



Artificial Intelligence (AI) Training Course

CURRICULUM

Module I – Foundations of Artificial Intelligence

- Definition and Scope of AI
- History and Evolution of AI
- Applications of AI in Real Life
- Types of AI: Narrow, General, Super
- Weak AI vs Strong AI
- AI vs Machine Learning vs Deep Learning
- Symbolic AI and Classical Approaches
- Intelligent Agents
- Turing Test and Rational Agents
- Ethics and Social Implications of AI

Module II – Python for AI

- Introduction to Python Programming
- Data Types, Variables, and Operators
- Control Structures and Loops
- Functions and Lambda Expressions
- Lists, Tuples, Dictionaries, Sets
- File Handling in Python
- Exception Handling
- Modules and Packages
- Object Oriented Programming
- Introduction to Virtual Environments

Module III – Numpy for Numerical Computation

- Introduction to NumPy and Its Importance
- Creating and Working with ndarrays
- Array Indexing, Slicing, and Iteration
- Array Operations
- Mathematical and Statistical Functions
- Reshaping, Stacking, and Splitting Arrays
- Linear Algebra Operations (Dot Product, Matrix Inversion, Determinants)
- Random Module – Generating Random Numbers
- Vectorization for Performance Optimization
- Real-world Examples: Image Representation, Computation Tasks

Module IV – Pandas for Data Manipulation

- Introduction to Pandas: Series and DataFrames
- Importing and Exporting Data (CSV, Excel, JSON)
- Indexing, Slicing, and Subsetting Data
- Data Cleaning: Handling Missing Values, Duplicates, Outliers
- Data Transformation: Apply, Map, Lambda Functions
- Filtering and Conditional Selection
- Merging, Joining, and Concatenating DataFrames
- Grouping and Aggregation (GroupBy)
- Pivot Tables and Crosstabs
- Time Series Handling
- Basic Descriptive Statistics and EDA

Module V – Matplotlib for Data Visualization

- Introduction to Data Visualization and Matplotlib
- Basic Plots: Line, Bar, Scatter, Histogram, Pie
- Customizing Plots: Titles, Labels, Legends, Colors, Styles
- Multiple Plots and Subplots
- Working with Figures and Axes Objects
- Annotations and Texts
- Plot Styling and Themes
- Saving Figures in Various Formats
- 3D Plotting (Optional: mpl_toolkits)
- Integrating Matplotlib with NumPy and Pandas

Module VI – Data Handling and Preprocessing

- Understanding Structured and Unstructured Data
- Data Collection Methods
- Data Cleaning and Imputation
- Feature Engineering
- Feature Scaling (Normalization, Standardization)
- Handling Missing and Outlier Values
- Encoding Categorical Variables
- Data Splitting: Train/Test/Validation
- Data Visualization for EDA
- Data Pipelines and Automation

Module VII – Foundations of Machine Learning

- Machine Learning Introduction
- Feature Extraction
- Fine Tuning Hyperparameters
- Semi-Supervised
- Gradient descent
- Overfitting and Underfitting
- Learning Curves and Model Diagnostics
- Regression
- GoogleColab

Module VIII – Supervised Machine Learning

- Introduction to Supervised Learning
- Linear Regression
- Logistic Regression
- Decision Trees
- Support Vector Machines (SVM)
- K-Nearest Neighbors (KNN)
- Naive Bayes Classifier
- Random Forests
- Gradient Boosting Machines (XGBoost, LightGBM)

Module IX – Unsupervised Machine Learning

- Introduction to Unsupervised Learning
- K-Means Clustering
- Hierarchical Clustering
- DBSCAN
- Principal Component Analysis (PCA)
- t-SNE for Dimensionality Reduction
- Anomaly Detection
- Association Rule Mining
- Recommendation Systems
- Evaluation of Clustering

Module X – Reinforcement Learning

- Basics of RL and Agents
- Environment and Reward System
- Markov Decision Process (MDP)
- Q-Learning
- Deep Q-Networks (DQN)
- Policy Gradient Methods
- Actor-Critic Models
- Exploration vs Exploitation
- OpenAI Gym Environments
- Real-life Applications of RL

Module XI – Scikit-Learn

- Introduction to Scikit-learn and Its Core Components
- Data Preparation and Preprocessing
- Building Supervised Learning Models
- Implementing Unsupervised Learning Techniques
- Evaluating Model Performance
- Improving Models with Validation and Tuning
- Automating Workflows with Pipelines
- Saving and Loading Models for Reuse
- Hands-on Mini Projects

Module XII – Foundations of Deep Learning

- Introduction to Deep Learning
- Difference between Machine Learning and Deep Learning
- Biological Inspiration: Neurons and the Human Brain
- Artificial Neural Networks (ANNs)
- Structure of a Neural Network (Input, Hidden, Output Layers)
- Activation Functions (Sigmoid, ReLU, Tanh, Softmax)
- Forward Propagation and Backpropagation
- Loss Functions and Optimization
- Gradient Descent and Learning Rate
- Overfitting, Underfitting, and Regularization Techniques

Module XIII – Neural Network Implementation with Python

- Building Neural Networks from Scratch (NumPy)
- Introduction to Deep Learning Frameworks (TensorFlow & PyTorch)
- Creating and Training Models in TensorFlow
- Creating and Training Models in PyTorch
- Hyperparameter Tuning
- Model Evaluation and Metrics
- Saving and Loading Models
- Batch Normalization and Dropout
- Early Stopping and Callbacks
- Practical Implementation on Simple Datasets

Module XIV – Model Creation with TensorFlow

- Introduction to TensorFlow Ecosystem
- Tensors, Variables, and Operations
- Building Neural Networks with Keras API
- Model Compilation, Training, and Evaluation
- Callbacks (EarlyStopping, ModelCheckpoint)
- TensorBoard for Visualization
- Implementing CNNs, RNNs, and Transfer Learning
- Hyperparameter Tuning (KerasTuner)
- Saving & Loading Models (SavedModel, H5)
- Hands-on Projects using TensorFlow

Module XV – Model Creation with PyTorch

- Introduction to PyTorch Framework
- Tensors, Autograd, and Computational Graph
- Building Neural Networks using nn.Module
- Optimizers and Loss Functions
- Training Loops and Backpropagation
- GPU Acceleration (CUDA)
- Implementing CNNs, RNNs, and Transformers
- Using torchvision and torchtext
- Model Saving, Loading, and Evaluation
- Real-World Projects with PyTorch

Module XVI – Computer Vision

- Introduction to CV and Image Processing
- Image Representation and Color Models
- Image Filtering and Edge Detection
- Object Detection (Haar, HOG)
- CNNs for Image Classification
- Image Augmentation Techniques
- Face Detection and Recognition
- Transfer Learning with CNNs
- Image Segmentation
- OCR (Optical Character Recognition)

Module XVII – Advanced Deep Learning Architectures

- Convolutional Neural Networks (CNNs) – Theory and Architecture
- Applications of CNNs (Image Classification, Object Detection)
- Recurrent Neural Networks (RNNs)
- Long Short-Term Memory (LSTM) and GRU Networks
- Autoencoders and Variational Autoencoders (VAEs)
- Generative Adversarial Networks (GANs)
- Transfer Learning and Pretrained Models
- Attention Mechanism
- Transformers Overview
- Practical Implementation with Real-World Use Cases

Module XVIII – Natural Language Processing (NLP) Fundamentals

- Introduction to NLP
- NLP Pipeline and Text Preprocessing
- Tokenization, Stopwords Removal, Stemming, and Lemmatization
- Bag of Words and TF-IDF
- Word Embeddings (Word2Vec, GloVe, FastText)
- Part-of-Speech (PoS) Tagging and Named Entity Recognition (NER)
- Language Models and Probability-Based NLP
- Text Classification and Sentiment Analysis
- Sequence-to-Sequence Models
- Evaluation Metrics in NLP
- Introduction to BERT
- Introduction to GPT Models
- Hugging Face NLP Pipelines

Module XVIII – Natural Language Processing (NLP) Fundamentals

- Modern Tokenization: BPE, WordPiece, SentencePiece
- NLP Evaluation Metrics

Module XIX – Neural Network-based NLP Models

- Embedding Layers in Deep Learning
- RNNs, LSTMs, and GRUs for Text Data
- Encoder-Decoder Architecture
- Attention Mechanism in NLP
- Transformer Models (BERT, GPT, T5)
- Fine-tuning Pretrained Models for NLP Tasks
- Text Generation and Summarization
- Machine Translation
- Chatbot and Question-Answering Systems

Module XX – Model Optimization and Regularization

- Gradient Descent Variants (SGD, Momentum, RMSProp, Adam)
- Learning Rate Scheduling
- Weight Initialization Techniques
- Dropout and Early Stopping
- Batch Normalization and Layer Normalization
- Data Augmentation as Regularization
- Loss Function Selection (MSE, Cross-Entropy, Huber, etc.)
- Dealing with Overfitting and Underfitting
- Practical Tips for Training Stability
- Tuning and Debugging Deep Models

Module XXI – Core Concepts in Generative AI

- What is Artificial Intelligence?
- What is Generative AI?
- Generative vs Discriminative Models
- Attention Mechanism
- Transformer Architecture
- Encoder, Decoder, and Encoder-Decoder Models
- Autoencoders and Variational Autoencoders (VAEs)
- Latent Space Representations
- Generative Adversarial Networks (GANs)
- Diffusion Models (Stable Diffusion concepts)
- Evolution of Generative Models over Time

Module XXII – Prompt Engineering

- What is Prompt Engineering?
- Components of an Effective Prompt
- Zero-Shot, One-Shot, Few-Shot Prompting
- Chain-of-Thought (CoT) Prompting
- Role Prompting and Contextual Prompting
- ReAct: Reasoning + Acting
- Retrieval-Augmented Prompting
- Self-Consistency Prompting
- Tree of Thought (ToT) Prompting
- Guardrails, Safety Prompts, and Fail-safe Design
- Prompt Injection and Defense Strategies
- Designing Prompts for Code, Reasoning, and Creativity

Module XXIII – Large Language Models (LLMs)

- What is an LLM?
- LLM Architecture
- Attention and Multi-Head Attention
- Token Embeddings, Positional Embeddings
- LLM Parameters and Scaling Laws
- Popular LLM Architectures: GPT, Claude, LLaMA, Gemini
- LoRA, QLoRA, and PEFT Techniques
- RLHF: Reinforcement Learning from Human Feedback
- LLM Distillation
- Evaluation of LLM Outputs
- Using LLM APIs (OpenAI, Hugging Face, Gemini)
- Understanding Context Length and Window Limitations

Module XXIV – Retrieval-Augmented Generation (RAG)

- Concept of RAG in AI Systems
- When and Why to Use RAG
- RAG Architecture and Workflow
- Embeddings for Retrieval
- Vector Databases Overview
- Document Splitting and Chunking
- Query Engines and Retriever Types
- RAG with LangChain
- RAG with LangGraph
- Multimodal RAG (text → image/video)
- Evaluation of RAG Systems
- Optimizing RAG for Latency and Accuracy

Module XXV – Agentic AI & Multi-Agent Systems

- What is Agentic AI?
- Traditional AI Pipelines vs Agentic Systems
- Agent Architecture (planner, memory, executor)
- Types of Memory (episodic, long-term, summarization)
- Action Execution with External Tools
- Multi-Agent Collaboration
- Agent-to-Agent Communication Patterns
- Model Context Protocol (MCP)
- Popular Agent Frameworks
- Designing Safe and Stable Agent Behavior

Module XXVI – CrewAI & Multi-Agent Orchestration

- Introduction to CrewAI
- Crew Roles and Task Distribution
- Tool Usage in CrewAI
- Creating Custom Tools
- Memory in CrewAI (short-term, long-term)
- Embeddings inside Crew Framework
- Knowledge Systems in CrewAI
- Planning, Reasoning, and Delegation
- CrewAI CLI for Automation
- CrewAI Flow for Workflow Design
- Case Study: Fraud Detection Using CrewAI
- Building Multi-Agent Teams for Real Use-Cases

Module XXVII – Advanced GenAI Frameworks (LangChain, LangGraph, LangFlow, LlamaIndex)

- Introduction to Modern GenAI Frameworks
- LangChain: Prompts, Chains, Agents & Memory
- Retrieval-Augmented Generation (RAG) Pipelines
- Vector DB Integration: FAISS, Pinecone, Chroma
- LangGraph: Nodes, State & Multi-Agent Orchestration
- Workflow Guardrails, Error Handling & State Machines
- LangFlow: No-Code Visual RAG & Agent Design
- LlamaIndex: Index Types, Contexts & Query Engines
- Advanced RAG Techniques (Fusion, Re-Ranking, Hybrid Search)
- Hands-On: Build a Production-Ready Multi-Agent RAG System

Module XXVIII – OpenAI Function Calling and Tool Use

- Introduction to OpenAI Function Calling (GPT-4 & Tools API)
- Designing structured functions for tool usage
- JSON schema, argument parsing, function routing
- Comparison: LangChain Tools vs. OpenAI Tools vs. ReAct
- Calling APIs like weather, calculator etc. with LLM
- Multi-function invocation
- Tool selection strategies and error handling
- Introduction to OpenAI Assistants API
- Using tools within context and chaining multiple calls
- Common tool-using LLM applications: Retrieval, Execution, Summarization

Module XXIX – Planning and Reasoning Techniques in Agents

- Overview of reasoning strategies in LLMs
- ReAct (Reasoning + Acting) pattern
- Plan-and-Execute architecture (LangChain, Meta's implementation)
- Task decomposition with LLMs
- Using LLMs for planning tasks (To-do lists, workflows, subtasks)
- Handling intermediate outputs, tool feedback
- Agent decision flow: IF-THEN conditions, tool selection, fallback logic
- Integrating external APIs and knowledge bases
- Multi-agent orchestration and communication
- Limitations of LLM-based reasoning (hallucinations, context loss)

Module XXX – Responsible, Ethical & Safe AI

- Bias in AI and Generative Models
- Dataset Bias Mitigation and Fairness Metrics
- Explainable AI (XAI) and Model Transparency
- Data Privacy, Protection, and Secure AI Practices
- Global AI Governance and Compliance (EU AI Act, India, Worldwide)
- Adversarial Attacks, Deepfakes, and Synthetic Media Risks
- Prompt Injection & Jailbreak Attacks in LLMs
- Safety Protocols, Red-Teaming, and Risk Assessment
- Responsible AI Frameworks and Ethical Deployment
- Ongoing Monitoring, Auditing, and Human Oversight

Module XXXI – Model Testing and Performance Analysis

- Evaluation Metrics for Classification & Regression
- Confusion Matrix, ROC-AUC, Precision-Recall & F1-Score Analysis
- Cross-Validation Techniques and Model Generalization
- Bias-Variance Tradeoff and Error Diagnostics
- Hyperparameter Tuning (Grid, Random, Bayesian Search)
- Handling Imbalanced Data (SMOTE, Class Weights, Resampling)
- Model Comparison, Benchmarking & Ensemble-Based Selection
- Experiment Tracking (MLflow, Weights & Biases)
- Model Interpretability and Trustworthiness
- Reproducibility, Documentation & Model Testing

Module XXXII – Model & Agent Deployment

- Model Serialization and Packaging (Pickle, Joblib, SavedModel)
- Building REST APIs with Flask and FastAPI
- Deploying Interactive Apps using Streamlit and Gradio
- Containerization and Deployment with Docker
- Cloud Deployment Strategies (AWS, GCP, Azure)
- Agent Lifecycle and Agentic RAG Deployment (LlamaIndex, LangChain)
- Workflow Automation using n8n for ML/GenAI Pipelines
- CI/CD for Model and Agent Deployments
- Monitoring, Logging, and Observability (LangSmith, PromptLayer)
- Model/Agent Versioning, Rollbacks, Safety, and Human-in-the-Loop Oversight

Module XXXIII – Project Development

- Defining Project Objectives and Success Criteria
- Problem Formulation for ML, DL, GenAI, and Agentic Systems
- Data Collection, Cleaning, Labeling, and Knowledge Base Preparation
- Model Selection (ML, Deep Learning, LLMs, RAG, Agents)
- Training, Fine-Tuning, Prompt Engineering & Optimization
- Evaluation and Metrics for Predictive, Generative & Agentic Tasks
- Deployment Strategy (APIs, Apps, RAG Pipelines, Agentic Workflows)
- Feedback Loop, Monitoring, and Continuous Improvement

Module XXXIII – Project Development

- Documentation, Reporting, and Experiment Tracking
- End-to-End Case Study Development (ML → DL → GenAI → Agents)

Thank You!

Thank you for downloading our curriculum!

We're excited to be part of your learning journey. Feel free to reach out if you need more information.

Contact Us



enquiries@proleed.academy



+91-9988475050



www.proleed.academy