19EE701 - AI TECHNIQUES IN ELECTRICAL ENGINEERING

UNIT 2 - NEURAL NETWORKS

Introduction to Artificial Neural Networks (ANNs)

1. Overview of Artificial Neural Networks

1.1. Definition and Motivation

- **Artificial Neural Networks (ANNs)**: Computational models designed to simulate the way the human brain processes information. They are used for tasks like classification, regression, and pattern recognition.
- **Inspiration**: ANNs are inspired by biological neural networks in the brain, where neurons are interconnected and transmit signals.

1.2. Key Concepts

- **Neurons**: Basic units of ANNs, similar to biological neurons. They receive input, process it, and pass it on to the next layer.
- **Weights and Biases**: Weights determine the strength of connections between neurons, while biases help adjust the output of neurons to better fit the data.

2. Structure of ANNs

2.1. Layers of a Neural Network

- **Input Layer**: The first layer that receives raw data. Each neuron in this layer represents a feature of the data.
- **Hidden Layers**: Intermediate layers between input and output. They perform computations and feature extraction. The number of hidden layers and neurons can vary.
- **Output Layer**: Produces the final output, such as classification labels or numerical values.

2.2. Neuron Model

- **Inputs**: Each neuron receives inputs, which are multiplied by corresponding weights.
- **Weighted Sum**: The sum of weighted inputs and a bias term.

3. Training ANNs

3.1. Forward Propagation

- Data is passed through the network from the input layer to the output layer.
- Each neuron applies its activation function to the weighted sum of inputs to produce an output.

3.2. Loss Function

- Measures the difference between the network's prediction and the actual target values.
- Common loss functions:
    - **Mean Squared Error (MSE)**: Used for regression tasks.

- **Cross-Entropy Loss**: Used for classification tasks.

3.3. Backpropagation

- Algorithm for updating the weights and biases to minimize the loss function.
- **Gradient Descent**: Optimization technique used to find the minimum of the loss function by adjusting weights in the opposite direction of the gradient.
  - **Learning Rate**: Determines the step size in the weight update process.

3.4. Training Process

1. **Initialization**: Start with random weights and biases.
2. **Forward Pass**: Compute output and loss.
3. **Backward Pass**: Compute gradients of the loss function with respect to weights and biases.
4. **Update Weights**: Adjust weights and biases using gradients.
5. **Iteration**: Repeat the process until convergence or for a specified number of epochs.

4. Types of Neural Networks

4.1. Feedforward Neural Networks (FNNs)

- Simplest type where connections do not form cycles.
- Data moves in one direction, from input to output.

4.2. Convolutional Neural Networks (CNNs)

- Designed for spatial data, such as images.
- Use convolutional layers to detect features like edges and textures.

4.3. Recurrent Neural Networks (RNNs)

- Suitable for sequential data, like time series or text.
- Have connections that form cycles, allowing them to maintain context across different time steps.

5. Applications of ANNs

5.1. Image Recognition

- Identifying objects, faces, and scenes in images.

5.2. Speech Recognition

- Converting spoken language into text.

5.3. Natural Language Processing (NLP)

- Understanding and generating human language.

5.4. Predictive Analytics

- Forecasting future trends based on historical data.

6. Challenges and Considerations

6.1. Overfitting

- When the network learns the training data too well, including noise, reducing its performance on new data.
- Mitigation techniques: Regularization, dropout, and cross-validation.

6.2. Computational Resources

- Training deep networks can be computationally expensive and time-consuming.

6.3. Interpretability

- Neural networks can be complex and difficult to interpret. Research in Explainable AI (XAI) aims to make these models more transparent.

7. Future Directions

7.1. Deep Learning

- Advances in networks with many layers, improving accuracy and capabilities.

7.2. Transfer Learning

- Reusing pre-trained models for new, related tasks to reduce training time and improve performance.

7.3. Explainable AI (XAI)

- Developing methods to make neural networks more understandable and transparent.