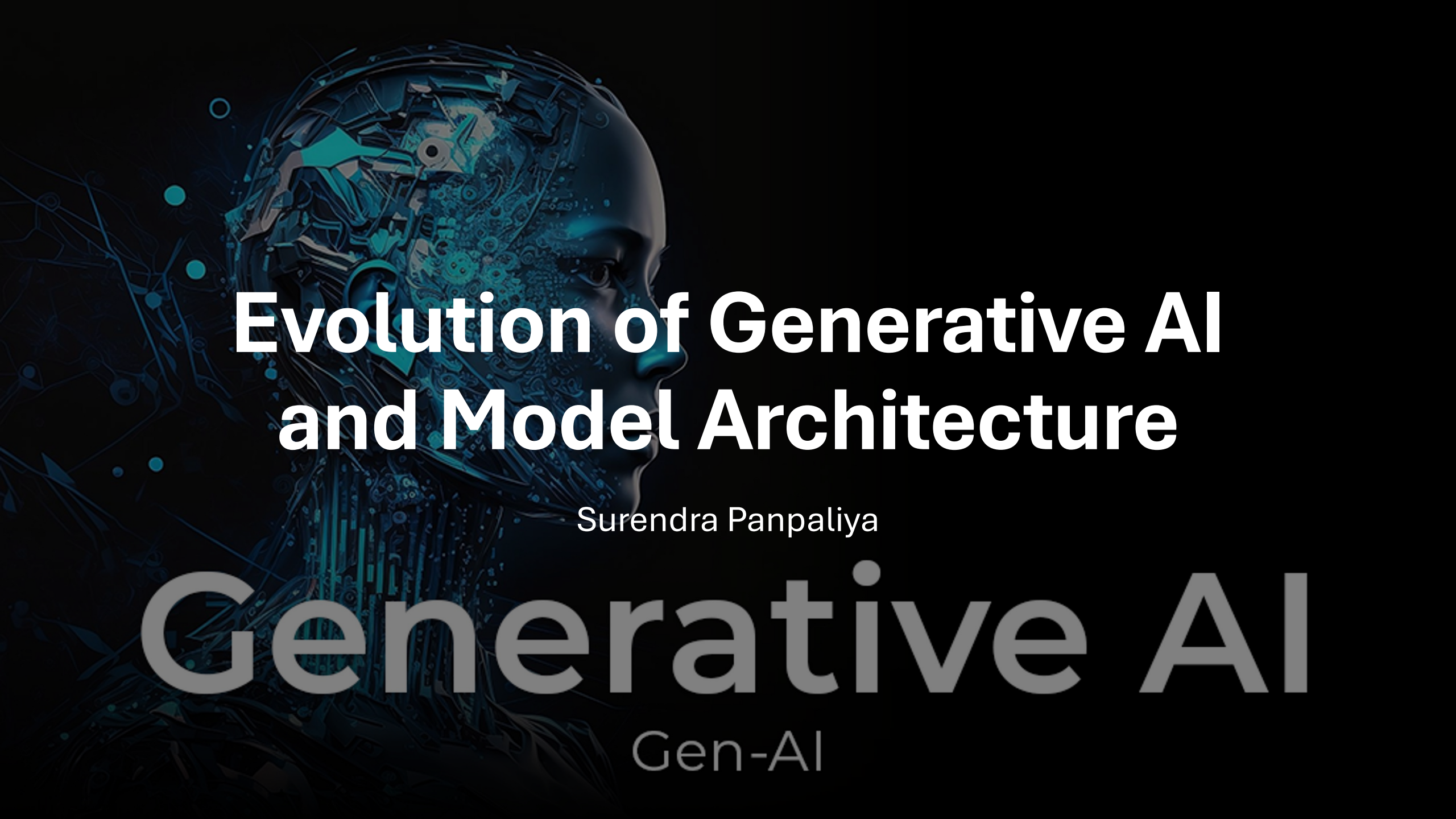


Evolution of Generative AI and Model Architecture

Surendra Panpaliya

PREREQUISITES

- Prior exposure to **Jupyter Notebooks** or **VS Code**
- Python for Data Science, Machine Learning
- Some experience with **LLMs** or **prompt engineering**

Lab Setup Requirements

Python **3.11 / 3.12**

Jupyter Notebook / VS Code

OpenAI Python SDK (openai)

OpenAI account with API key

Lab Setup Requirements

- pymilvus for Milvus Vector DB
- Streamlit
- GitHub for code submissions
- Milvus setup: Zilliz Cloud

Target Audience

Developers, Architect

Data Scientists,

AI Enthusiasts, and

Tech Professionals interested in building

Program Objectives

Understand **Generative AI evolution,**

Models, and architectures.

Learn how **LLMs (GPT-4, GPT-5)** work

Transformers, tokenization, encoding & decoding.

Program Objectives

Explore **Prompt Engineering**

(ACTORS Framework) and safe AI use.

Understand **RAG (Retrieval-Augmented Generation)**

AI Agent architectures.

Program Objectives

Study **LangChain, LangGraph, LangSmith,**

AutoGen ecosystems through demos.

Learn about **vector databases**

PGVector, Chroma, Milvus and their secure use.

Program Objectives

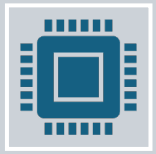
Apply concepts to **healthcare datasets**

Explore **AI governance, data privacy,**

Security in enterprise environments.

Agenda

DAY 1: Evolution of Generative AI and Model Architecture

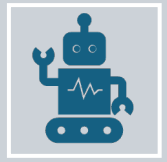


DAY 2: Prompt Engineering & Microsoft Copilot Integration

DAY 3: RAG Architecture and LangChain Fundamentals

Agenda

DAY 4: LangGraph, LangSmith & Multi-Agent Architecture



DAY 5: AutoGen Multi-Agent System and AI Governance

Evolution of Generative AI and Model Architecture

Surendra Panpaliya

GKTCS Innovations

https://www.gktcs.com

Agenda

- Generative AI in Enterprise
- Evolution of GPT models
- Tokenization, embeddings,
- Context window in GPT-5

Agenda

- OpenAI Python SDK setup &
- API integration
- Basics of Prompting
- (system, user, assistant roles)

Hands-On

- Setup GPT-5 environment
- Generate text: Q&A,
- Summarization, translation
- Create a chatbot skeleton

Agenda

1. What is Generative AI?

2. Neural Generative Modeling

3. Large Language Models (LLMs)

4. Transformer Architecture & Attention Mechanism

What is Generative AI?

BRANCH OF ARTIFICIAL INTELLIGENCE

FOCUSES ON **CREATING NEW CONTENT**

TEXT, IMAGES, CODE, AUDIO, VIDEO

BY LEARNING PATTERNS FROM LARGE DATASETS.

What is Generative AI?

Large Language Models (LLMs)

Understand natural language,

Context, and intent, and

Produce human-like responses.

What is Generative AI?

https://www.youtube.com/watch?v=rwF-X5STYks

Neebal Technologies
www.neebal.com

Generative AI Applications

- Art and Design
- Music Composition
- Content Creation
- Virtual Worlds
- Fashion and Product Design
- Financial Forecasting
- Customer Behavior Analysis
- Demand Forecasting
- Healthcare Diagnosis and Prognosis
- Fraud Detection

Why is it transformative for enterprises?

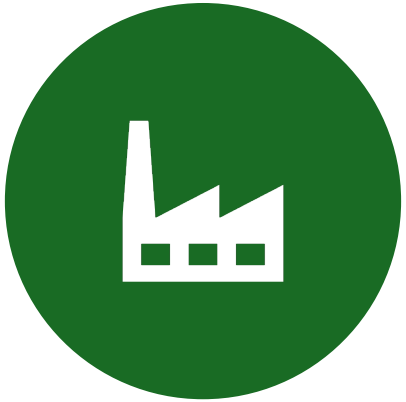
IMPROVES DECISION-MAKING WITH FASTER INSIGHTS

REDUCES TIME-TO-MARKET FOR DIGITAL SOLUTIONS

ENABLES PERSONALIZATION AT SCALE

Use Cases for Enterprises

1. BANKING & FINANCIAL SERVICES (BFSI)



2. MANUFACTURING

3. HEALTHCARE & LIFE SCIENCES

1. Banking & Financial Services (BFSI)

- Customer Support Copilots
- Fraud Detection Reports
- Risk & Compliance Automation
- Personalized Banking Advisors

Customer Support Copilots

AI-powered assistants

to handle loan queries,

credit card disputes, and

KYC FAQs.

Fraud Detection Reports

Automated analysis of

suspicious transaction patterns

with explanations in natural language.

Risk & Compliance Automation

Generating

compliance summaries,

audit reports, and

regulatory updates.

Personalized Banking Advisors

Recommending

financial products

based on customer profiles.

Example

AI generates a **weekly**

risk compliance dashboard

with insights for regulators,

saving analysts 10–15 hours per week.

2. Manufacturing

Predictive Maintenance Insights

Supply Chain Optimization

Digital Twins & Simulation

Workforce Training Copilots

Example

A **factory operations assistant**

that summarizes machine downtime logs and

suggests preventive actions.

3. Healthcare & Life Sciences

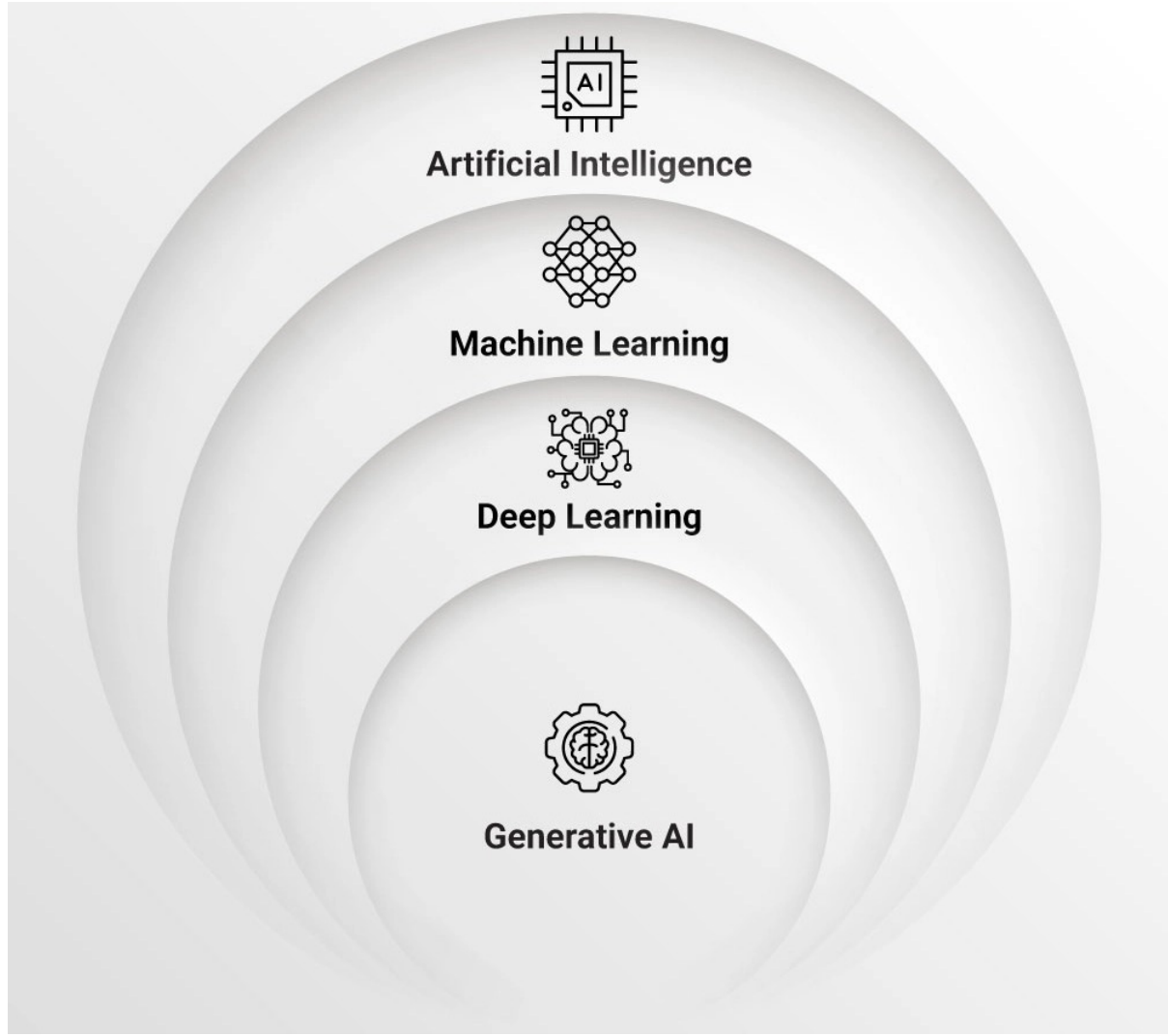
- Clinical Documentation
- Drug Discovery & Research
- Patient Engagement Chatbots
- Medical Imaging Analysis

Example

An **AI-powered patient summary generator**

that reads EHRs and produces

concise reports for doctors before consultations.


AI
Artificial Intelligence
Machine Learning
Deep Learning
Generative AI

Artificial Intelligence (AI)

AI is the broad science of

Making Machines *think*

Act like humans.

Workflow Explained

Define the Problem

Clearly identify the issue to be addressed, whether predicting an adverse event or another challenge.

Data Collection

Gather relevant information such as clinical records or manufacturing logs to support informed analysis.

Workflow Explained

Analytical Reasoning

Apply rule-based or statistical methods to extract insights and patterns from the collected data.

Informed Decision Making

Use analytical insights to recommend actions or optimise outcomes at each workflow stage.

Machine Learning

ML IS A **SUBSET OF AI** WHERE

MACHINES **LEARN FROM DATA**

INSTEAD OF BEING EXPLICITLY PROGRAMMED.

Predicting Drug Response

Personalised Treatment Factors

Age, weight, dosage, and biomarkers are key features considered to predict patient drug response and personalise treatment.

Improving Effectiveness and Safety

Personalised plans enhance drug effectiveness and patient safety, reducing adverse reactions and optimising outcomes.

Predicting Drug Response

Role of Advanced Analytics

Machine learning models analyse large datasets to identify patterns, helping clinicians make data-driven prescribing decisions.

How it works?

FEED IN DATA

TRAIN A MODEL

MAKE PREDICTIONS
ON NEW DATA

Efficient Machine Learning Workflow

Data Preparation

Collecting and cleaning data ensures it is accurate and relevant for machine learning models to learn effectively.

Training and Testing

Splitting the dataset enables unbiased evaluation, and training algorithms helps identify valuable patterns.

Efficient Machine Learning Workflow

Model Validation and Deployment

Validating model performance is key before deployment, ensuring real-world applications benefit from accurate predictions.

Types of Output Predictions

Numerical Outcomes

These predictions result in continuous values, such as forecasting house prices or temperatures based on input data.

Categorical Outcomes

Categorical predictions assign data to one of several groups, for example classifying animal species based on their attributes.

Types of Output Predictions

Binary Outcomes

Binary predictions are a form of categorical outcome with only two possible results, such as yes or no decisions.

Deep Learning (DL) – Neural Networks with Many Layers

Subset of ML that

uses **artificial neural networks**

(like the human brain)

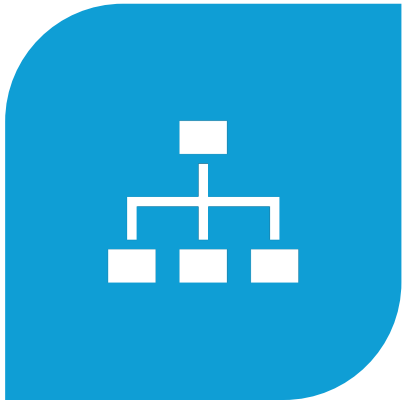
with **multiple layers**

to solve complex problems.

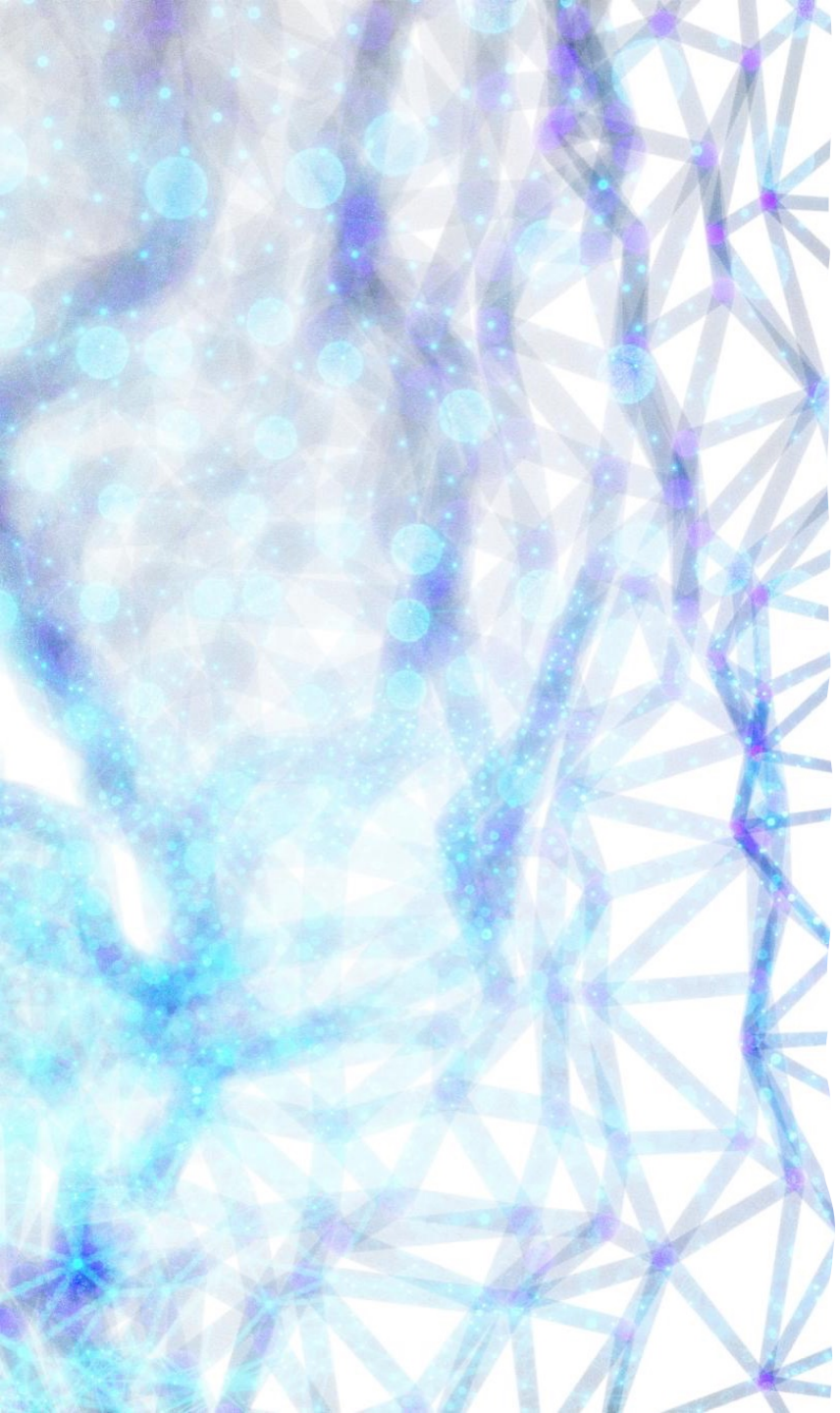
Deep Learning Goal

AUTOMATICALLY EXTRACT

COMPLEX FEATURES &



REPRESENTATIONS.



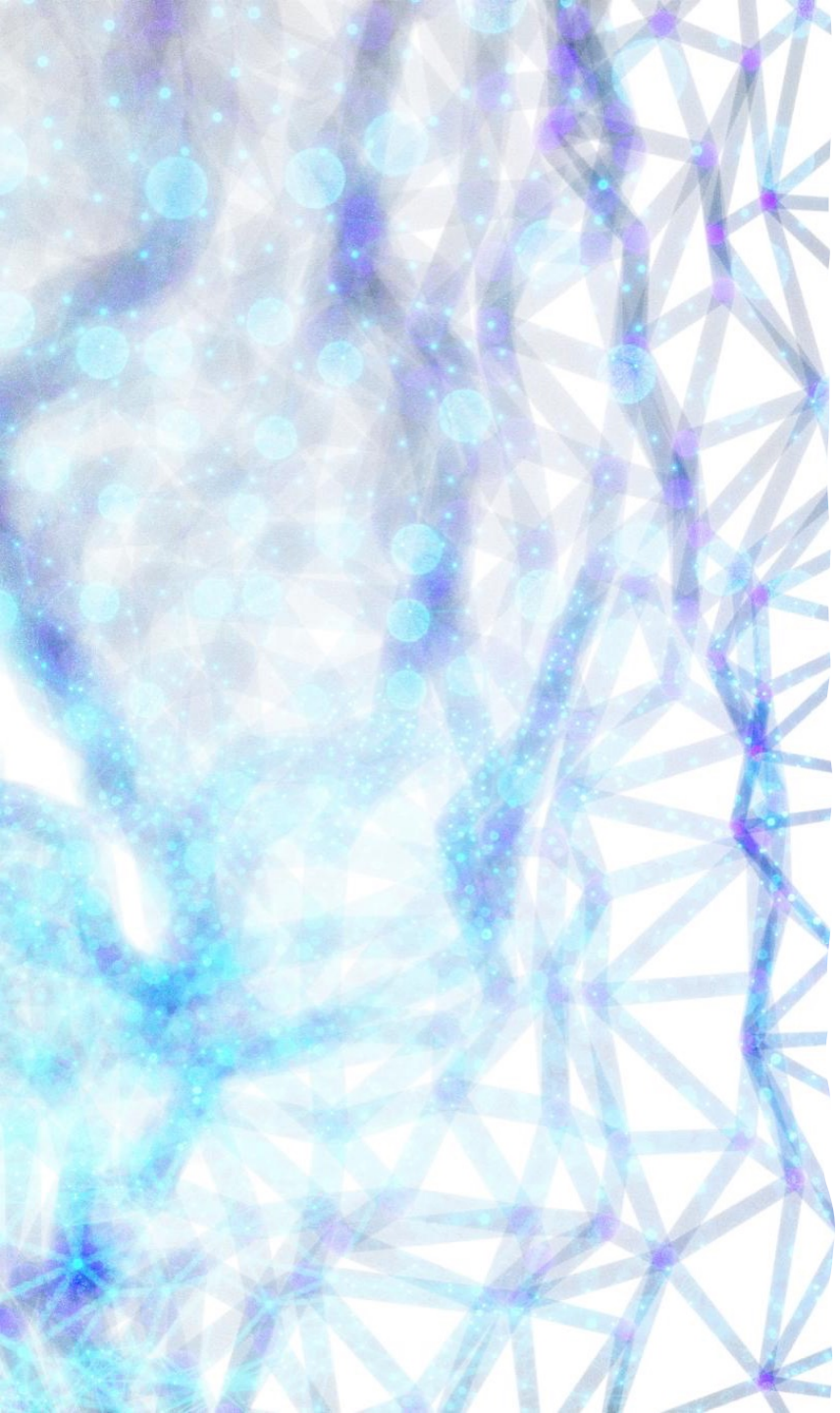
Deep Learning Goals Unveiled

Automatic Feature Extraction

Deep learning models extract complex features from raw data, minimising reliance on manual feature engineering.

Hierarchical Pattern Learning

Neural networks build layered representations, capturing intricate data patterns for improved decision-making..



Deep Learning Goals Unveiled

High-Level Abstraction for Tasks

Deep learning enables high-level abstraction, improving performance in image recognition, speech, and language understanding tasks.

Deep Learning Example in Pharma

- Identifying molecules with desired properties
- from chemical images.
- Analyzing MRI scans to detect anomalies.
- Automating transcription of doctor-patient interactions.

Deep Learning in Pharma

Accelerating Drug Discovery

Deep learning models identify promising molecules from chemical structures, speeding up the drug discovery process for new treatments.

Enhanced Medical Imaging Analysis

Deep learning analyses MRI scans to detect subtle anomalies, supporting earlier and more accurate medical diagnoses.

Deep Learning in Pharma

Automating Clinical Documentation

Automated transcription of doctor-patient conversations using deep learning improves record-keeping and streamlines clinical workflows.

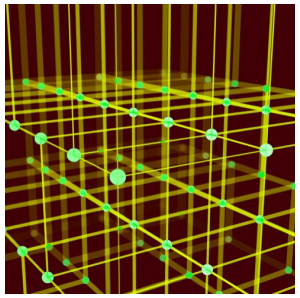
Deep Learning Workflow

- Data (images/text/audio)
- Neural Network (multiple layers: input → hidden → output)
- Forward propagation (inputs to outputs)
- Backpropagation (error correction)
- Model improves accuracy with training

Deep Learning Workflow Explained

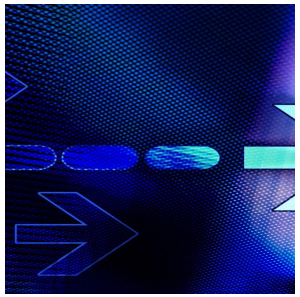
Data Collection

The deep learning workflow starts with gathering data such as images, text, or audio relevant to the problem.



Neural Network Architecture

A neural network with input, hidden, and output layers processes the collected data for pattern recognition.



Forward and Backpropagation

Forward propagation produces predictions, while backpropagation adjusts weights to reduce errors and improve accuracy.

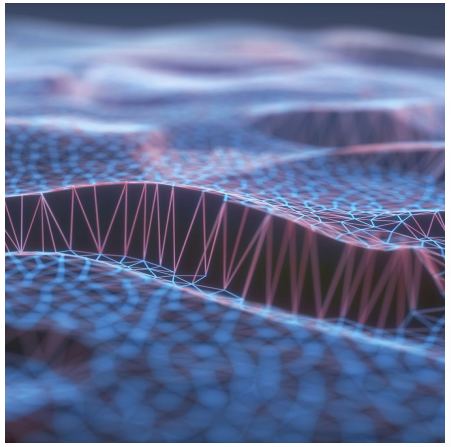
Deep Learning Output

High-dimensional representations

Feature embeddings

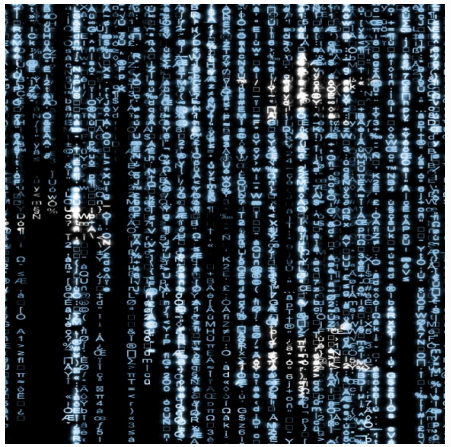
Probability maps

Deep Learning Output Explained



High-Dimensional Representations

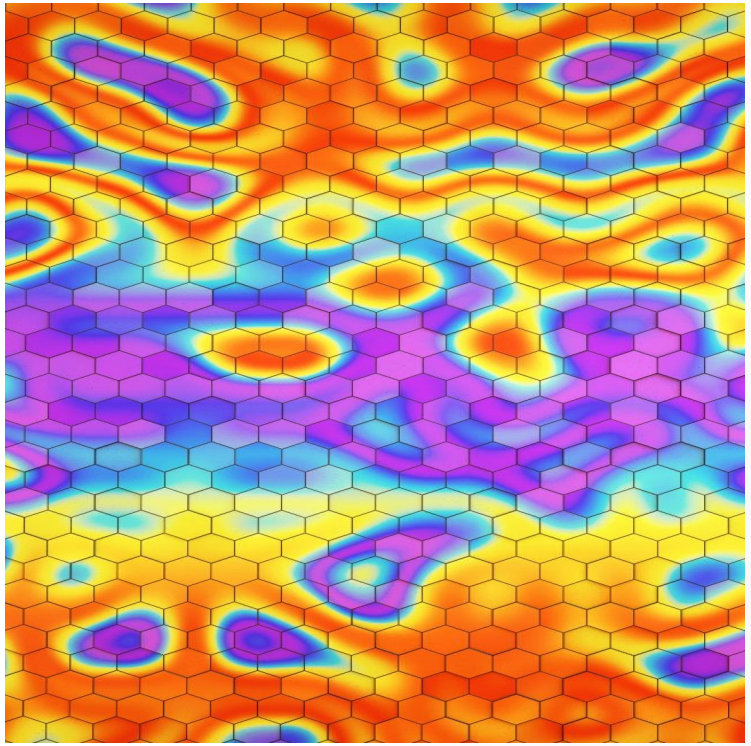
Deep learning models extract complex patterns, producing high-dimensional representations for advanced data understanding and processing.



Feature Embeddings

Feature embeddings encode input data as dense vectors, allowing efficient similarity comparisons and structured data representation.

Deep Learning Output Explained



Probability Maps

Probability maps show spatial or categorical probabilities, highlighting important regions or classes in segmentation and classification tasks.

Strengths

Handles huge datasets

Great for unstructured data (text, images, speech)

ANN, RNN, CNN, and FFN

Aspect	ANN (Artificial Neural Network)	FFN (Feedforward Neural Network)	RNN (Recurrent Neural Network)	CNN (Convolutional Neural Network)
Basic Structure	General term for networks of interconnected neurons	A type of ANN where data flows in one direction (input → output)	A type of ANN with recurrent (loop) connections	A type of ANN using convolution and pooling layers

ANN, **RNN**, **CNN**, and **FFN**

Aspect	ANN	FFN	RNN (	CNN
Main Use Case	General-purpose learning model	Classification/regression on fixed-size data	Sequential/time-series data (e.g., text, speech)	Image and spatial data (e.g., vision tasks)

ANN, **RNN**, **CNN**, and **FFN**

Aspect	ANN	FFN	RNN (	CNN
Key Components	Neurons, layers, activation functions	Input, hidden, output layers	Hidden state, recurrent connections	Convolutional layers, pooling, filters

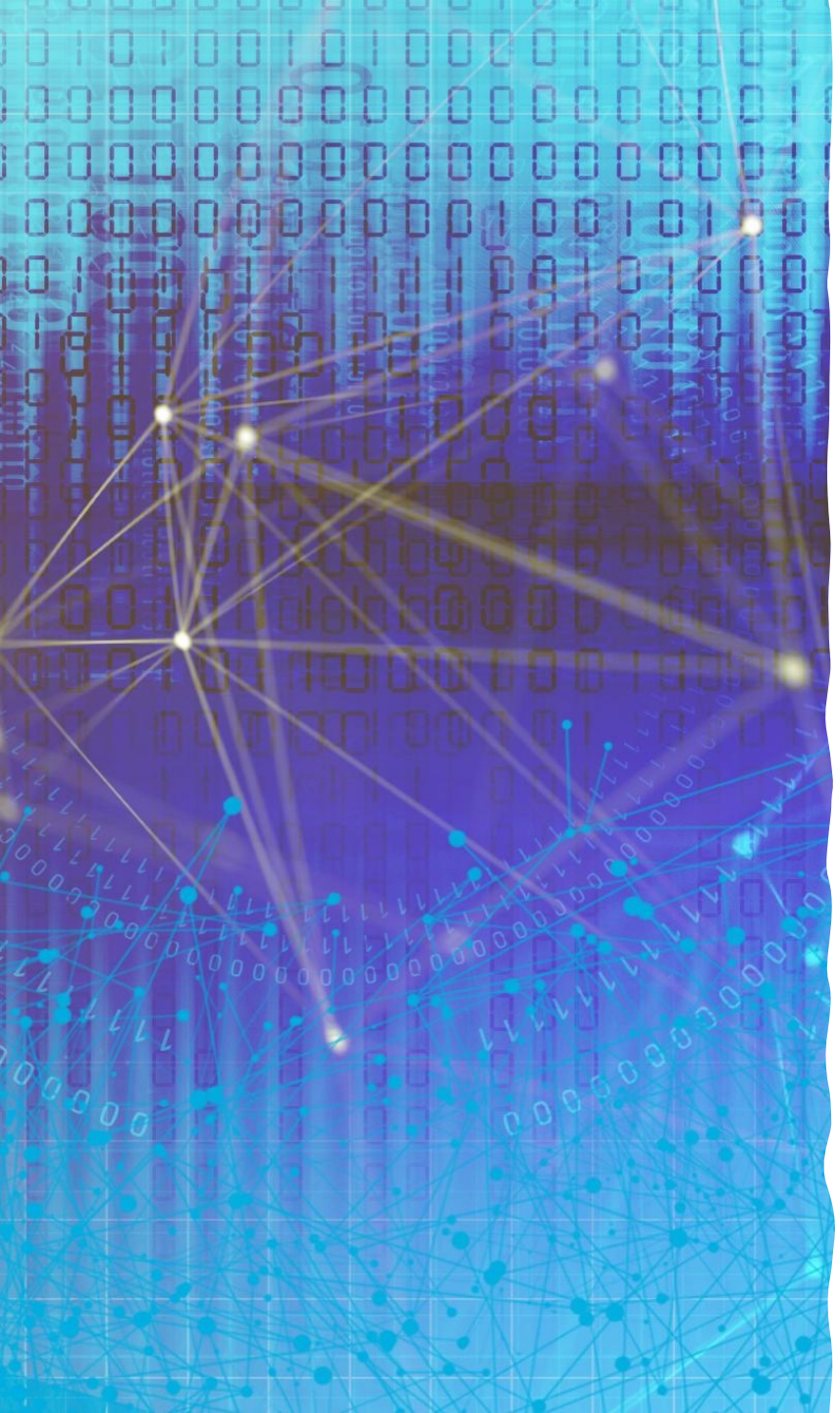
Generative AI

A branch of DL that focuses on

generating new content

like text, images, music, or code

by learning from existing data.

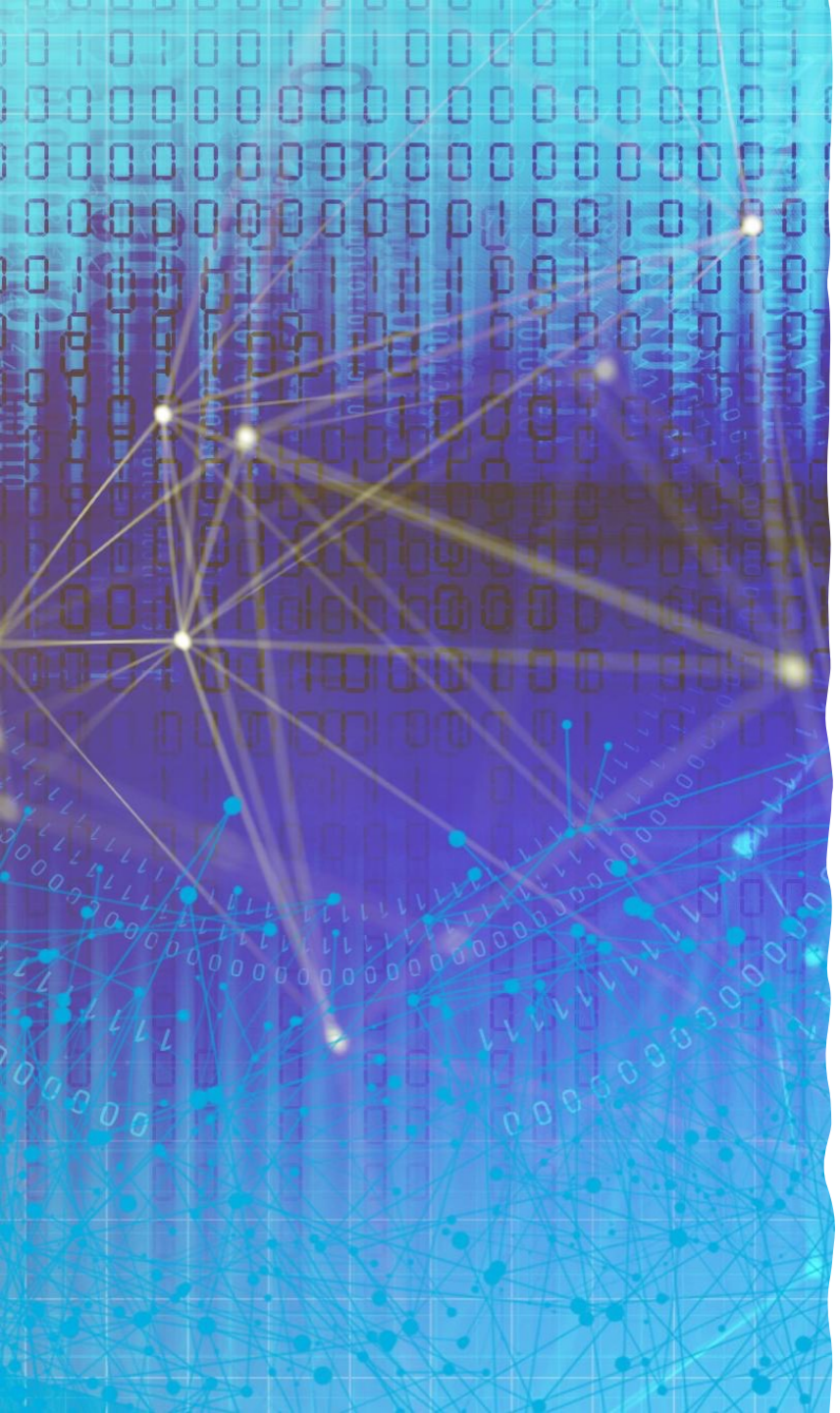


Generative AI

Generative AI analyses data and produces new, similar content, going beyond traditional data processing methods.

Learning from Patterns

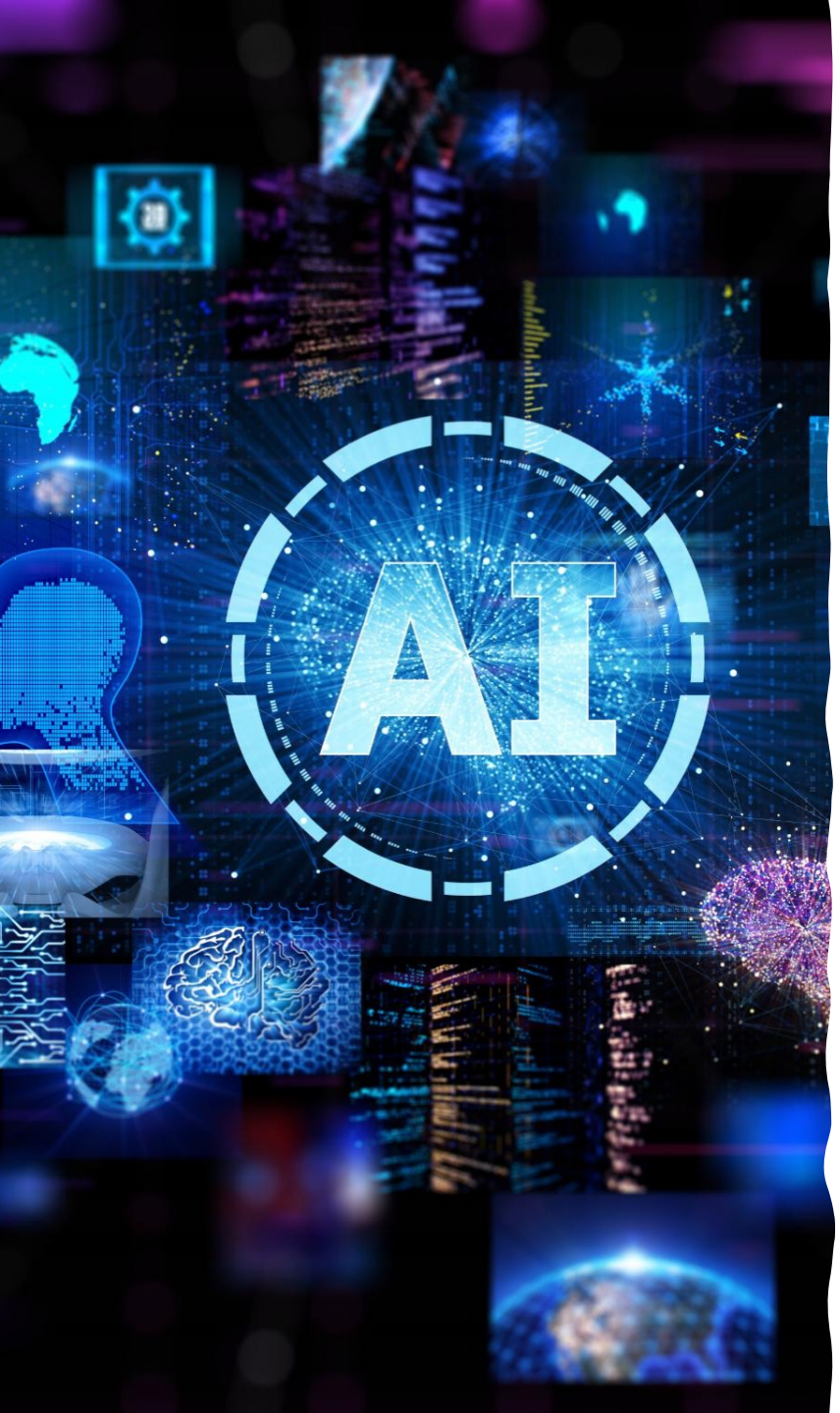
By learning patterns from large datasets, generative AI can create realistic images, text, or sounds that mimic human creativity.



Generative AI

Powering Modern Tools

Generative AI enables innovations like chatbots, image generators, and content creation systems, transforming creative and analytical work.



Generative AI Foundations

Foundation Model Training

Generative AI models are trained on massive datasets, allowing them to recognise complex patterns in language and other data.

Multimodal Content Generation

Generative AI can create new text, images, audio and molecular designs for varied applications across many industries.

Industry Applications

Generative AI enables innovation in fields such as media, healthcare, research, and design through advanced content generation.

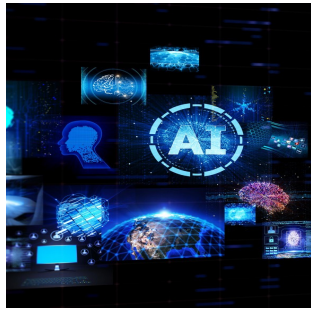
Generative AI Objectives

Enhancing Creativity and Productivity

Produces realistic, valuable content from user prompts, boosting creativity and productivity in various fields.

Advanced Machine Learning Models

Use sophisticated machine learning models to interpret inputs and generate tailored text, images, or audio.



Relevant and Effective Applications

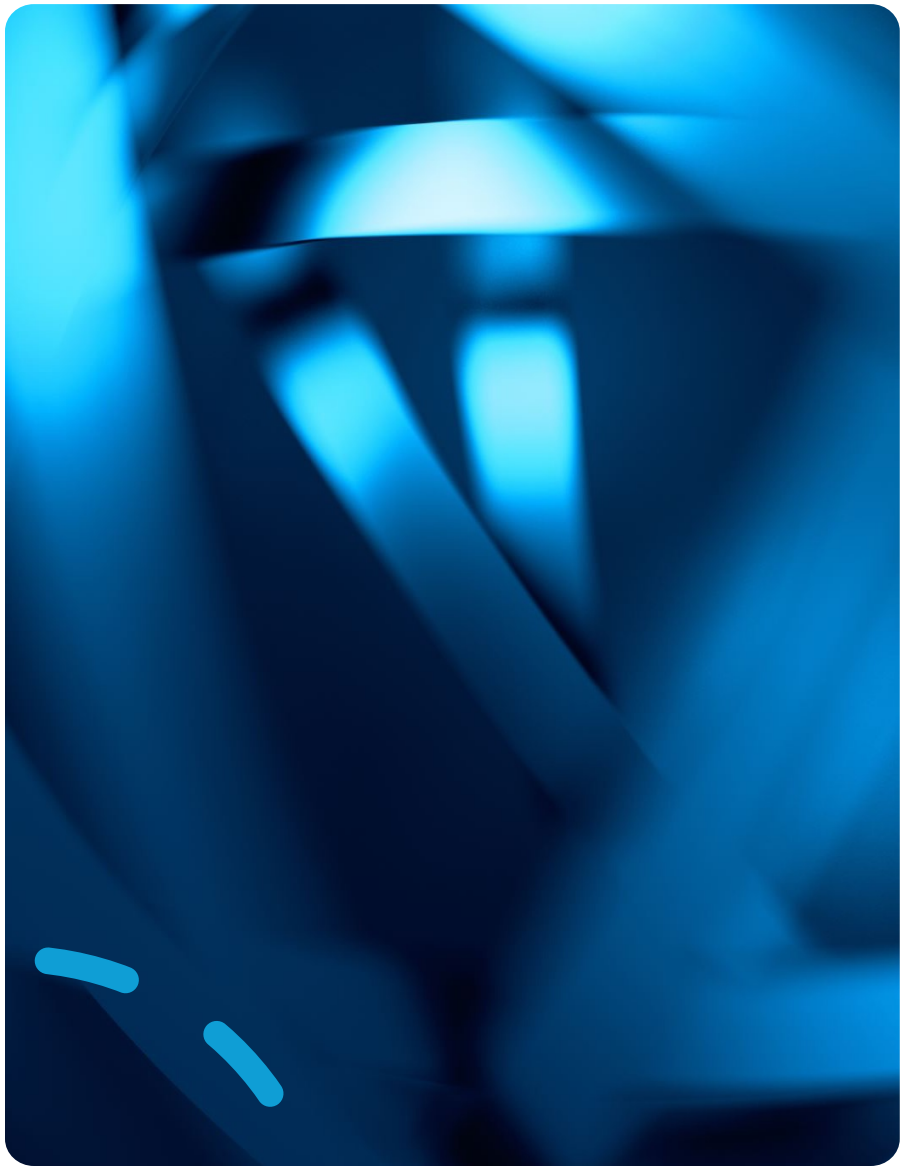
To provide content that is both relevant and effective for practical, real-world applications.

Example in Pharma

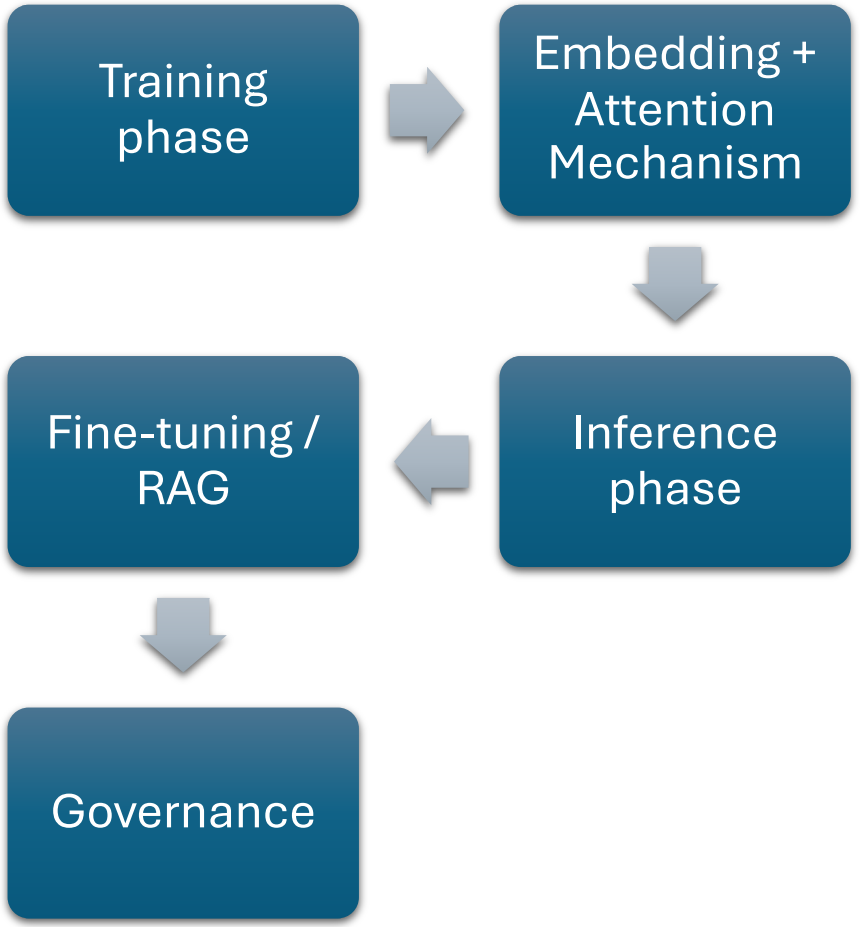
Writing clinical trial summaries automatically.

Designing new molecular compounds.

Generating synthetic (non-PII) patient data for model training.



Workflow


Training phase
Embedding + Attention Mechanism
Inference phase
Fine-tuning / RAG
Governance

Training phase

MODEL LEARNS PATTERNS FROM

BILLIONS OF TEXT/MOLECULE SEQUENCES.

Embedding + Attention Mechanism

Captures relationships

between

words or atoms.

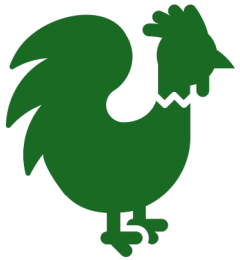
Inference phase

GIVEN A *PROMPT*,

MODEL PREDICTS
NEXT TOKENS

TO CREATE
TEXT/IMAGES.

Fine-tuning / RAG



Model refined for domain-specific pharma use-cases.

Governance

AI outputs monitored for

accuracy, bias, compliance.

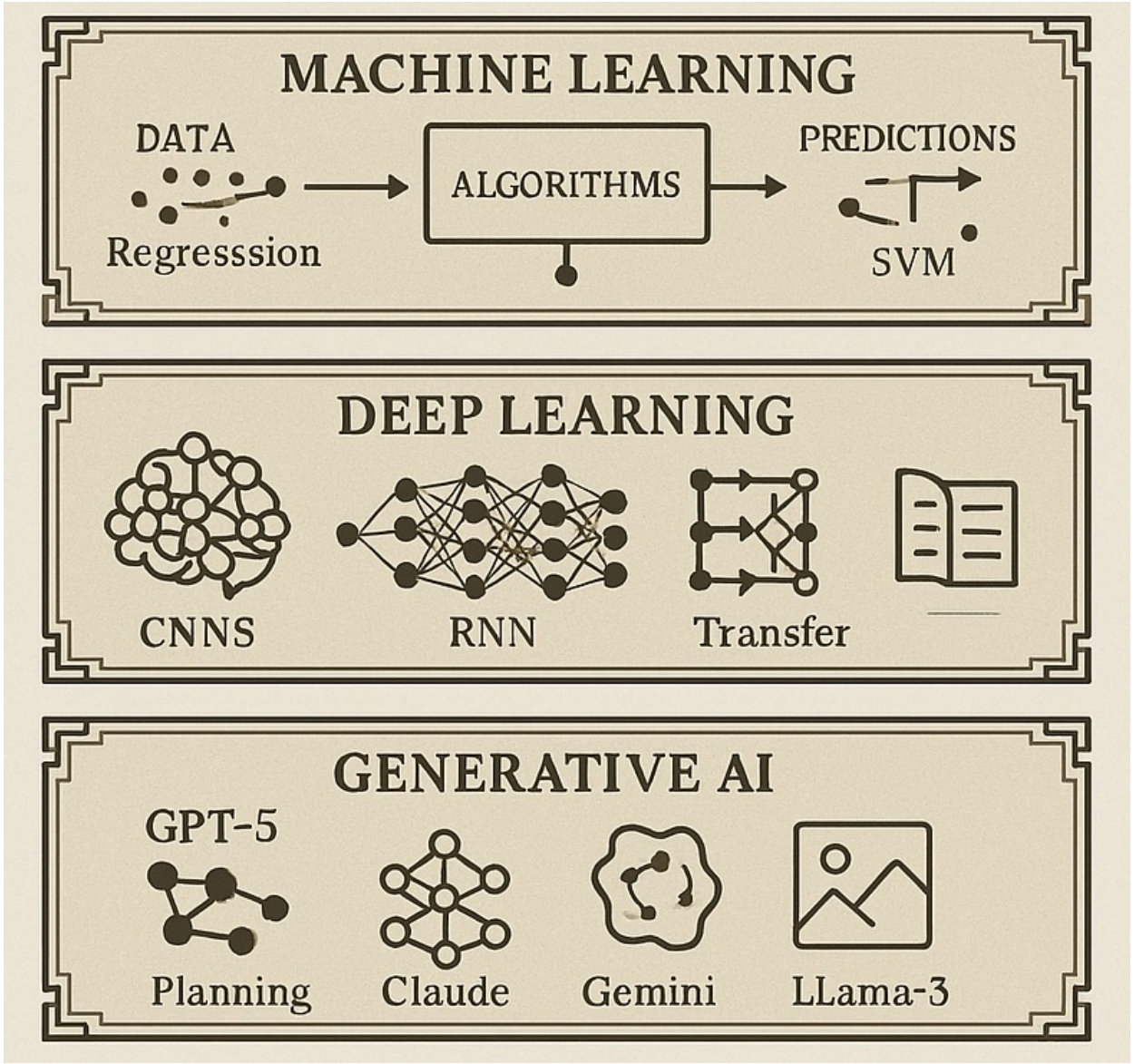
Output

New text, molecules, or visuals

generated from learned knowledge.

Combined Hierarchy

Level	Key Idea	Example in Pharma	Data Type
AI	Systems that mimic human thinking	QA bots for patient queries	Rules / logic
ML	Learn from data	Predict patient dropout	Structured data
DL	Learn from raw signals	Detect tumor from MRI	Images / audio / text
GenAI	Create new content	Generate clinical report summaries	Text / image / molecule


MACHINE LEARNING
DATA
Regresssion
ALGORITHMS
PREDICTIONS
SVM
DEEP LEARNING
CNNS
RNN
Transfer
GENERATIVE AI
GPT-5
Planning
Claude
Gemini
LLama-3

Large Language Models

MASSIVE GENERATIVE AI MODELS

TRAINED ON **BILLIONS OF TEXT EXAMPLES**

TO UNDERSTAND AND GENERATE

HUMAN-LIKE LANGUAGE.

Examples

GPT (OpenAI)

Claude (Anthropic)

Gemini (Google)

LLaMA (Meta)

AI vs ML vs DL vs GA vs LLMs

Feature	AI	ML	DL	GA	LLMs
Definition	Machines mimicking human intelligence	Learning from data	Neural networks with layers	Generate content from data	Giant models trained on text
Input	Rules / logic	Data + Labels	Lots of data (text, images)	Text, images, music	Massive text datasets
Output	Actions / decisions	Predictions / labels	Features / patterns	New content (text, image)	Fluent, context-aware language

Short Video

- 🎬 AI vs ML vs DL vs Generative AI – Quick Visual Explanation
- (Useful for learners and corporate teams)

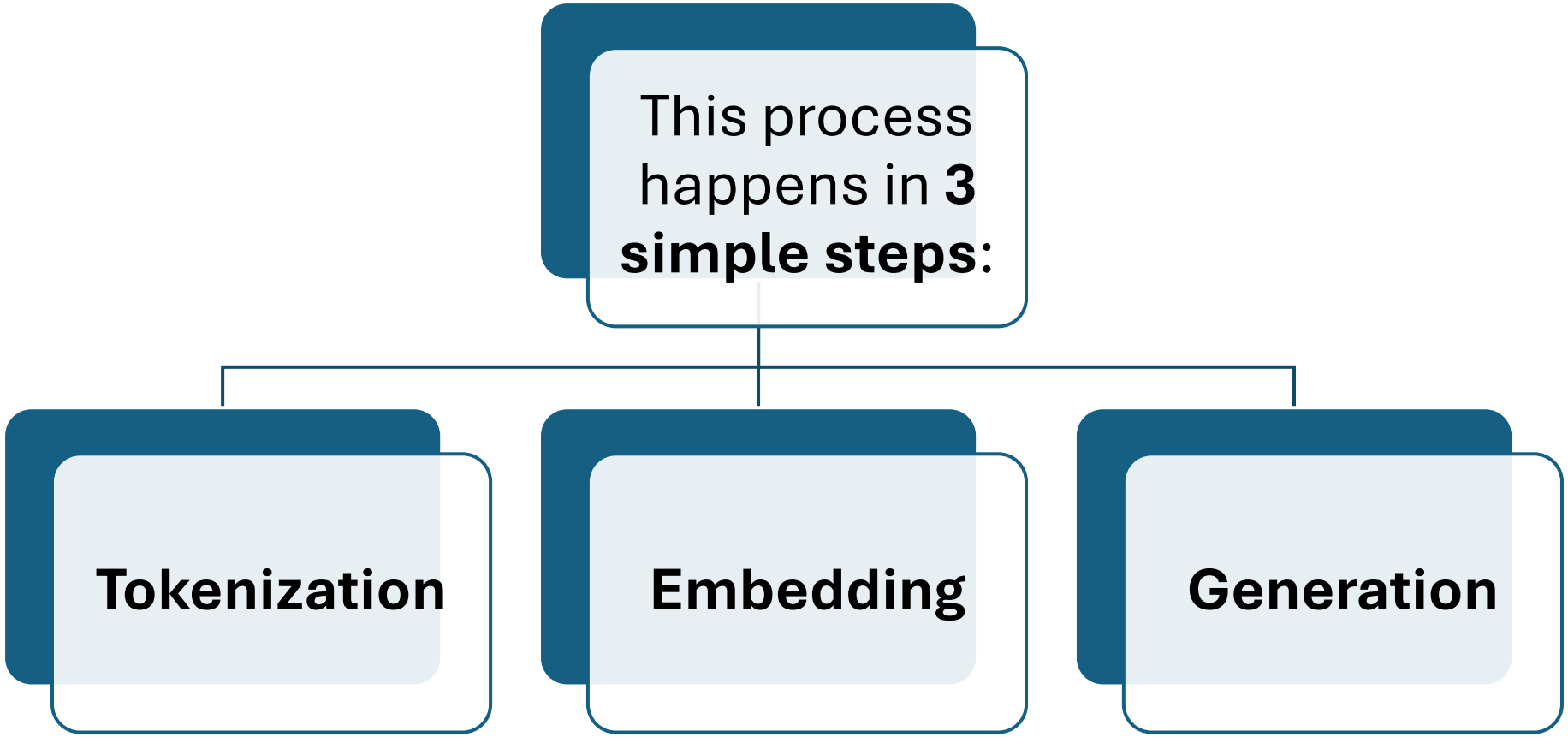
Neural Generative Modeling

TOKENIZATION: BREAKING DOWN INPUT TEXT

EMBEDDING: REPRESENTING LANGUAGE IN VECTOR SPACE

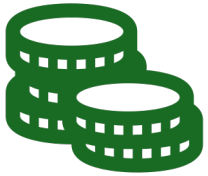
GENERATION: SAMPLING AND DECODING TECHNIQUES

What is Neural Generative Modeling?



Step 1: Tokenization

Break the input sentence



into **smaller units (tokens)**

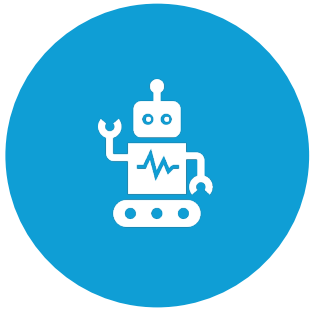
like words or parts of words

that a computer can understand.

Step 2: Embedding

CONVERT EACH TOKEN INTO A

VECTOR (A LIST OF NUMBERS)



SO THE MODEL CAN “UNDERSTAND”

ITS **MEANING AND CONTEXT** MATHEMATICALLY.

Step 3: Generation

The model generates a smart response

using **decoding techniques** like:

Greedy Search

choosing most probable word at each step

3. Large Language Models (LLMs)

Overview of popular models:

- GPT (OpenAI)
- Claude (Anthropic)
- Gemini (Google)
- LLaMA (Meta)

What Are LLMs?

AI SYSTEMS TRAINED

ON MASSIVE
AMOUNTS OF DATA

BOOKS, MANUALS,
WEB CONTENT, CODE

The Battle of AI Models: GPT-4 vs Gemini vs Claude vs LLaMA

GPT-4 Gemini Claude LLaMA

GPT vs. Gemini vs. Claude vs Llama

Which AI is better in 2025?

https://www.youtube.com/watch?v=5KiDabAa9JY

Core Strengths & Use Cases

ChatGPT-5

Best overall performer in coding,

creative writing, and

multimodal interactions.

Excelled in calculus

https://platform.openai.com/docs/overview

Core Strengths & Use Cases

Google Gemini (2.5 Pro/Flash)

Strong in multimodal context

(text+image+audio/video),

reasoning-heavy tasks, and

cost-effective free usage.

https://gemini.google.com/app

Core Strengths & Use Cases

Anthropic Claude (4 Sonnet / Opus)

Tops safety, ethical alignment,

long-context reasoning

best accuracy in reading comprehension

without hallucinations.

https://claude.ai/onboarding?returnTo=%2F%3F

Core Strengths & Use Cases

Meta Llama 4 (Scout & Maverick)

Competitive open-source option,

excels in reasoning and

massive-context tasks,

rivaling GPT-4o in benchmarks.

Meta Llama 4 (Scout & Maverick)

Meta_UI Interface

https://www.meta.ai/?utm_source=ai_meta_site&utm_medium=web&utm_content=AI_nav&utm_campaign=06112025_moment

Evolution of GPT Models → GPT-5

Generative Pretrained Transformers (GPT)

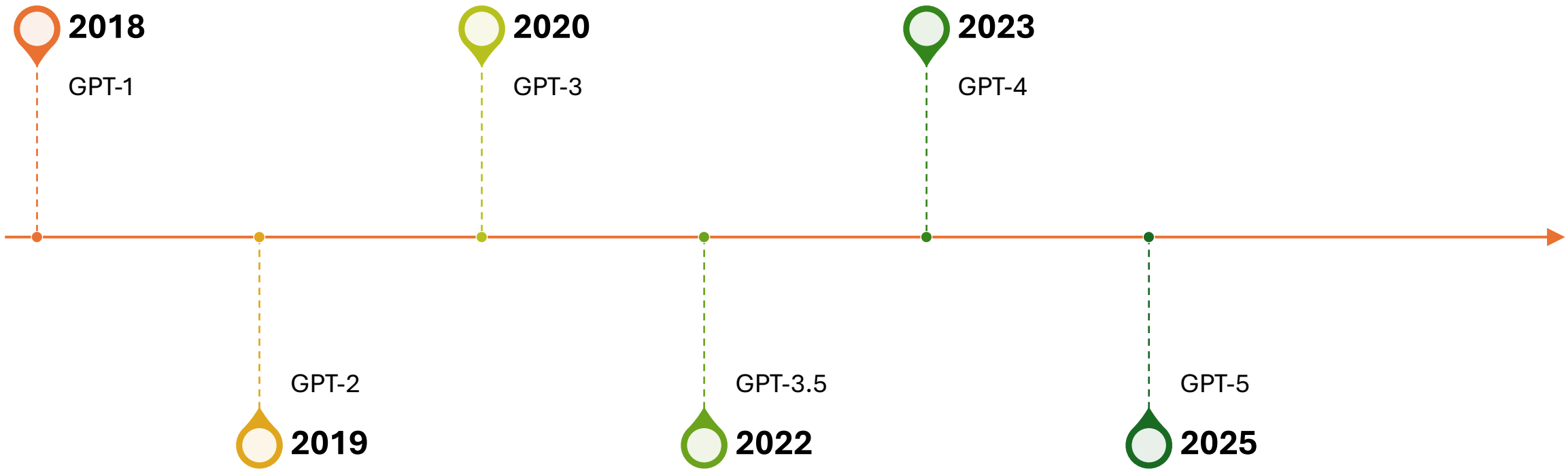
are a family of large language models

developed by **OpenAI**.

Evolution of GPT Models

Each new version represents a leap in **scale, architecture, reasoning, and enterprise applicability.**

Evolution of GPT Models



GPT-1 (2018)

The proof-of-concept.

Introduced the transformer architecture

for natural language understanding.

Trained on ~117M parameters.

GPT-2 (2019)

Showed strong text generation capabilities.

1.5B parameters.

Famous for being partially withheld

due to concerns about misuse.

GPT-3 (2020)

Breakthrough in size (175B parameters).

Demonstrated zero-shot, few-shot learning.

Powered the first wave of real-world LLM applications.

GPT-3.5 (2022)

Optimized variant of GPT-3

with reinforcement learning from human feedback (RLHF).

Provided better dialogue handling

the engine behind **ChatGPT’s early versions**.

GPT-4 (2023)

Huge improvement in reasoning,

multi-modal inputs (text + images),

reduced hallucinations,

stronger coding and enterprise reliability.

GPT-5 (2025)

Current state-of-the-art.

Expanded reasoning depth, memory,

multi-modal

text, images, audio, video

GPT-5 (2025)

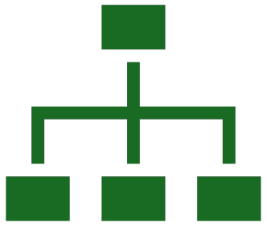
IMPROVED **FACTUAL ACCURACY**,

TOOL ORCHESTRATION,

AGENTIC AI SUPPORT.

GPT-5 (2025)

Designed for
enterprise integration



with **compliance, governance, and**

security in mind.

Comparative Chart: GPT Evolution

Feature / Model	GPT-3 (2020)	GPT-3.5 (2022)	GPT-4 (2023)	GPT-5 (2025)
Parameters	175B	~175B (optimized)	~1T est.	Multi-trillion scale (sparse/expert models)
Core Capability	Few-shot learning	Conversational fine-tuning (RLHF)	Advanced reasoning, multimodal	Deep reasoning, planning, multi-modal (text, images, audio, video)

Comparative Chart: GPT Evolution

Feature / Model	GPT-3 (2020)	GPT-3.5 (2022)	GPT-4 (2023)	GPT-5 (2025)
Context Window	2K–4K tokens	8K tokens	32K–128K	**Up to millions (long-context memory)**
Use Cases	Chatbots, Q&A, apps	Conversational AI	Enterprise copilots, coding, multimodal	**Agentic AI, enterprise copilots, compliance-ready AI assistants**

Comparative Chart: GPT Evolution

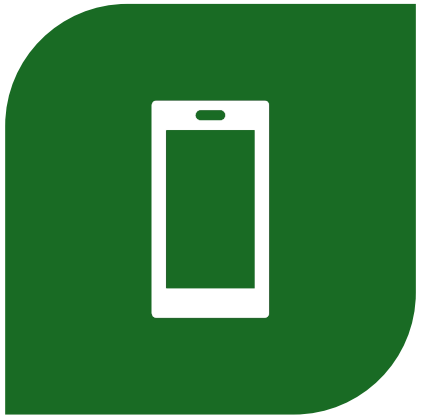
Feature / Model	GPT-3 (2020)	GPT-3.5 (2022)	GPT-4 (2023)	GPT-5 (2025)
Hallucinations	Moderate	Lower than GPT-3	Reduced further	**Minimal with retrieval + reasoning**
Fine-Tuning	Yes	Optimized	Yes, domain-specific	**Advanced domain adaptation with guardrails**

Comparative Chart: GPT Evolution

Feature / Model	GPT-3 (2020)	GPT-3.5 (2022)	GPT-4 (2023)	GPT-5 (2025)
Enterprise Focus	Startup adoption	Developer adoption	Enterprise pilots	**Enterprise-first (governance, observability, integration)**

Key Takeaway

GPT-5 REPRESENTS A SHIFT FROM



SMART ASSISTANT TO

ENTERPRISE-READY AGENT.

Key Takeaway

It not only generates but

reasons, plans, and

integrates securely

into enterprise ecosystems.

GPT-4 vs.GPT-5 parameters

GPT-4 Parameters (Developer-Facing Controls)

GPT-5 Parameters (Higher-Level Controls)

GPT-4 parameters

- Temperature
- Top-p (nucleus sampling)
- Max tokens
- Frequency penalty / Presence penalty

Temperature

Controls randomness in output.

Range: 0 → 2

lower = deterministic,

higher = more creative

Temperature

temperature=0

factual, repeatable answers

temperature=1.2

more diverse text.

Top-p (nucleus sampling)

Probability mass of tokens

When an LLM generates text,

it looks at the **probability distribution**

of all possible next words.

Top-p (nucleus sampling)

Top-p small (0.3–0.5) →

only the *very top words* are considered

More deterministic.

Top-p (nucleus sampling)

Top-p large (0.9–1.0)

wider set of possible words

More variety.

Top-p (nucleus sampling)

Top-p = 1.0

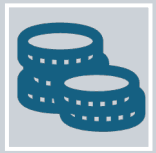
Means “no cutoff,” i.e.,

consider all tokens

(similar to default).

Max tokens

Hard cap on how long the model’s output can be.



Example: max_tokens=500

ensures the reply won’t exceed 500 tokens.

Frequency penalty / Presence penalty

Frequency penalty

Reduces repeated phrases.

Presence penalty

Encourages introducing *new topics*.

GPT-5 Parameters (Higher-Level Controls)

- Verbosity
- Directness / Formality
- Reasoning depth
- Context weighting

Verbosity

Controls the *level of detail*.

verbosity=low → concise summary,

verbosity=high → step-by-step breakdown with explanations.

Directness / Formality

Lets you specify tone

Conversational vs. Professional

Directness / Formality

directness=formal
for enterprise docs

directness=casual
for friendly
explanations.

Reasoning depth

Influences how much structured reasoning, decomposition, and self-explanation the model applies before responding.

Reasoning depth

Example:

reasoning=shallow for quick answers vs.

reasoning=deep for full problem-solving trace.

Context weighting

Ability to prioritize parts of input

system vs user instructions,

recency vs long-term memory

GPT-4 (Classic) vs GPT-5 (Next-Gen)

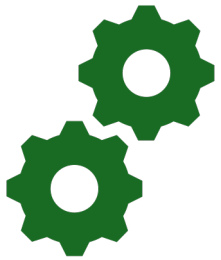
Aspect	GPT-4 (Classic)	GPT-5 (Next-Gen)
Creativity Control	temperature, top_p	creativity (abstracted, tuned with verbosity + reasoning)
Length Control	max_tokens	verbosity (style-based length, not just a cap)
Repetition Control	frequency_penalty, presence_penalty	Integrated into discourse management (automatic coherence)

GPT-4 (Classic) vs GPT-5 (Next-Gen)

Aspect	GPT-4 (Classic)	GPT-5 (Next-Gen)
Tone/Style	Manual prompting	Parameters like formality, directness
Reasoning	Emergent, prompt-engineered	Explicit: reasoning=deep vs reasoning=fast
Context Handling	Fixed token window	Weighted prioritization + extended memory alignment

Summary

GPT-4 is like controlling a car



with manual gears

(temperature, max tokens).

Summary

GPT-5 gives you assisted driving modes

verbosity, reasoning depth, tone

more semantic,

less low-level tweaking.

Transformer Architecture Visual and Evolution

MODEL	YEAR	PARAMETERS	KEY FEATURES
GPT-1	2018	117M	Basic transformer decoder
GPT-2	2019	1.5B	Improved text generation
GPT-3	2020	175B	Few-shot learning
GPT-4	2023	~1T (estimated)	Multimodal capabilities
GPT-5	2025	~10T (estimated)	Advanced reasoning and memory

Model Comparison Table

Model	Key Release Info	Context / Modes	Strengths	Considerations
GPT-4	Was major prior generation; now a baseline	Multimodal (text + some vision)	Very mature, widely used; strong general reasoning	Higher cost; closed weights; might lag newest reasoning
GPT-5	Announced August 2025.	Unified multimodal + agentic capabilities	Big leap: better coding, reasoning, tool-use, “thinking longer when needed”	Proprietary; resource / cost heavy; access may be limited

Model Comparison Table

Model	Key Release Info	Context / Modes	Strengths	Considerations
Claude 4 (Opus / Sonnet / Haiku series)	Latest: “Claude Opus 4.1” (Aug 2025)	Text / vision / agentic workflows	Strong for agentic tasks, coding, reasoning, high context window	Proprietary; may require enterprise licensing; newer so ecosystem somewhat smaller

Model Comparison Table

Model	Key Release Info	Context / Modes	Strengths	Considerations
Gemini 2.5	Google’s “thinking” model (2025)	Multimodal, reasoned mode, tool-use	Good integration in Google ecosystem; flexible for agentic use cases	As with others: cost, closed weights (mostly); may require Google Cloud investments

Model Comparison Table

Model	Key Release Info	Context / Modes	Strengths	Considerations
Mistral Large 2	Open-weight model by Mistral, ~123B parameters, long context window (~128K tokens)	Text / code; can self-host	Big plus: open weights, on-prem deploy possible; long context support	Might still trail in some benchmark reasoning compared to top closed models; infrastructure heavy too

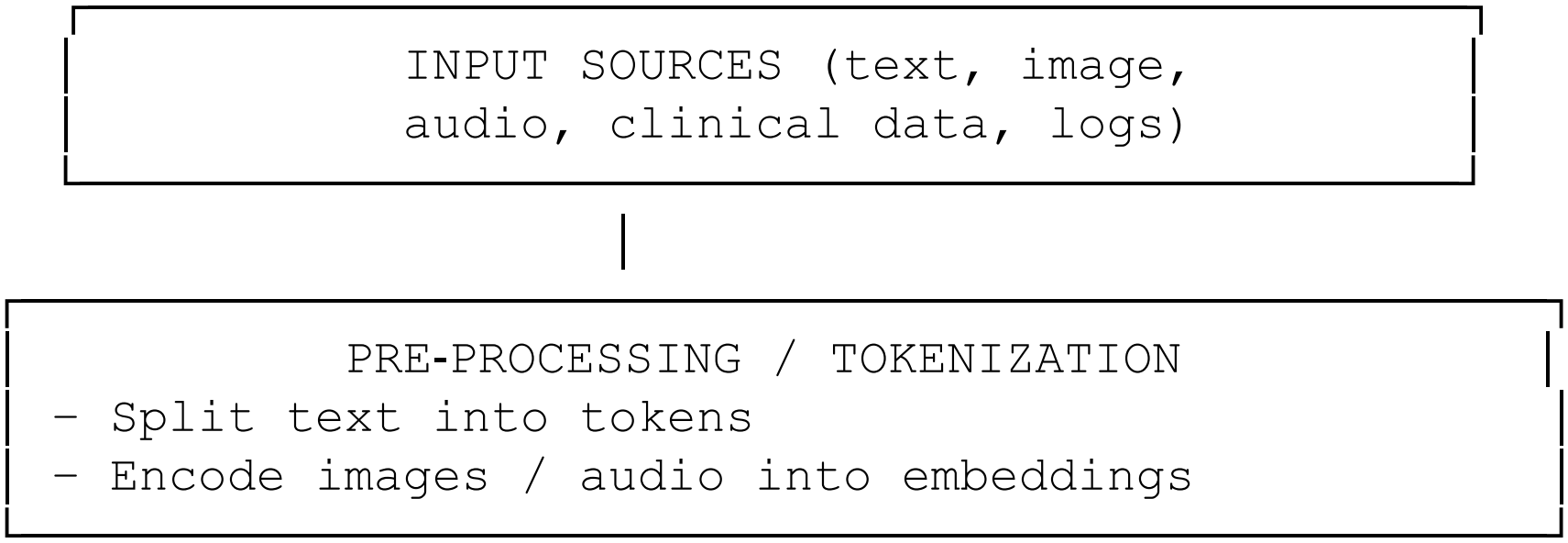
Model Comparison Table

Model	Key Release Info	Context / Modes	Strengths	Considerations
Llama 4	Released April 2025 by Meta; multimodal, MoE (Mixture of Experts) architecture	Text + images + video + multilingual	Open weight; good cost efficiency; flexible deployment	Newer; ecosytem and tooling may still be evolving; model size and performance trade-offs depending variant

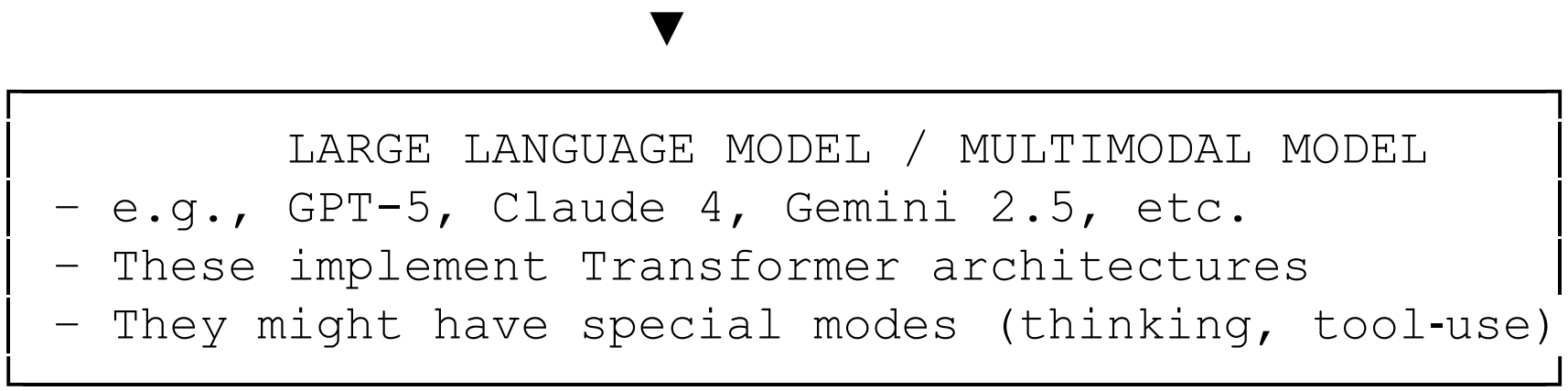
Architecture Diagram

How These Models Fit Into A GenAI Ecosystem

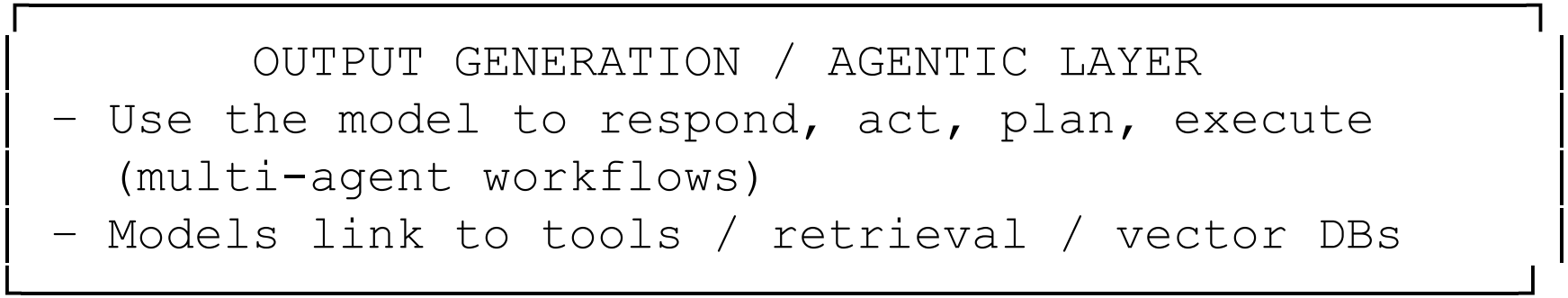
Architecture Diagram



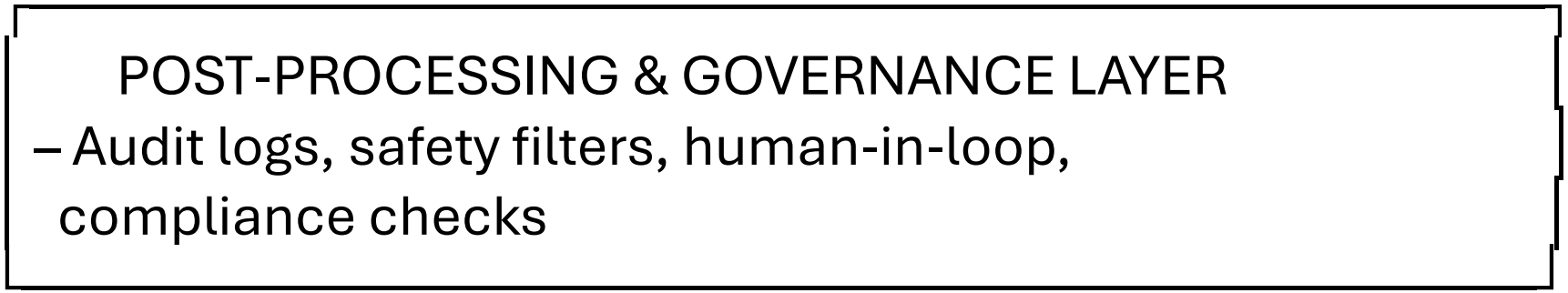
Architecture Diagram



Architecture Diagram



Architecture Diagram



How to Choose In a Pharma Context?

As an AI Architect for pharma

(regulated, high-stakes domain)

Key decision criteria

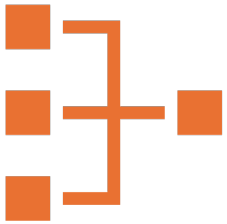
Compliance & Data Residency

If you need on-prem or private cloud

for patient data, open-weight

Mistral, Llama may excel.

Tooling & Agentic Workflows



If you need complex multi-agent orchestration

Models with strong agentic tooling

(GPT-5, Claude 4, Gemini 2.5) are favourable.

Cost vs Performance

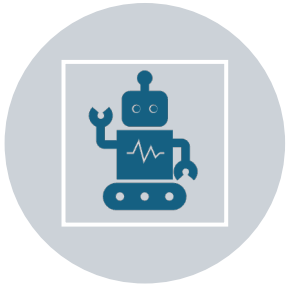
Open models may reduce cost

but could require

more infrastructure/ops.

Long Context / RAG

If you need to ingest many documents



for retrieval + generation,

models supporting long contexts

(Mistral Large 2) are beneficial.

Ecosystem / Support

For integration with enterprise stacks

(Azure, AWS, Google Cloud)

check which model has

strong support in that cloud.

Understanding Parameters, Context Windows, and Reasoning Abilities

Surendra Panpaliya

GKTCS Innovations

Parameters — “The Brain Connections of an AI”

Learned weights

Inside a neural network.

Store what the model has learned

during training patterns, grammar,

facts, reasoning rules

Parameters Example

GPT-3 → 175 billion parameters

GPT-4 → around 1 trillion (estimated)

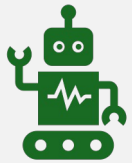
GPT-5 → 1.8 trillion+ (multi-modal, multi-agent optimized)

Parameters Example

- Llama 3 → 70 billion (open-source variant)
- Claude 4 Opus → similar scale to GPT-4
- but optimized for reasoning
- Mistral Large 2 → 123 billion open-weights

What They Do?

Parameters are like the **stored knowledge**



Helps the model:

Know that “paracetamol treats fever.”

What They Do?

Understand grammar:

“The patient *is* stable,” not “The patient *are* stable.”

Link facts across domains:

“High BP + obesity → cardiac risk.”

What They Do?

More parameters

Model can represent **more complex patterns**

usually better understanding,

creativity, and reasoning.

Context Window — “The Short-Term Memory”

The **context window** is how much text

(in tokens) the model can see or

“remember” *at one time*

during a conversation or prompt.

Typical Context Sizes

Model	Context Window	Equivalent in Words	Notes
GPT-4	32 K–128 K tokens	~25K–100K words	Enough for long docs
GPT-5	up to 1 million tokens	~750K words	Great for RAG / multi-doc
Claude 4	200 K+ tokens	~150K words	Excellent for long inputs
Gemini 2.5	1 million tokens	~750K words	Handles entire codebases
Mistral Large 2	128 K tokens	~100K words	Open-weights option
Llama 4	128 K tokens	~100K words	Open, on-prem friendly

Reasoning Abilities — “The Thinking Power”

WHAT IS REASONING?

REASONING IS THE MODEL’S ABILITY TO **CONNECT**

FACTS, PLAN STEPS, AND **DRAW CONCLUSIONS**

INSTEAD OF JUST RECALLING MEMORIZED DATA.

How It Works?

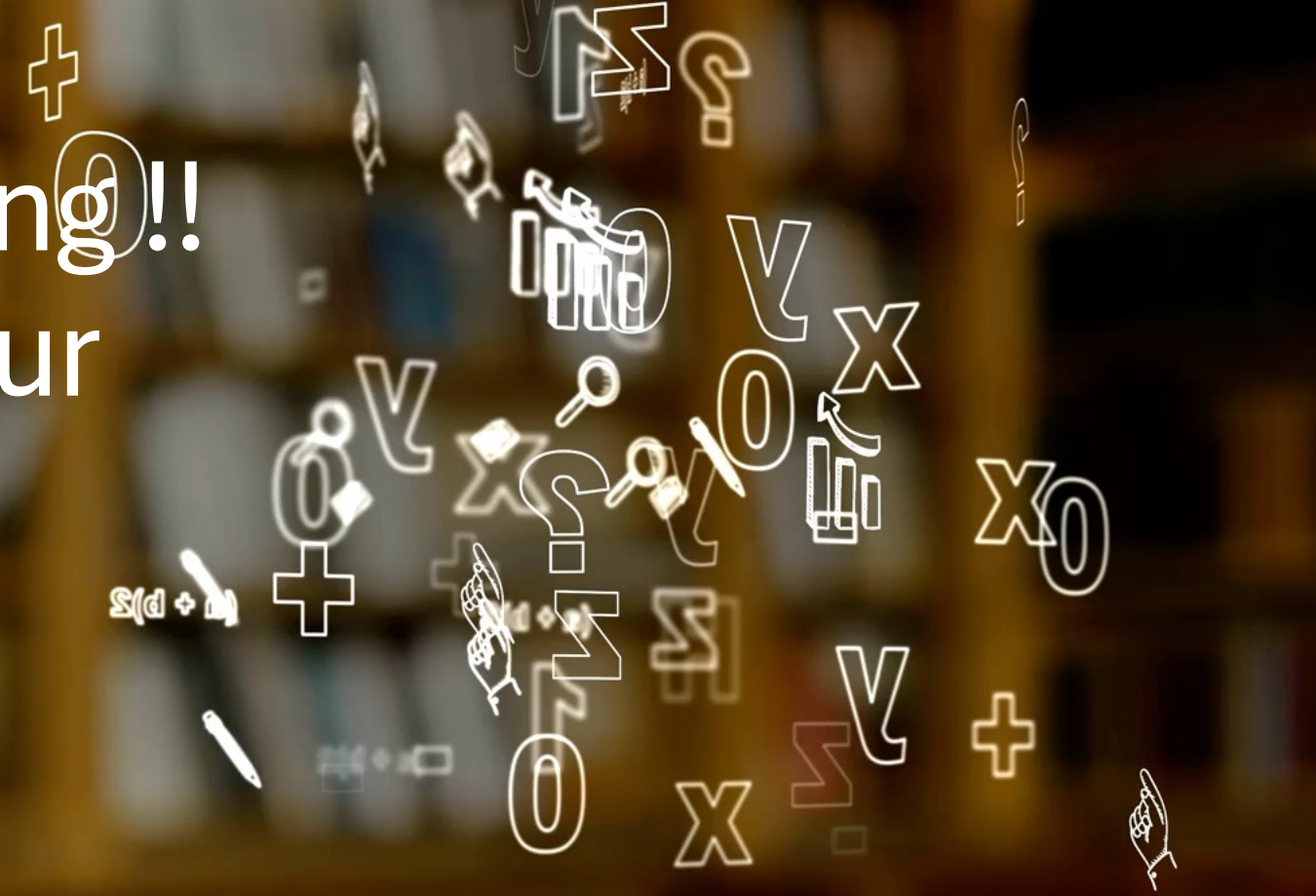
Internally, reasoning = **pattern chaining**.

The model looks at tokens

compares relationships

forms logical connections.

Happy Learning!! Thanks for Your Patience ☺



Surendra Panpaliya

GKTCS Innovations