

# Comprehensive Guide to Generative AI

Estimated time needed: 15 minutes

## Introduction

Generative AI (GenAI) has evolved from basic text and image generation to powering complex systems such as AI agents, enterprise automation, and reasoning engines. This guide explores **core concepts**, **tools**, and **frameworks** for building modern GenAI applications, including **RAG**, **multi-agent systems**, **prompt engineering**, and cutting-edge libraries like LangGraph.

Whether you're developing chatbots, automation workflows, or knowledge systems, this guide provides a roadmap to the latest advancements. It also introduces additional terms not covered in the course videos. These terms are essential for enhancing your understanding of the course concepts.

## Core GenAI Concepts & Terminologies

### Foundational Concepts

| Term                                 | Definition  | Examples/Use Cases  |
|--------------------------------------|---|---|
| LLM                                  | A type of AI model trained on vast amounts of text data to understand and generate human-like language. | GPT-o1, Claude, LLaMA   |
| Prompting                            | A technique for designing input instructions to guide LLM outputs.                                      | "Write a summary in 3 sentences," "Answer as a cybersecurity expert."             |
| Prompt Templates                     | Reusable, structured prompts with placeholders for dynamic inputs.                                      | "Explain {concept} like I'm 5 years old."   |
| RAG (Retrieval-Augmented Generation) | Combines retrieval from external knowledge sources with LLM generation to enhance factual accuracy.     | Answering questions with real-time data (for example, <a href="#">RAG Paper</a> ) |
| Retriever                            | A system component designed to fetch relevant information from a dataset or database.                   | Vector similarity search using FAISS, Elasticsearch                               |
| Agent                                | An autonomous AI system that can plan, reason, and execute tasks using tools.                           | AutoGPT, LangChain Agents   |
| Multi-Agent System                   | A framework in which multiple AI agents collaborate to solve complex tasks.                             | Microsoft AutoGen, CrewAI   |
| Chain-of-Thought                     | A prompting technique that encourages models to decompose problems into intermediate steps.             | "Let's think step by step..."   |
| Hallucination Mitigation             | Strategies to reduce incorrect or fabricated outputs from LLMs.   | RAG, fine-tuning, prompt constraints  |
| Vector Database                      | A database optimized for storing and querying vector embeddings.  | Pinecone, Chroma, Weaviate  |
| Orchestration                        | Tools to manage and coordinate workflows involving multiple AI components.                              | LangChain, LlamaIndex   |
| Fine-tuning                          | Adapting pre-trained models for specific tasks using domain-specific data.                              | LoRA (Low-Rank Adaptation), QLoRA (quantized fine-tuning)                         |

## Tools & Frameworks

### Model Development & Deployment

| Tool/Framework | Definition  | Examples/Use Cases                           | Reference Link               |
|----------------|---|--|------------------------------|
| Hugging Face   | A platform hosting pre-trained models and datasets for NLP tasks.         | Accessing GPT-2, BERT, Stable Diffusion      | <a href="#">Hugging Face</a> |
| LangChain      | A framework for building applications with LLMs, agents, and tools.       | Creating chatbots with memory and web search | <a href="#">LangChain</a>    |
| AutoGen        | A library for creating multi-agent conversational systems.                | Simulating debates between AI agents         | <a href="#">AutoGen</a>      |
| CrewAI         | A framework for assembling collaborative AI agents with role-based tasks. | Task automation with specialized agents      | <a href="#">CrewAI</a>       |
| BeeAI          | A lightweight framework to build production-ready multi-agent systems     | Distributed problem-solving systems          | <a href="#">BeeAI</a>        |
| LlamaIndex     | A tool to connect LLMs to structured or unstructured data sources.        | Building Q&A systems over private documents  | <a href="#">LlamaIndex</a>   |
| LangGraph      | A library for building stateful, multi-actor applications with LLMs.      | Cyclic workflows, agent simulations          | <a href="#">LangGraph</a>    |

### Retrieval & Infrastructure

| Tool/Framework | Definition  | Examples/Use Cases                         | Reference Link           |
|----------------|---|--|--------------------------|
| FAISS          | A library for efficient similarity search of dense vectors. | Retrieving top-k documents for RAG         | <a href="#">FAISS</a>    |
| Pinecone       | A managed cloud service for vector database operations.     | Storing embeddings for real-time retrieval | <a href="#">Pinecone</a> |
| Haystack       | An end-to-end framework for building RAG pipelines.         | Deploying enterprise search systems        | <a href="#">Haystack</a> |

Advanced Prompting Techniques

| Concept             | Definition   | Example  |
|---------------------|--|--|
| Few-Shot Prompting  | Providing examples in the prompt to guide the model's output format. | "Translate to French: 'Hello' → 'Bonjour'; 'Goodbye' → __"             |
| Zero-Shot Prompting | Directly asking the model to perform a task without examples.        | "Classify this tweet as positive, neutral, or negative: {tweet}"       |
| Chain-of-Thought    | Encouraging step-by-step reasoning.                                  | "First, calculate X. Then, compare it to Y. Final answer: __"          |
| Prompt Chaining     | Breaking complex tasks into smaller prompts executed sequentially.   | Prompt 1: Extract keywords → Prompt 2: Generate summary from keywords. |

Key Architectures & Workflows

RAG Pipeline

- 1. **Retrieval:** Query vector database (for example, Pinecone) for context.
- 2. **Augmentation:** Combine context with user prompt.
- 3. **Generation:** LLM (for example, GPT-4) produces final output.

Multi-Agent System

- **Agents:** Specialized roles (for example, researcher, writer, critic).
- **Orchestration:** LangGraph for cyclic workflows, AutoGen for conversations, and so on.
- **Tools:** Web search, code execution, API integrations, and so on.

References

- 1. [Retrieval-Augmented Generation \(RAG\) Paper](#)
- 2. [Chain-of-Thought Prompting](#)
- 3. [LangGraph Documentation](#)
- 4. [CrewAI Documentation](#)
- 5. [Prompt Engineering Guide](#)

Author

[Hailey Quach](#)



Skills Network