(a) Explain how you can implement DL in a real-world application.

(b) What is the use of Activation function in Artificial Neural Networks? What would be the problem if we don't use it in ANN networks.

Answer :-

Problem Definition: Clearly define the problem you want to solve and identify how DL can help address it. This could involve tasks such as image classification, natural language processing, object detection, or time series forecasting.

Data Collection and Preparation: Gather relevant data for your problem domain. This could include structured data from databases, unstructured data from text documents or images, or time series data. Clean and preprocess the data to remove noise, handle missing values, and scale features if necessary.

Model Selection: Choose appropriate DL architectures and models based on your problem requirements. Common DL architectures include Convolutional Neural Networks (CNNs) for image-related tasks, Recurrent Neural Networks (RNNs) for sequential data, and Transformer models for natural language processing tasks.

Model Training: Split the data into training, validation, and test sets. Train the selected DL model using the training data and optimize its hyperparameters to achieve better performance. This involves adjusting parameters like learning rate, batch size, and regularization techniques.

Model Evaluation: Evaluate the trained model using the validation set to assess its performance metrics such as accuracy, precision, recall, or F1-score, depending on the problem domain. Fine-tune the model if necessary based on the evaluation results.

Deployment: Deploy the trained DL model into production, either on-premises or in the cloud. Integrate the model into your application or workflow to make predictions on new, unseen data. Monitor the model's performance and retrain it periodically to maintain accuracy and adapt to changing data distributions.

Scalability and Optimization: Ensure that the deployed DL model is scalable and optimized for efficient inference. This may involve techniques like model quantization, pruning, or deploying the model on specialized hardware such as GPUs or TPUs for faster computation.

Maintenance and Updates: Continuously monitor the performance of the deployed DL model and make updates or improvements as needed. This could involve retraining the model with new data, fine-tuning hyperparameters, or incorporating feedback from users or domain experts.

(b) Activation functions are crucial components of Artificial Neural Networks (ANNs) as they introduce non-linearity into the network, enabling it to learn complex patterns and relationships in the data. The activation function decides whether a neuron should be activated (output a signal) or not based on the input it receives.

The main uses of activation functions in ANNs are:

Introduction of Non-Linearity: Activation functions introduce non-linear transformations to the output of a neuron. Without non-linear activation functions, the entire network would behave like a single-layer perceptron, resulting in a limited ability to learn complex patterns from the data.

Enable Network to Learn Complex Patterns: By introducing non-linearity, activation functions enable ANNs to learn and represent complex relationships and patterns in the data. This is crucial for tasks such as image recognition, natural language processing, and many other real-world applications.

If we don't use activation functions in ANNs, the network would essentially be a linear model, and the entire network would collapse into a single linear transformation. This would severely limit the expressiveness and learning capacity of the network, rendering it unable to learn complex patterns and relationships in the data. As a result, the network's performance would be poor, especially on tasks that require modeling non-linear relationships. Therefore, activation functions are essential components of ANNs and play a crucial role in their ability to learn from data and make accurate predictions.

Q.2