height and width dimensions correspond to the spatial dimensions of the image.

9. How does a color input interact with a convolution?

Ans: In a convolutional operation applied to a color input, each filter convolves with each color channel separately.

The resulting feature maps are then summed across all channels to produce the output feature map.

Convolutional filters learn spatial patterns and features independently across each color channel, allowing the network to extract rich representations from color images.