

## ◆ UNIT 1: Neural Networks & Backpropagation

### 1. Human Brain

- Biological neurons → Artificial Neurons
- Dendrites = Inputs
- Cell body = Processing (activation)
- Axon = Output

### 2. Model of Neuron

$$y = f(\sum w_i x_i + b)$$

- Weights (w)
- Bias (b)
- Activation function (ReLU, Sigmoid, Tanh)

### 3. Perceptron

- Single-layer binary classifier
- Works only for linearly separable data

### 4. Multilayer Perceptron (MLP)

- Input + Hidden + Output layers
- Solves non-linear problems

### 5. Backpropagation

- Uses gradient descent
- Minimizes loss using chain rule
- Steps: Forward pass → Loss → Backward pass → Update weights

### 6. XOR Problem

- Not linearly separable
- Requires hidden layer

### 7. Improve Backprop

- Learning rate tuning
- Momentum

- Weight initialization
  - Normalization
- 

## ◆ UNIT 2: Optimization & Regularization

### 1. Gradient Descent

$$\theta = \theta - \eta \nabla J(\theta)$$

Types:

- Batch GD
- Stochastic GD (SGD)
- Mini-batch GD

### 2. Momentum

- Accelerates learning
- Reduces oscillation

### 3. RMSProp

- Adaptive learning rate
- Uses squared gradient average

### 4. Adam

- Combines Momentum + RMSProp
- Most popular optimizer

### 5. Regularization

- L2 (Weight decay)
- Dropout
- Early stopping

### 6. Dataset Augmentation

- Rotate, flip, crop images

### 7. Semi-Supervised Learning

- Uses small labeled + large unlabeled data
-

## ◆ UNIT 3: CNN

### 1. Convolution

- Extracts features
- Uses filters/kernels

### 2. Padding

- Same / Valid

### 3. Stride

- Step size of filter

### 4. Pooling

- Max pooling
- Average pooling

### 5. Normalization

- Batch Normalization

### 6. Architectures

- LeNet
- AlexNet
- VGG
- GoogLeNet (Inception)
- ResNet (Skip connections)
- EfficientNet

### 7. Transfer Learning

- Use pretrained model
- 

## ◆ UNIT 4: Sequence Models

### 1. RNN

- Handles sequential data
- Has memory

### 2. Vanishing Gradient

- Gradients become very small

### **3. LSTM**

- Cell state + Gates
- Solves vanishing gradient

### **4. GRU**

- Simpler than LSTM

### **5. Bidirectional RNN**

- Forward + Backward pass

### **Applications**

- Text generation
  - Sentiment analysis
  - Time series forecasting
- 

## **◆ UNIT 5: Attention & Transformers**

### **1. Attention**

- Focus on important words

### **2. Self-Attention**

- Attention within same sequence

### **3. Multi-Head Attention**

- Multiple attention mechanisms

### **4. Encoder–Decoder**

- Used in translation

### **5. Autoencoders**

- Basic AE
- Sparse AE
- Denoising AE
- VAE

### **6. Transformer**

- No RNN
- Uses self-attention

### **7. BERT**

- Encoder-based
- Bidirectional

### **8. GPT**

- Decoder-based
- Text generation

---

## **◆ UNIT 6: Generative Models**

### **1. GAN**

- Generator
- Discriminator

### **2. DCGAN**

- CNN-based GAN

### **3. CycleGAN**

- Image-to-image translation

### **4. StyleGAN**

- High-quality face generation

### **Applications**

- Image generation
- Deepfake
- Super resolution

**UNIT 1-2**

1. Perceptron solves?  
A) Non-linear  
B) Linear  
C) Regression  
D) Clustering  
**Ans: B**
2. XOR requires?  
A) Single layer  
B) Hidden layer  
C) No weights  
D) Linear model  
**Ans: B**
3. Backprop uses?  
A) Chain rule  
B) Bayes  
C) PCA  
D) SVM  
**Ans: A**
4. Adam combines?  
A) SGD + RMSProp  
B) Momentum + RMSProp  
C) L2 + Dropout  
D) CNN + RNN  
**Ans: B**
5. L2 regularization reduces?  
A) Loss  
B) Bias  
C) Overfitting  
D) Accuracy  
**Ans: C**
6. SGD updates using?  
A) Full data  
B) One sample  
C) No data  
D) Batch average  
**Ans: B**

**UNIT 4**

16. RNN is used for?  
A) Images  
B) Sequences  
C) Clustering  
D) PCA  
**Ans: B**
17. LSTM solves?  
A) Overfitting  
B) Vanishing gradient  
C) Regression  
D) CNN  
**Ans: B**
18. GRU has?  
A) 3 gates  
B) 2 gates  
C) No gate  
D) 5 gates  
**Ans: B**
19. Bidirectional RNN reads?  
A) Left only  
B) Right only  
C) Both directions  
D) None  
**Ans: C**
20. Sentiment analysis uses?  
A) CNN only  
B) RNN/LSTM  
C) PCA  
D) SVM  
**Ans: B**

**UNIT 5**

21. Transformer removes?  
A) Attention  
B) RNN  
C) CNN

<p>7. Momentum helps in?  A) Slow learning  B) Oscillation reduction  C) Overfitting  D) Clustering  <b>Ans: B</b></p> <p>8. RMSProp adjusts?  A) Batch size  B) Learning rate  C) Neurons  D) Filters  <b>Ans: B</b></p> <p>9. Activation function adds?  A) Linearity  B) Non-linearity  C) Noise  D) Data  <b>Ans: B</b></p> <p>10. ReLU formula?  A) <math>x^2</math>  B) <math>1/x</math>  C) <math>\max(0,x)</math>  D) <math>\tanh</math>  <b>Ans: C</b></p>	<p>D) Loss  <b>Ans: B</b></p> <p>22. BERT is?  A) Decoder  B) Encoder  C) GAN  D) CNN  <b>Ans: B</b></p> <p>23. GPT is?  A) Encoder  B) Decoder  C) RNN  D) CNN  <b>Ans: B</b></p> <p>24. Attention calculates?  A) Similarity score  B) Loss  C) Gradient  D) Bias  <b>Ans: A</b></p> <p>25. VAE is used for?  A) Classification  B) Generation  C) Sorting  D) Clustering  <b>Ans: B</b></p>
<p><b>UNIT 3 (CNN)</b></p> <p>11. Convolution extracts?  A) Labels  B) Features  C) Bias  D) Loss  <b>Ans: B</b></p> <p>12. Pooling reduces?  A) Channels  B) Parameters  C) Dimension</p>	<p><b>UNIT 6</b></p> <p>26. GAN has?  A) 1 network  B) 2 networks  C) 3 networks  D) 4 networks  <b>Ans: B</b></p> <p>27. Generator creates?  A) Labels  B) Fake data</p>

D) Loss

**Ans: C**

13. ResNet uses?

A) Pooling

B) Skip connections

C) LSTM

D) GAN

**Ans: B**

14. VGG uses?

A) Large filters

B) 3x3 filters

C) RNN

D) GAN

**Ans: B**

15. Padding keeps?

A) Loss

B) Size

C) Filters

D) Bias

**Ans: B**

C) Gradient

D) Loss

**Ans: B**

28. Discriminator acts as?

A) Teacher

B) Judge

C) Generator

D) Encoder

**Ans: B**

29. DCGAN uses?

A) RNN

B) CNN

C) LSTM

D) SVM

**Ans: B**

30. CycleGAN is used for?

A) Classification

B) Translation

C) Clustering

D) Regression

**Ans: B**