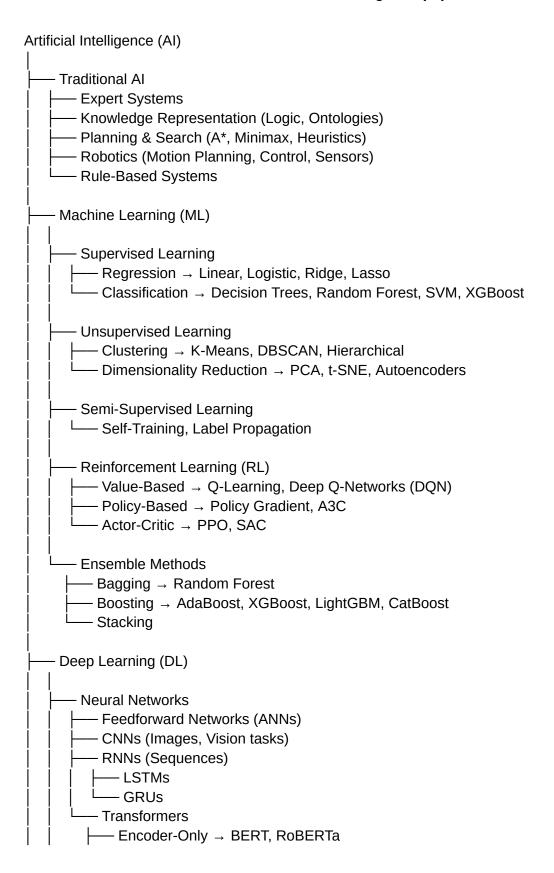
Artificial Intelligence (AI)



— Generative AI — Large Language Models (LLMs → GPT, LLaMA, Mistral) — Diffusion Models (Stable Diffusion, DALL·E, MidJourney) — GANs (Generative Adversarial Networks → DeepFakes, Data Synthesis) — Code Generation (Codex, Code Llama, StarCoder)
 Federated Learning (Privacy-first ML) Synthetic Data Generation (GANs, Diffusion for training data) Explainable AI (XAI → SHAP, LIME, InterpretML) Graph Neural Networks (GNNs → Fraud, Recommenders, Drug Discovery) Edge AI (Deploying models on mobile/IoT) Quantum Machine Learning (QML → Qiskit, PennyLane)

	Trustworthy & Responsible AI (Bias, Fairness, Safety)
 	— Application Domains
	Natural Language Processing (NLP) → Translation, Chatbots, Summarization
	Computer Vision (CV) → Object Detection, Face Recognition, Medical Imaging
	— Speech & Audio → ASR, TTS, Voice Assistants
	Robotics → Autonomous Driving, Industrial Robots, Drones
	Healthcare AI → Diagnosis, Drug Discovery, Predictive Analytics
	Finance Al → Fraud Detection, Algo-Trading, Risk Assessment
	Recommendation Systems → E-commerce, Media, Ads
	Al for Sustainability → Energy Optimization, Climate Modeling

Artificial Intelligence (AI) Knowledge Tree

This part covers the **conceptual foundation**:

- Traditional AI → Rule-based systems, expert systems, logic (useful in early AI).
- **Machine Learning (ML)** → Supervised, unsupervised, reinforcement, semi-supervised, ensembles.
- **Deep Learning (DL)** → CNNs, RNNs, Transformers, LLMs, generative models.
- **Trending Topics** → RAG, agentic AI, multimodal, federated, explainability, GNNs, edge AI, quantum ML.
- Applications → NLP, CV, speech, robotics, healthcare, finance, recommendation, sustainability.

AI/ML Tools

```
AI/ML Tools

    Core Python ML/DL Frameworks

     — TensorFlow / Keras → Deep Learning
     — PyTorch / Fastai → DL research & applications
   — Scikit-learn → Classical ML
     — Statsmodels → Statistical models
   — XGBoost / LightGBM / CatBoost → Gradient boosting
   — Hugging Face Transformers → NLP, LLMs
   — JAX → High-performance ML research
   RAPIDS (cuML, cuDF) → GPU ML pipelines

    Data Handling & Preprocessing

   — Pandas / NumPy → Data analysis & math
     — Dask → Parallel data processing

    PySpark (MLlib) → Distributed ML

     — Vaex → Big dataframes
   Featuretools → Automated feature engineering

    Visualization & Explainability
```

```
— Matplotlib / Seaborn → Static plots
Plotly / Bokeh → Interactive dashboards
— SHAP → Global & local explanations
— LIME → Local interpretability
Yellowbrick → ML model visualization

    Experimentation & Tracking

— MLflow → Tracking + model registry
— Weights & Biases (W&B) → Live monitoring
— Neptune.ai / Comet ML → Experiment tracking
ClearML → Orchestration + monitoring

    Deployment & Serving

Model Serving

   — TensorFlow Serving
   — TorchServe
   — ONNX Runtime
   L— BentoML
  Web Frameworks (Python)
   Flask → Lightweight APIs for ML
```

```
FastAPI → High-performance async APIs
   ☐ Django → Enterprise-scale apps
   - Containers & Infra
     — Docker
     Kubernetes

Model Demos

     – Streamlit
     Gradio
- MLOps & Pipelines
— Kubeflow → End-to-end ML pipelines
— Apache Airflow → Workflow orchestration
Prefect → Workflow orchestration
— DVC (Data Version Control) → Dataset & model versioning
 — Feast → Feature Store
— Metaflow → Production pipelines

    Cloud AI Platforms

— AWS → SageMaker, Rekognition, Comprehend, Bedrock

    Google Cloud → Vertex AI, AutoML, BigQuery ML, GenAI Studio

    Microsoft Azure → Azure ML, Cognitive Services, OpenAl Service
```

AI/ML Tools Tree

This part is about the "how to build" side:

- Frameworks → TensorFlow, PyTorch, Scikit-learn, Hugging Face, etc.
- Data Handling → Pandas, Dask, PySpark, Featuretools.
- Visualization & Explainability → Matplotlib, SHAP, LIME.
- Experiment Tracking → MLflow, W&B, ClearML.
- Deployment & Serving → Flask, FastAPI, Django, Streamlit, Gradio, Docker, Kubernetes.
- MLOps & Pipelines → Kubeflow, Airflow, DVC, Feast.

- Cloud Platforms → AWS SageMaker, GCP Vertex AI, Azure ML, IBM Watson, Databricks, Snowflake.
- **Specialized Tools** → OpenVINO, TensorRT, LangChain, Haystack.

Steps to cover all topic

Step 1 – Foundations

- Python, NumPy, Pandas, Matplotlib/Seaborn.
- Statistics, probability, linear algebra.

Step 2 - ML Basics

- Scikit-learn (regression, classification, clustering).
- XGBoost, LightGBM.

Step 3 – Deep Learning

- PyTorch or TensorFlow.
- CNNs (CV), RNN/LSTM (time-series), Transformers (NLP).

Step 4 – Advanced Al

- Generative AI (GANs, VAEs, Diffusion).
- LLMs + Hugging Face + LangChain.
- RAG + vector databases.

Step 5 – Deployment & MLOps

Flask, FastAPI, Docker, Kubernetes.

- MLflow, W&B for tracking.
- Cloud platforms (AWS/GCP/Azure).

Step 6 – Specialization

- Pick a domain: NLP, CV, speech, robotics, healthcare, finance, recommender systems.
- Explore trending: GNNs, federated learning, quantum ML, edge Al.