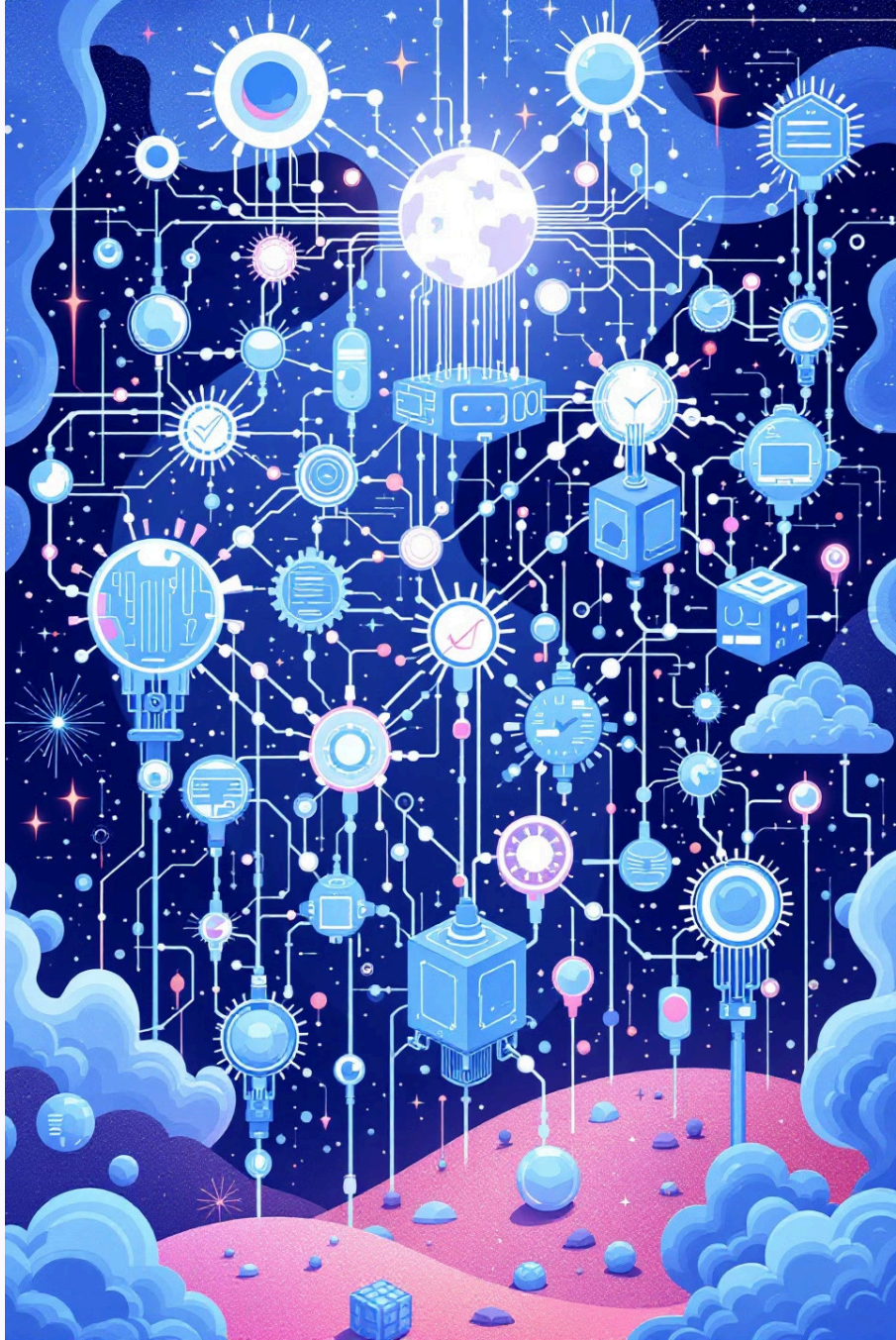
What is Artificial Intelligence (AI)?

Definition: AI is the branch of computer science that creates systems capable of performing tasks requiring human intelligence, such as understanding language, recognizing patterns, decision-making, and problem-solving.

What are Large Language Models (LLMs)?

Definition

LLMs are advanced AI models trained on vast amounts of text data to understand, generate, and interact with natural language using deep learning techniques, particularly transformers.



How LLMs Work

1

Pre-Training

Trained on diverse datasets (books, articles, code, etc.) to predict the next word in a sentence.

Example: Given "The capital of Italy is," the model predicts "Rome."

2

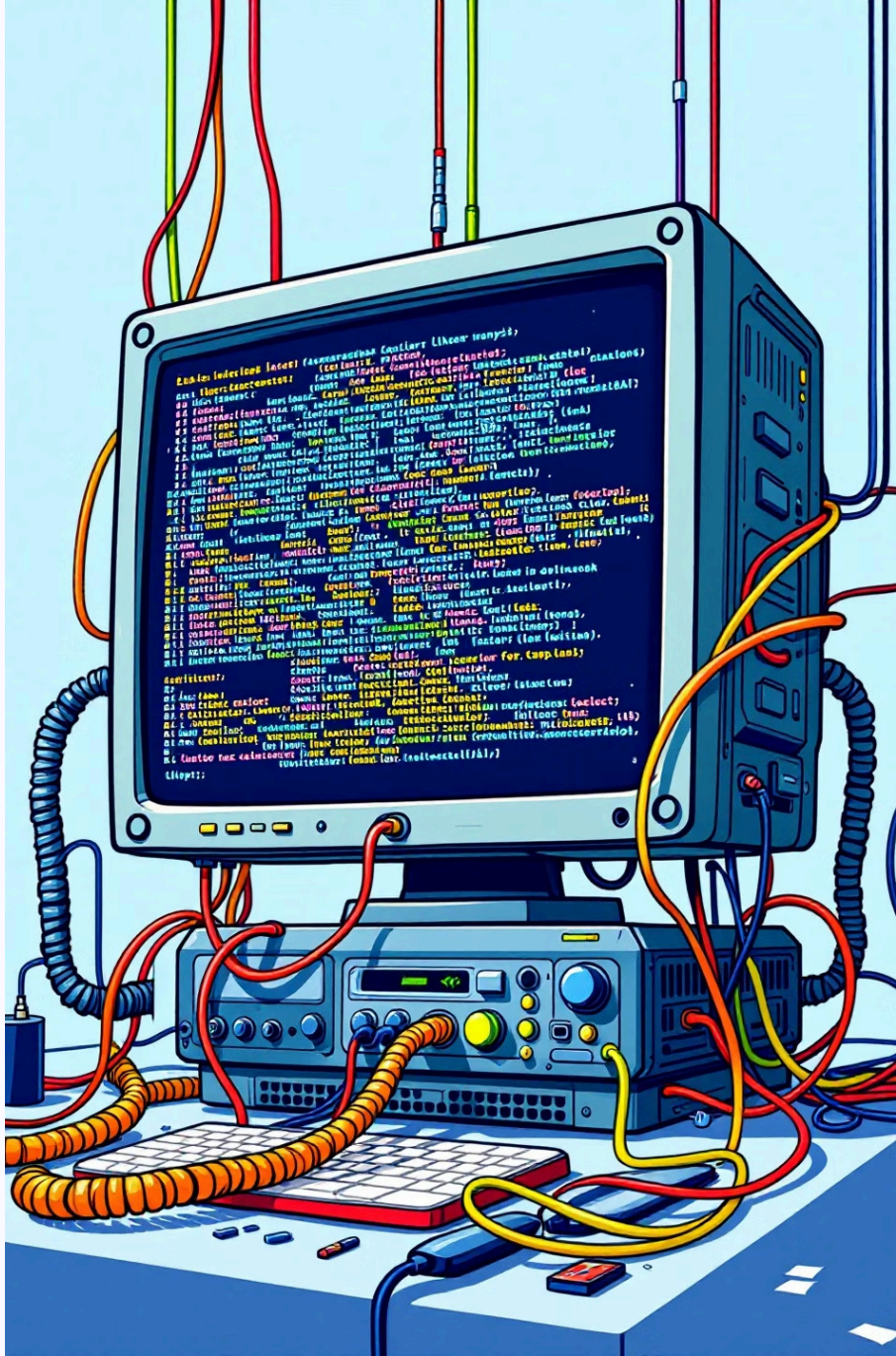
Fine-Tuning

Refined for specific tasks like customer support or medical diagnosis.

3

Inference

Generates responses to user queries using pre-trained knowledge.



Top 5 Large Language Models: One-Liner Summaries

1 **GPT-4 (OpenAI)**

Versatile model excelling in multi-tasking, reasoning, and conversational AI.

2 **Gemini (Google)**

Next-gen AI with advanced reasoning and multi-modal capabilities.

3 **Claude (Anthropic)**

Safety-focused conversational model optimized for ethical AI.

4 **GitHub Copilot (OpenAI Codex)**

AI assistant for coding, aiding in writing and debugging.

5 **Perplexity AI**

Combines retrieval-augmented generation with conversational AI for fact-based responses.

LLM Challenges :

1. Outdated Knowledge: Pre-trained LLMs may lack recent or domain-specific information (e.g., company policies, FAQs).
2. Data Privacy: LLMs cannot inherently verify or access private/secure documents.
3. Hallucination: LLMs can generate fabricated or inaccurate information.

RAG Process: End-to-End

1

User Input

The user submits a query (e.g., "How do solar panels work?").

2

Query Embedding

Converts the query into a vector representation. Allows comparison with document vectors in a vector database (e.g., FAISS, Pinecone).

3

Document Retrieval

Searches the vector database for the closest matching documents. Example: Retrieves snippets like "Solar panels convert sunlight into electricity using photovoltaic cells."

4

Input Combination

Combines the user query and retrieved documents into a single prompt for the LLM.

5

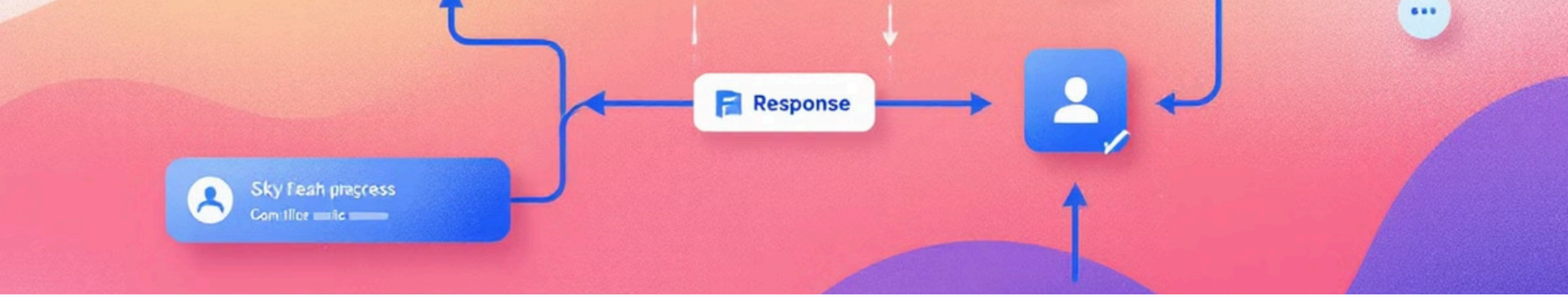
Response Generation

The LLM processes the prompt and generates a natural language response. Example: "Solar panels work by converting sunlight into electricity using photovoltaic cells."

6

Final Response

Returns the response, grounded in the retrieved knowledge.



Flow of Data in RAG

1

Query Embedding

User query → Dense vector.

2

Document Retrieval

Query vector → Relevant document snippets.

3

Input to LLM

Query + Retrieved documents → Structured prompt.

4

Response Generation

Prompt → Natural language response.

Key Advantages of RAG

General Advantages

1. **Domain-Specific Knowledge**: Accesses custom knowledge bases (e.g., company policies, FAQs).
2. **Dynamic Updates:** Responds with real-time, up-to-date data.
3. **Reduced Hallucination**: Ensures reliability by grounding responses in retrieved content.

RAG in Product-Based Companies

- **Cost Efficiency**: Reduces the need to train massive LLMs from scratch.
- **Privacy**: Ensures sensitive data stays within secure, localized knowledge bases.
- **Enhanced Interactions:** Powers LLM chatbots for accurate, context-aware responses.

Why not just code to access docs directly instead of LLM?

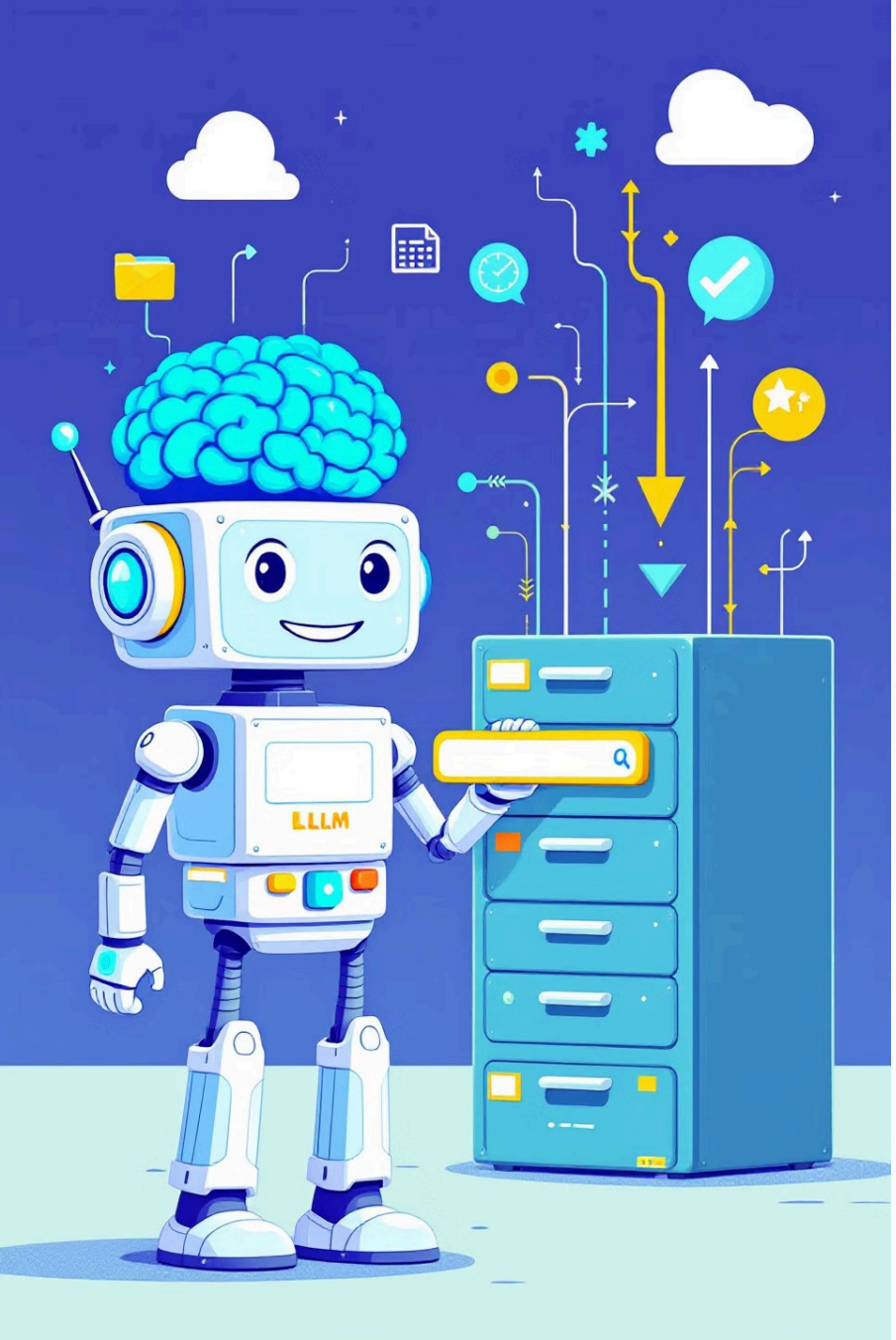
Challenges

1. **Understanding Natural Language:** Human queries are complex and nuanced.
2. **Locating Precise Information:** Long documents make pinpointing answers difficult.
3. **Synthesizing Information**: Combining and summarizing content from multiple sources is challenging.

Conclusion

LLMs act as a researcher, turning raw document data into clear, human-readable answers.

In short, while you have the "ground truth" in your documents, you need the LLM to act as the bridge between that raw information and a meaningful, human-understandable answer.



Why Combine LLM with Retrieval?

- **Without RAG:** LLMs might generate hallucinated or incorrect information.
- **With RAG:** The LLM is constrained to use only the retrieved, accurate documents, improving reliability.

1. **Dynamic Information Retrieval** •
   Unlike static hardcoded document access, RAG dynamically fetches updated or domain-specific data in real-time.
   - Example: Corporate policies or FAQs that change frequently.

Should I Use LLM Training Data if Information is Missing?

Option 1: Allow LLM to Use Pre-Trained Data

Pros:

1. Broader knowledge base.
2. Maintains user satisfaction.
3. Acts as a fallback when retrieval fails.

Cons:

1. Risk of hallucination.
2. Potential loss of trust.

Option 2: Do Not Use Pre-Trained Data

Pros:

1. Ensures transparency and control.
2. Encourages knowledge base improvement.

Cons:

1. Can frustrate users with "no answer found."
2. Misses opportunities for general knowledge responses.

Practical Recommendations

1

When to Opt for LLM Check •

For general-purpose applications where user satisfaction and broad coverage are priorities. • Example: A public chatbot for general queries.

When Not to Opt for LLM Check •

For domain-specific applications requiring strict accuracy and reliability. • Example: A medical assistant system retrieving verified healthcare guidelines.

Hybrid Approach To balance these options,

Consider a hybrid solution:

1. Primary Source: Attempt retrieval from the document database. 2. Fallback: o Use the LLM for queries flagged as low-risk (general knowledge). o Clearly tag responses as “based on pre-trained knowledge” when using the LLM.