

DEEP LEARNING

★ MODULE 1: Introduction to Deep Learning

- What is Deep Learning?
 - Neural Networks vs Machine Learning
 - When to use Deep Learning?
 - DL applications: CV, NLP, Speech
 - Why DL needs GPUs
 - Deep Learning frameworks:
 - PyTorch (primary)
 - TensorFlow/Keras (secondary)
 - Training workflow: dataset → model → loss → optimizer
-

★ MODULE 2: Math Foundations

- Vectors, matrices, tensors
 - Dot product & matrix multiplication
 - Derivatives & gradients
 - Chain rule
 - Loss functions (MSE, Cross-entropy)
 - Optimization algorithms:
 - SGD
 - Momentum
 - RMSprop
 - Adam
-

★ MODULE 3: Artificial Neural Networks (ANNs)

- Perceptron
 - Multi-Layer Perceptron (MLP)
 - Forward & backward propagation
 - Activation functions:
 - ReLU, Sigmoid, Tanh, Softmax
 - Regularization:
 - Dropout
 - L2 regularization
 - Early stopping
 - Batch Normalization
 - Building ANN using:
 - TensorFlow
 - PyTorch
-

Projects

- ANN for classification
 - ANN for regression
-

★ MODULE 4: Convolutional Neural Networks (CNNs) – Core CV

- Convolution operation
- Filters, kernels, stride, padding
- Pooling (MaxPool, AvgPool)
- CNN architecture blocks
- Training & fine-tuning CNNs
- Common CNN architectures:
 - LeNet
 - AlexNet
 - VGG
 - ResNet (VERY IMPORTANT)
 - Inception
 - EfficientNet (intro)
- Image augmentation techniques
- Data preprocessing for images

Projects

- Cats vs Dogs
 - Fashion MNIST classifier
 - Face mask detector
-

★ MODULE 5: Advanced Computer Vision

- Object detection:
 - YOLO (v5 or v8)
 - SSD
 - Faster R-CNN
- Semantic segmentation:
 - U-Net
- Image classification with Transfer Learning
- Feature extraction using pretrained CNNs
- Real-time inference with OpenCV

Projects

- Helmet detection
 - Vehicle detection
 - Medical image segmentation (U-Net)
-

★ MODULE 6: Natural Language Processing WITH Deep Learning (No Transformers)**RNN Family**

- Sequence modeling
- Working with time-series text
- RNNs
- LSTMs
- GRUs

Text Data

- Tokenization
- Embeddings (Word2Vec, GloVe)

- Padding sequences
- Building text classification models
- Encoder–Decoder basics (without attention/transformers)

Projects

- Sentiment analysis using LSTM
- Text classification (news/emails)
- Text generation using RNN/LSTM

★ **MODULE 7: Model Optimization & Training**

Tricks

- Learning rate scheduling
- Model initialization techniques
- Gradient clipping
- Regularization
- Early stopping
- Overfitting / underfitting handling
- Data pipelines and batch loading
- GPU training basics
- Mixed precision training (optional)

★ **MODULE 8: Deep Learning Deployment (Optional)**

- Saving/loading DL models
- Building inference pipelines
- Deployment Frameworks:
 - FastAPI
 - Streamlit
 - Flask
- Deployment Platforms:
 - AWS EC2
 - HuggingFace Spaces
 - Render
- ONNX (basic intro only)

Projects

- Image classifier web app
- LSTM sentiment analysis web app

★ **MODULE 9: Capstone Projects**

✓ **Computer Vision (choose 1)**

- YOLO object detection
- Disease detection from images
- Face recognition
- Image classification using ResNet

✓ **NLP (choose 1)**

- LSTM sentiment classifier
- Text generation using RNN/LSTM

- Sequence-to-sequence text model

★ **DEEP LEARNING TOOLS USED**

Frameworks:

- PyTorch (recommended)
- TensorFlow/Keras

Vision Libraries:

- OpenCV
- torchvision

NLP Libraries:

- NLTK
- SpaCy
- Gensim

Deployment:

- FastAPI
 - Streamlit
 - Flask
-