

Agentic AI

by Qasim Nadeem

<https://qasimnadeem.com>

What is AI?

AI itself is a discipline like Physics, Chemistry deals with the creation of intelligent agents, and systems that can reason, learn, and act autonomously.

Think of Artificial Intelligence (AI) as a way of making a computer act more like a human "brain" and less like a calculator.

While a traditional computer program only does exactly what it's told, AI is designed to learn, adapt, and make decisions based on the information it receives.

To understand AI, it helps to compare it to how we usually use technology:

- **Traditional Programming:** This is like a **recipe**. You give the computer a strict list of instructions (e.g., "If the user clicks this button, turn the screen blue"). If something happens that isn't in the recipe, the computer gets stuck.
- **Artificial Intelligence:** This is like a **student**. Instead of giving it a recipe, you give it thousands of examples. It looks for patterns in those examples to "learn" how to do a task on its own.

How AI "Thinks" (The Three Pillars)

You don't need a computer science degree to understand the three main things AI does:

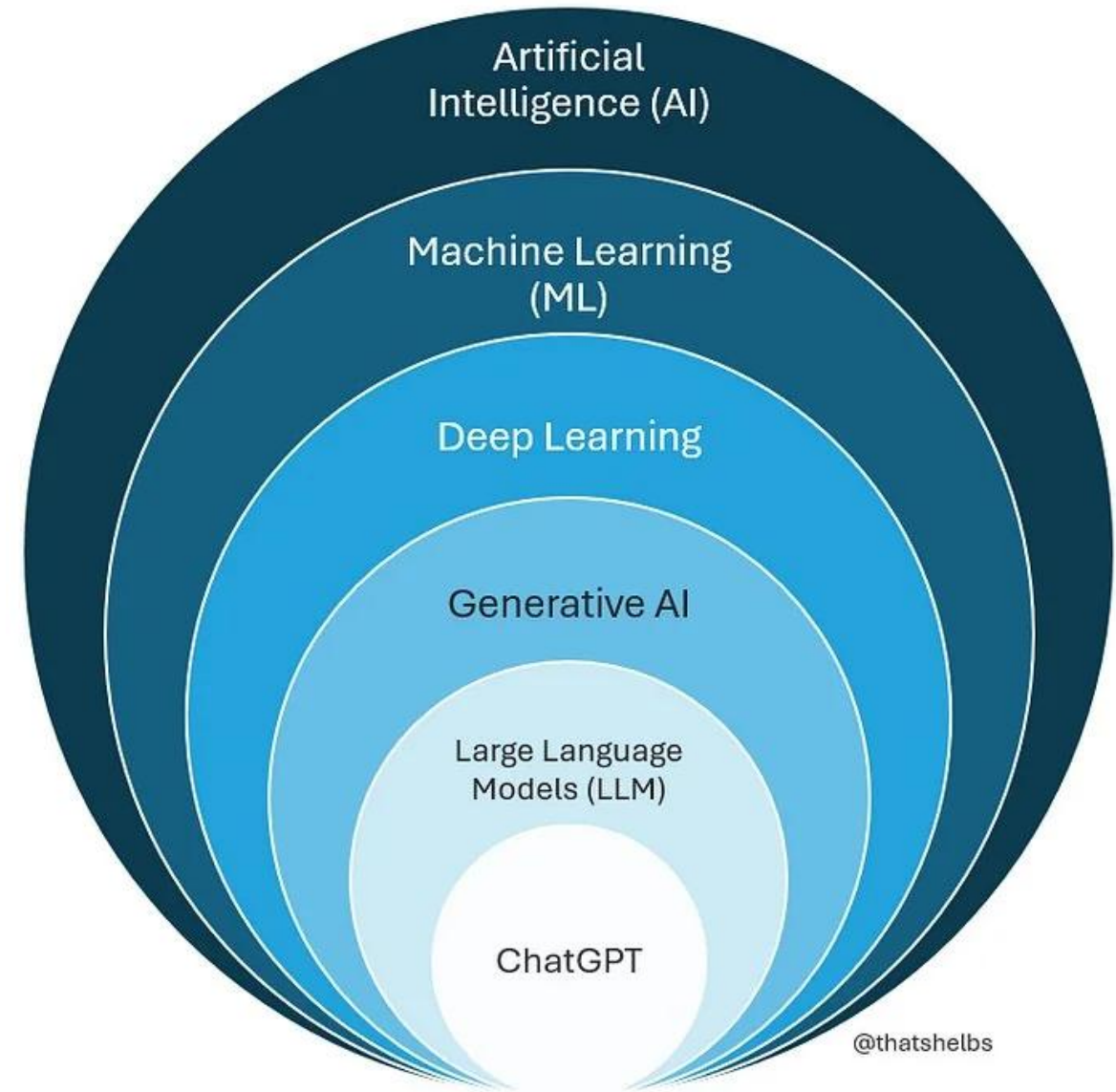
- **Learning:** AI consumes massive amounts of data (text, images, or numbers) to find patterns. If you show an AI 10,000 pictures of cats, it eventually figures out what "cat-ness" looks like.
- **Reasoning:** Once it recognizes a pattern, it uses logic to reach a conclusion. For example, "This new picture has pointy ears and whiskers, so there is a 95% chance it's a cat."
- **Self-Correction:** AI gets better over time. If it guesses "dog" and you tell it "no, that's a cat," it adjusts its internal logic so it doesn't make that mistake again.

The Hierarchy: One Inside the Other

Artificial Intelligence: The broad concept of machines acting "smart." It includes everything from the simple code in a microwave to a futuristic robot.

Machine Learning: A subset of AI. Instead of being programmed with strict rules, the machine "learns" by looking at thousands of examples and finding patterns.

Deep Learning: A subset of ML that uses **Neural Networks** (layers of math that mimic how human neurons fire). This is the "brain" behind things like self-driving cars and ChatGPT.



Machine Learning?

Machine learning, is a program or system that trains a model from input data.

The trained model can make useful predictions new data (never-seen-before), drawn from the data used to train the model.

Which also means it gives the computer the ability to learn without explicit programming.

ML has 2 main models:

Supervised: Requires human supervision during the training phase to label data and evaluate performance, e.g. spam email, recognizing objects in an image, forecasting stock prices, diagnosing diseases, etc.

UnSupervised: No direct human supervision during training; the algorithm identifies patterns on its own, e.g. Grouping customers by purchasing behavior, identifying fraudulent transactions without prior examples, etc.

LLM (Large Language Model)

LLM is a type of artificial intelligence trained on massive amounts of text data to understand, generate, and manipulate human language.

These models are "large" because they contain billions (and now trillions) of parameters—the internal variables the AI uses to make decisions—and are trained on vast datasets like books, websites, and code repositories.

The AI landscape is divided into **Proprietary (Closed)** models, which are accessible via specific apps or APIs, and **Open-Weight (Open Source)** models, which developers can download and run on their own hardware.

Popular LLMs

Proprietary Leaders (Closed Models)

These models are generally considered the "frontier" of AI performance and are usually accessed through subscriptions. i.e. GPT5.2, Gemini3, Claude5 etc.

Open-Weight Leaders (Open Source)

These models have democratized AI, allowing businesses to keep their data private by hosting the models locally. i.e. Llama4 (Meta), Deepseek-V3, Mistral, Qwen-3 (Alibaba) etc.

How to Choose the Right One?

- Pure reasoning & math: DeepSeek R1 or GPT-5.2 (Reasoning Mode).
- Creative writing: Claude 5 Sonnet or GPT-5.
- Analyzing massive documents: Gemini 3 Pro or Llama 4 Scout.
- Privacy & local hosting: Llama 4 or Mistral.

Tokens and Context Windows

Large language models process your input by breaking it into smaller units called tokens.

A general rule of thumb is that one token corresponds to about four characters of English text, including spaces and punctuation.

This translates to roughly three fourth of a word, so 100 tokens are approximately 75 words.

These tokens count toward what's called the context window of the LLM.

<https://platform.openai.com/tokenizer>

This tool helps you understand how a piece of text is split into tokens, and shows the total token count for that text.

Tokens and Context Windows

Context window is the maximum number of tokens the model can process at once, including both input and output tokens.

The **context window** is also referred to as **window size or token limit**.

For example:

- GPT4 has a context window limit of 128K Tokens, where the maximum output token limit is 16K and the remaining tokens up to 128K are reserved for input or the prompt.
- Google's Gemini Flash model have 1 million tokens Context window, this means you can process enormous amounts of information, such as 11 hours of audio or more than 30,000 lines of code, or over 700,000 words.

Setting up Python Virtual Env.

Creates a folder in current location and activates the virtual env.

```
prompt> python -m venv env_name
```

```
prompt> env_name\scripts\activate
```

```
prompt> env_name\scripts\deactivate
```

List packages installed

```
prompt> pip freeze
```

Add all requirements to the file and install them when needed

```
prompt> pip freeze > requirements.txt
```

```
prompt> pip install -r requirements.txt
```

Using Virtual Env. In VSCode

Open VSCode and [Install Python Extension if not installed](#)

In VSCode view/command palette select followings:

- Python: select interpreter
- Then if you found your virtual environment [created in python] select it.
- Goto virtual environment folder and run activate.ps1

[Or](#)

directly from terminal: path> environment_name\scripts\activate

[If terminal prompt doesn't change do the following:](#)

[In windows powershell run following command:](#)

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope LocalMachine
```

```
Get-ExecutionPolicy -List
```

Installing and Setting Jupyter Notebook

Install Jupyter:

```
prompt> pip install jupyter
```

To run jupyter notebook:

```
prompt> jupyter notebook
```

AI Agents & Agentic AI

Both use AI, but one is a “tool” and the other is a “manager”.

AI Agents & Agentic AI

AI Agent = A smart assistant with one job

This is an AI that can do tasks on its own, but only when you tell it what to do.

Ex:

You say: “AI Agent, write 5 ad copies for Narin 05 Mocha abaya.”

It goes, writes the copies, gives them back. Job done.

It’s like hiring a freelancer. You give clear instructions → they execute → you review → done.

Key traits:

1. *Reactive*: Waits for your command
2. *Single task*: Does what you ask, then stops
3. *Limited autonomy*: Can’t decide “what should I do next?”

Meta’s <http://AdCreative.ai> making ad variants = AI agent.

AI Agents & Agentic AI

Agentic AI = A team of AI agents with a boss

This is a system where multiple AI agents work together + have a “brain” that plans, decides, and keeps going without you micromanaging.

Ex:

You say: “Agentic AI, run Meta campaigns for Narin collection and keep ROAS above 3.” It then:

1. Agent 1: Researches competitors
2. Agent 2: Creates ad creatives
3. Agent 3: Sets up campaigns in Meta
4. Agent 4: Checks performance every 2 hours
5. Agent 5: Pauses bad ads + scales winners
6. Boss AI: Decides “CPA is high, let’s test new audience” and tells agents what to do next

It’s like hiring a full marketing agency + manager. You give the goal → they plan + execute + adapt → you get weekly reports.

Key traits:

1. ***Proactive***: Sets its own sub-goals and next steps
2. ***Multi-step***: Can run for days/weeks on one objective
3. ***High autonomy***: Decides “how” to reach the goal

Tools like AutoGPT, Manus, or advanced Madgicx workflows = Agentic AI.

AI Agents vs Agentic AI

	AI Agent	Agentic AI
Mindset	Tell me what to do	Tell me the goal, I'll figure out how
Analogy	Smart intern	Project manager + team
Example	Make 10 captions for Ad	Launch Narin campaign, optimize for sales, send me ROAS daily
Control	You control every step	You control the goal, AI controls the steps

Generative AI vs Agentic AI

Generative AI is a powerful subfield of artificial intelligence that focuses on creating new and original content i.e.: Audio, Video, Images, Text, Code etc.

Generative AI learns the patterns and structures of massive datasets to produce novel outputs (new and original outputs) that resemble the data it was trained on.

Unlike traditional AI models (known as **Discriminative AI**) that might classify or analyze existing data

Generative AI is a tool that **creates new content** (text, images, code) in reaction to user prompts. **Agentic AI** is a system that **executes tasks and workflows** on your behalf