www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

GEN AI LEARNING ROADMAP

Ready to Take the Leap?!

LIKE | SHARE | REPOST

See full Roadmap ⟶

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- *1. Python Basics*
  - *Variables, Data Types (int, float, string, bool)*
  - *Control Structures (if-else, loops)*
  - *Functions and Lambda expressions*
  - *Classes and Object-Oriented Programming (OOP)*

- *2. Data Structures*
  - *Lists, Tuples, Dictionaries, Sets*
  - *Comprehensions (list, dictionary comprehensions)*

- *3. File Handling & Data I/O*
  - *Reading/writing files (text, JSON, CSV)*
  - *Working with APIs (requests module)*
  - *Environment variables handling (os, dotenv)*

- *4. Python Libraries for AI Development*
  - *NumPy (numerical computations)*
  - *Pandas (data manipulation)*
  - *Matplotlib/Seaborn (data visualization)*
  - *Requests (API calls)*
  - *dotenv (handling environment variables securely)*

→

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- ***5. Working with APIs & Asynchronous Programming***
  - ***Calling APIs for language models (OpenAI, Anthropic)***
  - ***Async programming basics with asyncio (useful for responsive multi-agent systems)***

- ***6. Object-Oriented Design for AI Agents***
  - ***Classes to encapsulate AI agents, tools, and pipeline components***
  - ***Design patterns to structure multi-agent collaborations***

- ***7. Basic Web Frameworks (for deployment & interfaces)***
  - ***Flask or FastAPI basics for serving AI models or agents as web apps***

- ***8. Handling JSON and Nested Data***
  - ***Parsing and manipulating complex JSON data returned from AI APIs***

LIKE | SHARE | REPOST

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- *1. Fundamental Concepts*
  - *Difference between AI, ML, DL, GenAI, Agentic AI*
  - *Supervised vs. Unsupervised vs. Reinforcement Learning*
  - *Underfitting, Overfitting, Bias-Variance Tradeoff*
  - *Model Validation: Train/Test Split, Cross-Validation*

- *2. Key Algorithms & Techniques*
  - *Regression (linear, logistic)*
  - *Classification (decision trees, SVM, k-NN, naive Bayes)*
  - *Clustering (k-means, hierarchical)*
  - *Dimensionality Reduction (PCA, t-SNE)*

- *3. Model Evaluation & Metrics*
  - *Accuracy, Precision, Recall, F1-Score*
  - *Confusion Matrix, ROC-AUC*
  - *Loss functions (MSE, Cross-Entropy)*

- *4. Feature Engineering*
  - *Feature Selection & Extraction*
  - *Data Preprocessing (scaling, encoding, imputation)*
  - *Handling outliers and missing values*

→

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- ***5. Ensemble Methods***
  - ***Bagging, Boosting (Random Forest, AdaBoost, Gradient Boosting)***
  - ***Stacking & Voting***

- ***6. Deep Learning Basics (for Gen AI)***
  - ***Neural Networks: Perceptron, Activation Functions, Backpropagation***
  - ***CNNs for image data, RNNs for sequence data***
  - ***Intro to Transformers & Attention Mechanisms***
  - ***Basics of Generative Models (GANs, VAEs)***

- ***Practical Skills***
  - ***Using ML libraries: scikit-learn, XGBoost, LightGBM***
  - ***Model building, saving/loading, and inference***
  - ***Hyperparameter tuning and grid/random search***

- ***Advanced and Agentic AI-Relevant Topics***
  - ***Sequence Modeling (for dialogue/LLMs: autoregressive models, sequence labeling)***
  - ***Reinforcement Learning basics (for goal-directed agents)***
  - ***Understanding Vector Embeddings and Similarity Search***
  - ***Transfer Learning: Using pre-trained models***

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- *1. Fundamentals of Neural Networks*
  - *Perceptron model and multilayer perceptrons (MLP)*
  - *Activation functions (ReLU, sigmoid, tanh)*
  - *Forward propagation & backpropagation*
  - *Loss functions and optimization algorithms (gradient descent, Adam)*

- *2. Convolutional Neural Networks (CNNs)*
  - *Convolution operations, filters, and feature maps*
  - *Pooling layers (max, average)*
  - *Applications in image recognition and feature extraction (important for multimodal generative AI)*

- *3. Recurrent Neural Networks (RNNs) and Variants*
  - *Basics of RNNs for sequential data*
  - *Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU)*
  - *Applications in NLP, time series, and sequence generation*

- *4. Transformers and Attention Mechanisms*
  - *The concept of attention and self-attention*
  - *Transformer architecture (encoder, decoder)*
  - *Popular models based on transformers (BERT, GPT, T5)*
  - *Foundation for large language models (LLMs) used in generative and agentic AI*

→

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- *5. Generative Models*
  - *Generative Adversarial Networks (GANs) and their components (generator, discriminator)*
  - *Variational Autoencoders (VAEs)*
  - *Diffusion models (emerging in image and audio generation)*

- *6. Transfer Learning and Fine-tuning*
  - *Using pre-trained models and adapting them to specific generative or agentic AI tasks*
  - *Benefits in efficiency and performance*

- *7. Reinforcement Learning (RL) Basics*
  - *Markov Decision Processes (MDP)*
  - *Policy gradients and Q-learning*
  - *Importance for goal-directed behavior in agentic AI*

- *8. Deep Learning Frameworks and Tools*
  - *TensorFlow, PyTorch, Keras for building and training models*
  - *Practical skills in model deployment and serving*

GreekWave AI

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- *1. Basics of NLP*
  - *Text processing: tokenization, stemming, lemmatization*
  - *Part-of-Speech (POS) tagging*
  - *Named Entity Recognition (NER)*
  - *Dependency parsing and syntactic analysis*

- *2. Text Representation*
  - *Bag-of-Words, TF-IDF*
  - *Word embeddings: Word2Vec, GloVe, FastText*
  - *Transformer-based contextual embeddings (BERT, GPT embeddings)*

- *3. Language Modeling*
  - *Understanding language models and their training*
  - *N-grams, statistical language models*
  - *Pretrained large language models (LLMs) and fine-tuning them*

- *4. Natural Language Understanding (NLU) Tasks*
  - *Intent recognition and classification*
  - *Sentiment analysis and emotion detection*
  - *Text classification and topic modeling*
  - *Semantic similarity and paraphrase detection*

→

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- *5. Natural Language Generation (NLG) Tasks*
  - *Text generation and summarization (extractive and abstractive)*
  - *Machine translation*
  - *Question answering systems*
  - *Dialogue systems and chatbots*
- *6. Advanced NLP Techniques for Generative AI*
  - *Transformers and attention mechanisms for sequence modeling*
  - *Encoder-decoder architectures for complex generation tasks*
  - *Contextual understanding for coherent and meaningful output*
  - *Handling ambiguity, coreference resolution, and context retention*
- *7. Tools and Libraries*
  - *NLTK, SpaCy, Gensim for basic NLP*
  - *HuggingFace Transformers for state-of-the-art models*
  - *Sentiment analysis libraries and APIs*

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

Generative AI (Gen AI) Fundamentals & Advanced

- ***Fundamentals***
  - ***1. Large Language Models (LLMs)***
    - ***What are LLMs and how they work (Transformers, attention mechanism)***
    - ***Pretraining vs. fine-tuning***
    - ***Popular models: GPT, BERT, T5, LLaMA***
  - ***2. Embeddings***
    - ***Word vs. sentence vs. document embeddings***
    - ***Embedding models: OpenAI, HuggingFace, Cohere***
    - ***Use cases: semantic search, clustering, similarity matching***
  - ***3. Vector Databases***
    - ***Purpose: storing and retrieving embeddings efficiently***
    - ***Tools: FAISS, ChromaDB, Weaviate, Pinecone***
    - ***Concepts: indexing, similarity metrics (cosine, Euclidean)***

→

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- 4. Retrieval-Augmented Generation (RAG)
  - Architecture: retriever + generator
  - Benefits: grounding LLMs with external knowledge
  - Implementation: LangChain + FAISS/ChromaDB + OpenAI
- 5. LangChain Framework
  - Components: Chains, Agents, Tools, Memory
  - Use cases: chatbots, document Q&A, autonomous workflows
  - Integrations: OpenAI, HuggingFace, Google Search, SQL, APIs

Advanced

- 1. Model Optimization
  - Fine-tuning techniques: LoRA, QLoRA, PEFT
  - Quantization & pruning for deployment
  - Multi-modal models: CLIP, Flamingo, Gemini
- 2. Advanced RAG Techniques
  - Chunking strategies: semantic vs. fixed-size
  - Hybrid search: combining keyword + vector search
  - Cross-encoder vs. bi-encoder retrieval
  - Memory-augmented RAG for long-term context

→

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- ***3. Prompt Engineering***
  - ***Prompt templates, chaining, and dynamic construction***
  - ***Few-shot, zero-shot, and chain-of-thought prompting***
  - ***Guardrails and prompt injection mitigation***

- ***4. Frameworks & Ecosystems***
  - ***LlamaIndex: document indexing, query engines***
  - ***LangChain Advanced: custom agents, callbacks, tracing***
  - ***OpenAgents, AutoGen, CrewAI for agentic workflows***

- ***5. Security & Ethics***
  - ***Hallucination reduction strategies***
  - ***Bias detection and mitigation***
  - ***Safety protocols and responsible AI practices***

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

Agentic AI

- *1. Introduction to Agentic AI*
  - *What makes an AI agentic (autonomy, goal-driven, reasoning loop)*
  - *Key differences: Gen AI vs. AI Agents vs. Agentic AI*
  - *Examples: AutoGPT, BabyAGI, CrewAI*

- *2. Core Components of Agents*
  - *Planning → breaking down big goals into subtasks*
  - *Tool Use → interacting with APIs, databases, calculators*
  - *Memory → short-term, long-term, semantic, episodic*
  - *Reflection → evaluating own outputs & adjusting*

- *3. Agent Workflows*
  - *Reasoning + Acting loop (ReAct pattern)*
  - *Goal → Plan → Execute → Reflect → Iterate*
  - *How agents differ from scripted automation*

- *4. MCP (Multi-Component Protocol)*
  - *Purpose: modular coordination between agents and tools*
  - *Agent ↔ Tool communication standards*
  - *Adoption in AutoGen, CrewAI, OpenAgents*

→

GreekWave AI

www.greekwavetech.in

Python ML DL NLP Gen AI Agentic AI

- ***5. Frameworks & Tools***
  - ***LangChain Agents → task automation with tool use***
  - ***LlamaIndex → knowledge-augmented agents***
  - ***AutoGen, CrewAI, OpenAgents → multi-agent workflows***

- ***6. Multi-Agent Systems***
  - ***Collaboration strategies: role assignment, leader-follower, peer-to-peer.***
  - ***Shared memory and communication protocols.***
  - ***Emergent behavior: group decision-making & swarm intelligence.***
  - ***Multi Agents Orchestration.***
  - ***Use cases: research groups, startup simulations, distributed teams.***

SCAN ME FOR

GREEKWAVE TECHNOLOGIES

GEN AI & AGENTIC AI USE CASES

LIKE | SHARE | REPOST