🤖 Advanced AI, Generative AI & Agentic AI with Python

Comprehensive Training Program Curriculum

☰ Course Modules Overview

Module 1: Python Programming

Module 2: EDA & Machine Learning

Module 3: Artificial Intelligence

Module 4: Natural Language Processing

Module 5: Deep Learning | Neural Networks

Module 6: Generative AI

Module 7: LLM (Large Language Models)

Module 8: Prompt Engineering

Module 9: Agentic AI

Module 10: MCP (Model Context Protocol)

Module 11: Vector Database

Module 12: RAG

Module 13: Cloud Integration with Generative AI

Module 14: LLMOps

1 Python Programming

Basic Python Content

Introduction to Python: Installation, setup, Jupyter Notebook

Data Types & Variables: int, float, string, boolean, complex

Operators: Arithmetic, Relational, Logical operators

Control Flow: Conditional statements, loops, nested structures

Data Structures: Strings, Lists, Tuples, Sets, Dictionaries

Functions: Definition, parameters, scope, decorators

Object-Oriented Programming (OOPs)

OOPs Fundamentals: Classes, Objects, Methods

Inheritance: Single, Multiple, Multilevel, Hierarchical

Polymorphism: Method Overloading, Duck Typing

Encapsulation: Private Variables, Getters, Setters

Abstraction: Abstract Classes and Methods

Dunder Methods: Magic Methods, Operator Overloading

🔧 **Mini Projects:** Banking System, Library Management, Student Database, Employee Payroll System

Advanced Python Content

Advanced Functions: Recursion, Closures, Decorators

Iterators & Generators: Custom iterators, yield, memory efficiency

Exception Handling: Try-Except, Custom exceptions

File Operations: File handling, CSV, JSON, OS operations

Regular Expressions: Pattern matching, text cleaning

Multithreading: Parallel processing, thread safety

2 EDA & Machine Learning

Supervised Learning Algorithms

Linear Regression: Simple & Multiple, R-squared evaluation

Logistic Regression: Binary classification, Sigmoid function

k-NN Classifier: Distance metrics, k-value selection

Naive Bayes: Bayes' theorem, feature independence

Decision Trees: Splitting criteria, pruning, overfitting

SVM: Hyperplane, kernel trick, C and gamma parameters

Random Forest: Bagging, feature randomness, OOB score

Gradient Boosting: Boosting concept, regularization

Unsupervised Learning

K-Means Clustering: Elbow method, silhouette score

Hierarchical Clustering: Agglomerative vs divisive, dendrograms

DBSCAN: Density-based clustering, outlier detection

Dimensionality Reduction: PCA, t-SNE, UMAP

Model Evaluation

Performance Metrics: Accuracy, Precision, Recall, F1-Score

Validation Techniques: Train-test split, Cross-validation

> **Advanced Metrics:** ROC Curve, AUC, Log Loss

4 Natural Language Processing (NLP)

Text Preprocessing & Representation

> **Text Preprocessing:** Tokenization, stopwords, stemming, lemmatization

> **Advanced Preprocessing:** POS tagging, NER, regex cleaning

> **Text Representation:** Bag of Words, TF-IDF

> **Word Embeddings:** Word2Vec, GloVe, FastText

> **Sentence Embeddings:** Doc2Vec, Sentence Transformers

Feature Engineering & Language Modeling

> **Feature Engineering:** N-grams, co-occurrence matrices

Vectorization: CountVectorizer, TfidfVectorizer

Language Models: N-gram models, neural language models

Evaluation: BLEU, ROUGE, METEOR, Perplexity

Optimization & Advanced Techniques

Model Optimization: Quantization, pruning, distillation

Quality Control: Hallucination detection

Performance: Memory & latency optimization

5 Deep Learning (ANN, CNN, RNN)

Artificial Neural Networks (ANN)

Fundamentals: Perceptron, multilayer perceptron, activation functions

Training: Backpropagation, gradient descent, loss functions

Optimization: SGD, Adam, RMSProp, batch normalization

Regularization: Dropout, weight regularization

🔧 Building ANN using TensorFlow/Keras for classification and regression

Convolutional Neural Networks (CNN)

CNN Fundamentals: Convolution, pooling, parameter sharing

CNN Layers: Convolution, activation, pooling, fully connected

Advanced Concepts: Transfer learning, data augmentation

Architectures: LeNet-5, AlexNet, VGGNet, ResNet, InceptionNet

🔧 Image classification using CNN (MNIST/CIFAR-10), feature visualization

Recurrent Neural Networks (RNN)

RNN Basics: Sequential data, temporal dependencies, hidden states

RNN Variants: LSTM, GRU, Bidirectional RNNs

Applications: Text generation, sentiment analysis, time series

Challenges: Vanishing gradients, BPTT

🔧 LSTM for text classification, GRU for stock price prediction

6 Generative AI

Foundational Models

Text-Based Models: GPT-4, LLaMA, Mistral training with Transformers

Multimodal Models: CLIP, DALL·E, Gemini, GPT-4 Turbo

CLIP Architecture: Text-image pairing, contrastive learning

🔧 Running OpenAI's CLIP model for text-to-image retrieval

Advanced Generative Techniques

VQGAN & Transformers: Vector quantized GANs, AI art generation

Autoencoders & VAEs: Variational autoencoders, high-resolution images

Stable Diffusion: Text-to-image generation, fine-tuning

🔧 Generating AI Art using VQGAN + CLIP, implementing VAE in PyTorch

AI Tools & Platforms

Hugging Face Ecosystem: Transformers, model deployment, Spaces

CrewAI: Multi-agent AI systems, role-based task assignment

Groq: High-speed AI inference, model optimization

GitHub Copilot: AI-powered coding, best practices

LlamaIndex: Document processing, enterprise AI applications

FastAPI: AI model deployment, REST APIs

7 Large Language Models (LLM)

Major LLM Ecosystems

OpenAI: GPT models, LangChain framework, Whisper speech-to-text

Google: Gemini AI, Google Vision, multimodal capabilities

Meta: LLaMA 3, foundation models, cloud deployment

Other Leaders: DeepSeek, Mistral, Mixtral, Grok, Claude

> 🔧 Running LLaMA 3 locally, implementing chatbots with various APIs

Fine-Tuning & Optimization

Fine-Tuning Techniques: Full fine-tuning vs prompt engineering

LoRA: Low-rank adaptation, computation reduction

QLoRA: Quantized LoRA, memory optimization

Deployment: FastAPI, model security, governance

🔧 **Fine-tuning LLaMA 3 with LoRA & QLoRA, deploying as FastAPI service**

8 Prompt Engineering

Prompt Fundamentals

Prompt Basics: Instructions, context, input data, output constraints

Model Settings: Temperature, top-k, top-p, stop tokens

Shot-Based Prompting: Zero-shot, one-shot, few-shot techniques

Advanced Prompting Techniques

Chain of Thought (CoT): Step-by-step reasoning in LLMs

Self-Consistency: Multiple answers generation and selection

Role-Playing: AI personas for specialized tasks

ReAct Framework: Reasoning + acting for

autonomous agents

Dynamic Structured Prompting: Context-based adaptive prompts

🔧 Implementing CoT prompting, creating AI assistants with different personas

Integration with External Systems

RAG Integration: Retrieval-augmented generation with vector databases

Out-of-Date Learning: Handling outdated information

Tool Integration: External APIs and data sources

9 Agentic AI - Single-Agent & Multi-Agent Systems

Single-Agent Systems

Agent Architecture: Decision-making, memory, long-term planning

LLM Integration: LangChain, OpenAI, Hugging Face integration

Tool Capabilities: User inputs, actions, tool-use capabilities

🔧 Building goal-driven single AI agent with LangChain, memory-based reasoning with FAISS

Multi-Agent Systems (MAS)

Agent Communication: Coordination, role assignment

Swarm Intelligence: Distributed agents, collective behavior

CrewAI Framework: Team-based AI agents, workflow pipelines

🔧 Creating multi-agent research assistant, AI-powered content generation team

Specialized Agent Frameworks

SmolAgent: Lightweight AI agents, cost optimization

Phi Data: AI agent memory, context optimization

Deployment: Cloud deployment, web application integration

N8N Integration: Workflow automation, node-based systems

10 MCP (Model Context Protocol)

MCP Architecture & Components

Core Components: MCP Host, MCP Client, MCP Server

Communication: Data flow, resource management

Integration: LLMs and AI agents integration

Implementation: MCP SDKs and frameworks

Practical Applications

Resource Management: Tools, prompts, and resources

Use Cases: AI and data science applications

Setup & Configuration: Implementation best practices

11 Vector Database

Vector Database Fundamentals

Database Concepts: Traditional vs vector databases

Vector Embeddings: Text, image, audio to vectors

Similarity Metrics: Cosine similarity, Euclidean distance

Embedding Models: OpenAI, Hugging Face, BERT, SentenceTransformers

🔧 Generating embeddings with OpenAI text-embedding-ada-002, BERT models

Vector Database Platforms

FAISS: Facebook AI similarity search, large-scale indexing

Pinecone: Managed vector database, real-time AI

search

ChromaDB: LangChain integration, RAG pipelines

Comparison: FAISS vs Pinecone vs Milvus

🔧 Building question-answering chatbot with OpenAI + Pinecone, ChromaDB RAG implementation

AI Search Applications

Document Retrieval: FAISS-based search engines

Real-time Search: AI-powered search applications

Integration: LLMs with vector databases for enhanced AI responses

---

Advanced AI, Generative AI & Agentic AI with Python

Comprehensive training program covering 14 modules from Python fundamentals to cutting-edge AI technologies

Designed for professionals seeking expertise in modern AI development and deployment