Various Neural Network Architect Overview Assignments

Q1. Describe the basic structure of a Feedforward Neural Network (FNN). What is the purpose of the activation function?

ANS:

A Feedforward Neural Network (FNN) is a type of artificial neural network where the connections between the nodes do not form cycles. Information moves in one direction—from the input layer, through the hidden layers (if any), and finally to the output layer.

Input Layer: The input layer takes the features from the data. Hidden Layers: Intermediate layers where computation is performed. Each neuron in the hidden layers applies a weight to the input and passes it through an activation function. Output Layer: Produces the final prediction or classification. Purpose of the Activation Function:

The activation function introduces non-linearity into the network, allowing the model to capture complex patterns and relationships in the data. Without it, no matter how many layers the network has, the output would be a linear transformation of the input. Common activation functions include ReLU, Sigmoid, and Tanh.

Q2. Explain the role of convolutional layers in CNN. Why are pooling layers commonly used, and what do they achieve?

ANS:

In Convolutional Neural Networks (CNNs), the convolutional layers perform feature extraction by applying filters (or kernels) to the input image or feature map.

Convolution Operation: Each filter slides over the input and performs element-wise multiplication, producing feature maps that highlight various aspects of the input (such as edges, textures, or patterns). Role: Convolutional layers enable the network to learn spatial hierarchies, recognizing features like edges, corners, and complex objects at various levels of abstraction. Pooling Layers:

Pooling layers are used to reduce the spatial dimensions of the feature maps, leading to fewer parameters and computations, which makes the model more efficient and reduces overfitting. Max Pooling: Takes the maximum value from a region of the feature map. Average Pooling: Takes the average value from a region of the feature map. Pooling layers help downsample the feature map while retaining the most important information.

Q3. What is the key characteristic that differentiates Recurrent Neural Networks (RNNs) from other neural networks? How does an RNN handle sequential data?

ANS:

The key characteristic that differentiates Recurrent Neural Networks (RNNs) from other neural networks is their ability to handle sequential data. In RNNs, the output of a neuron is fed back into the network as input to the next step. This feedback loop enables RNNs to retain information from previous steps, making them effective for time-series data, natural language processing, and other sequential tasks.

How RNN Handles Sequential Data:

At each time step, the RNN processes the current input along with the hidden state from the previous time step. The hidden state acts as a memory, storing information about previous inputs in the sequence. This allows RNNs to capture dependencies and patterns in sequential data.

Q4. Discuss the components of a Long Short-Term Memory (LSTM) network. How does it address the vanishing gradient problem?

ANS:

LSTM is a special type of RNN designed to overcome the vanishing gradient problem and retain long-term dependencies in sequential data. LSTMs contain multiple gates that regulate the flow of information:

Forget Gate: Decides which parts of the previous memory to forget. Input Gate: Decides which new information to store in the memory. Cell State: The memory of the network, which is updated by the input and forget gates. Output Gate: Determines what to output at the current time step, based on the cell state. How LSTM Addresses the Vanishing Gradient Problem:

By using gates, LSTMs can control the flow of information and gradients more effectively, preventing the gradients from becoming too small (or "vanishing") over long sequences. This allows the model to learn long-term dependencies.

Q5. Describe the roles of the generator and discriminator in a Generative Adversarial Network (GAN). What is the training objective for each?

ANS:

Generative Adversarial Networks (GANs) consist of two competing networks: the generator and the discriminator.

Generator: The generator takes random noise as input and attempts to create fake data that resemble the real data.

Objective: Fool the discriminator into believing that the generated (fake) data is real. Training Goal: Minimize the discriminator’s ability to correctly classify fake data by improving the

quality of generated data over time. Discriminator: The discriminator takes real or fake data as input and classifies whether the data is real or fake.

Objective: Correctly distinguish between real data and fake data generated by the generator. Training Goal: Maximize its ability to identify real data from fake data. Training Objective:

The generator tries to minimize the loss function by generating better fake data, while the discriminator tries to maximize the loss by correctly identifying fake data. This adversarial process pushes both networks to improve over time.