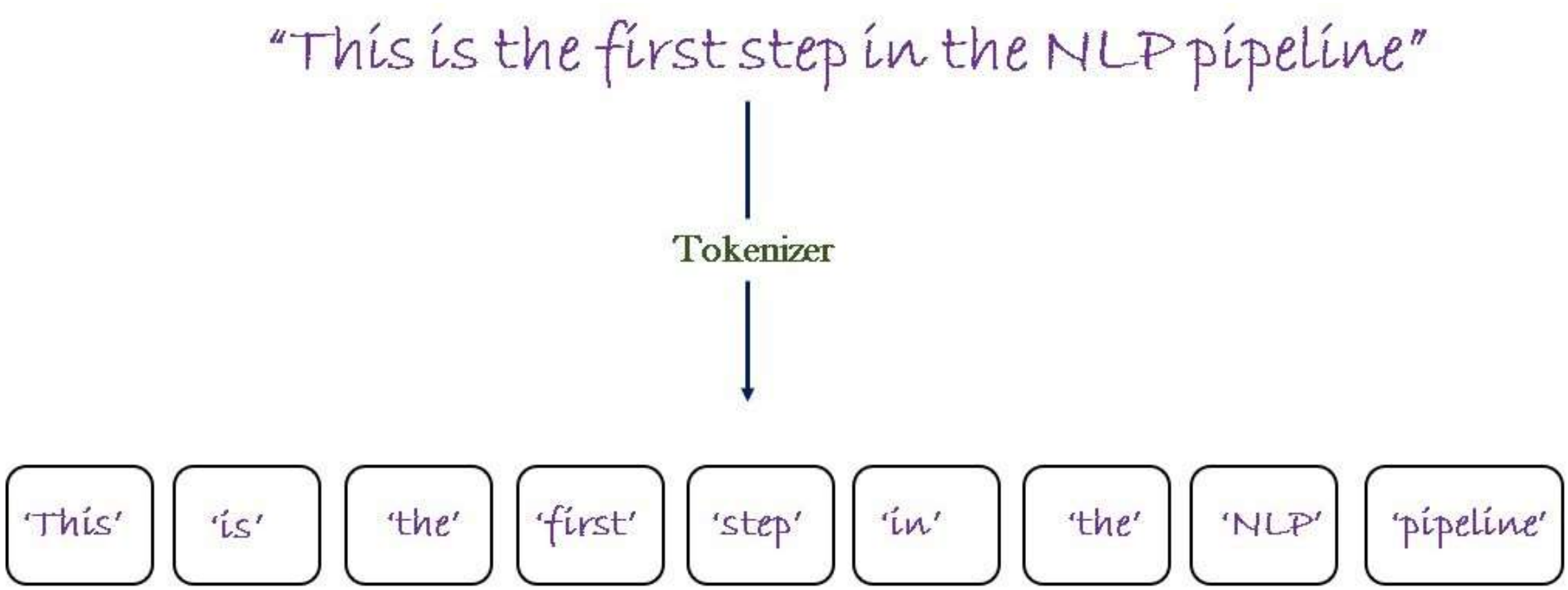
LLM Dictionary

1

1. Tokenization

Breaking text into smaller chunks (tokens).

- Splits words into pieces for processing.
- Essential for LLM input handling.
- Example: "Hello, AI!" → [Hello] [,] [AI] [!]

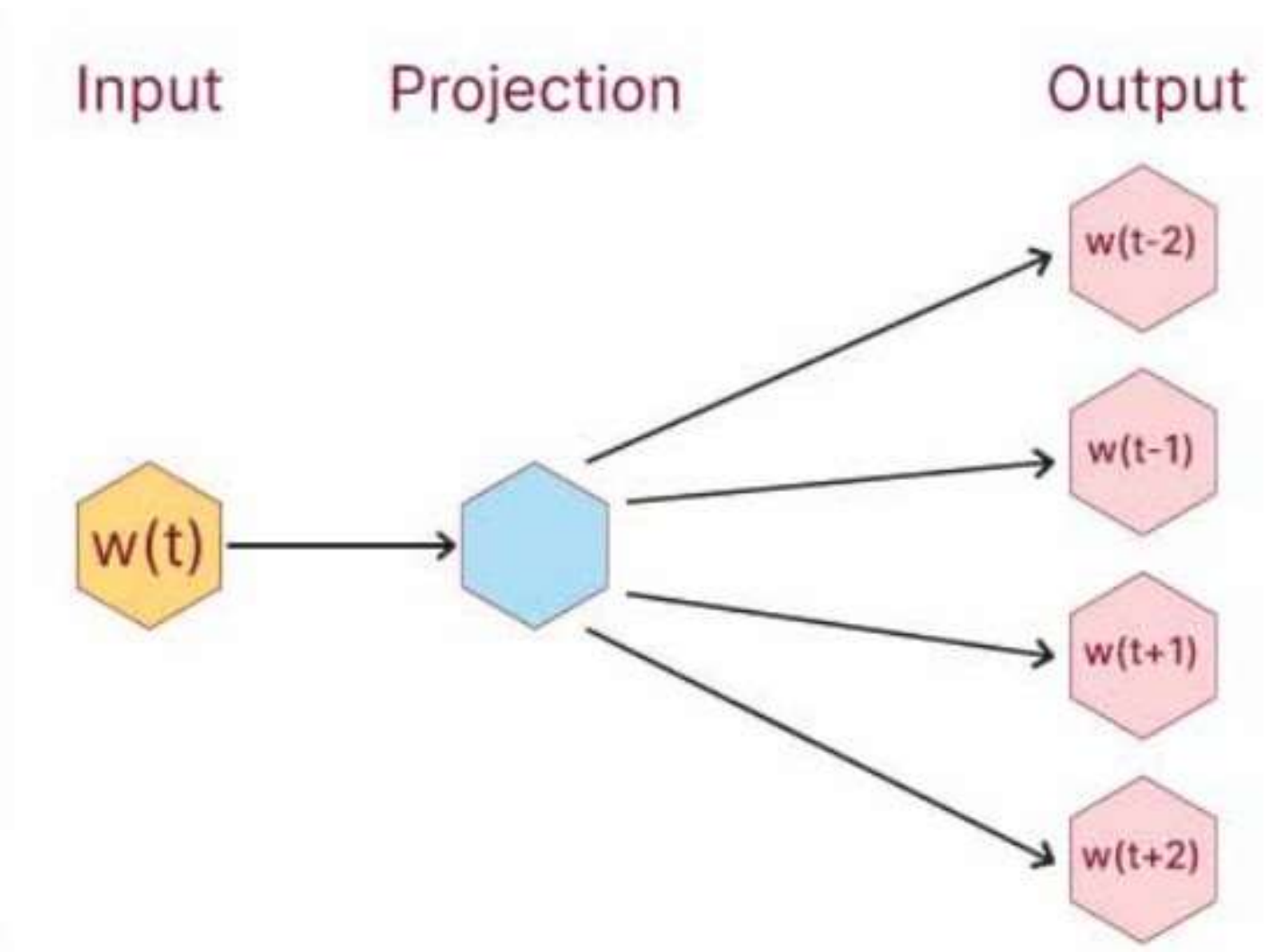


2

2. Embeddings

Converting words into numerical representations.

- Transforms text into vectors.
- Helps models understand meaning.
- Used in search, RAG, and NLP.

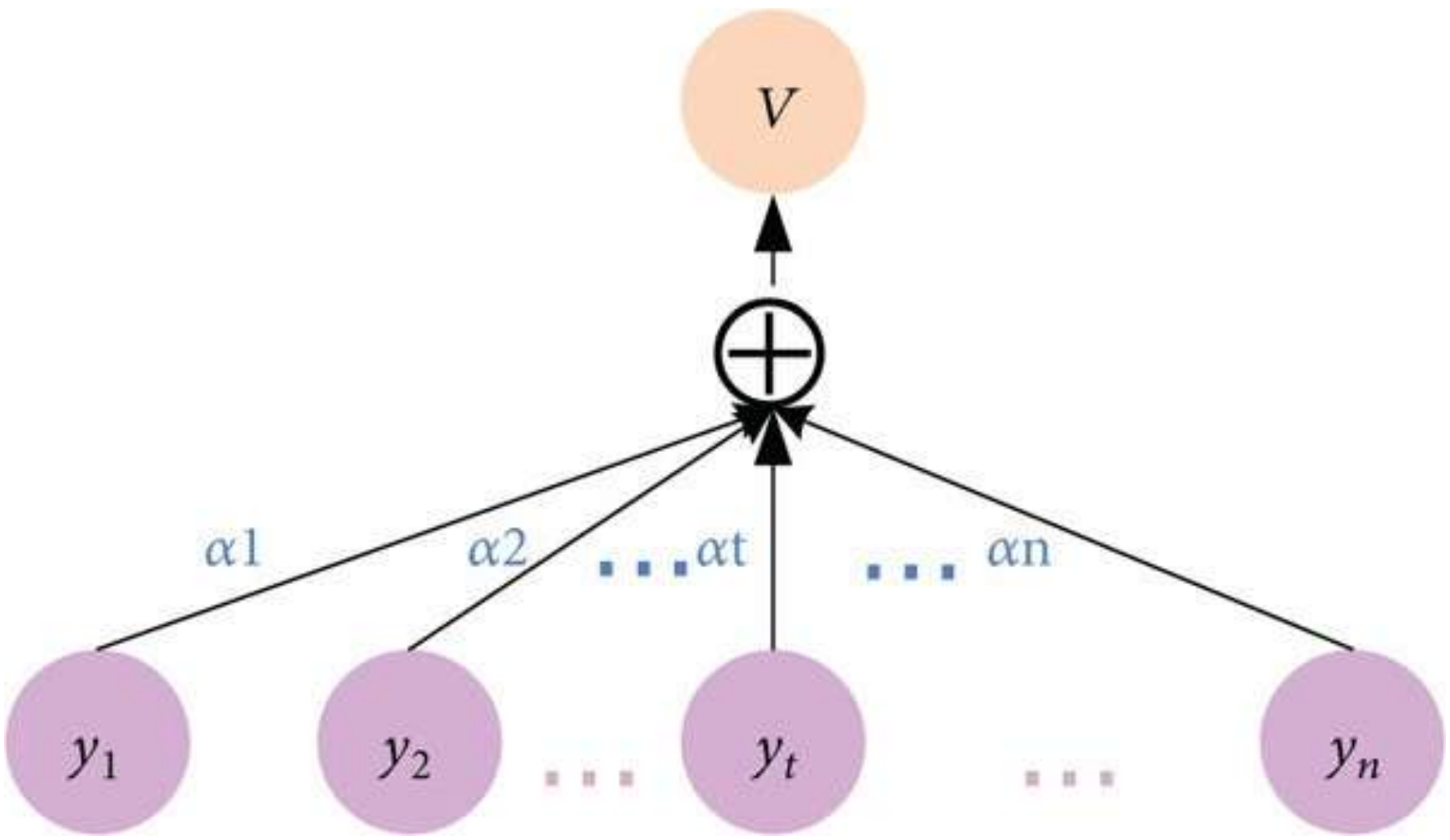


3

3. Attention Mechanism

How LLMs focus on important parts of input.

- Key feature of Transformers.
- Determines word relevance.
- Example: Self-Attention in GPT.

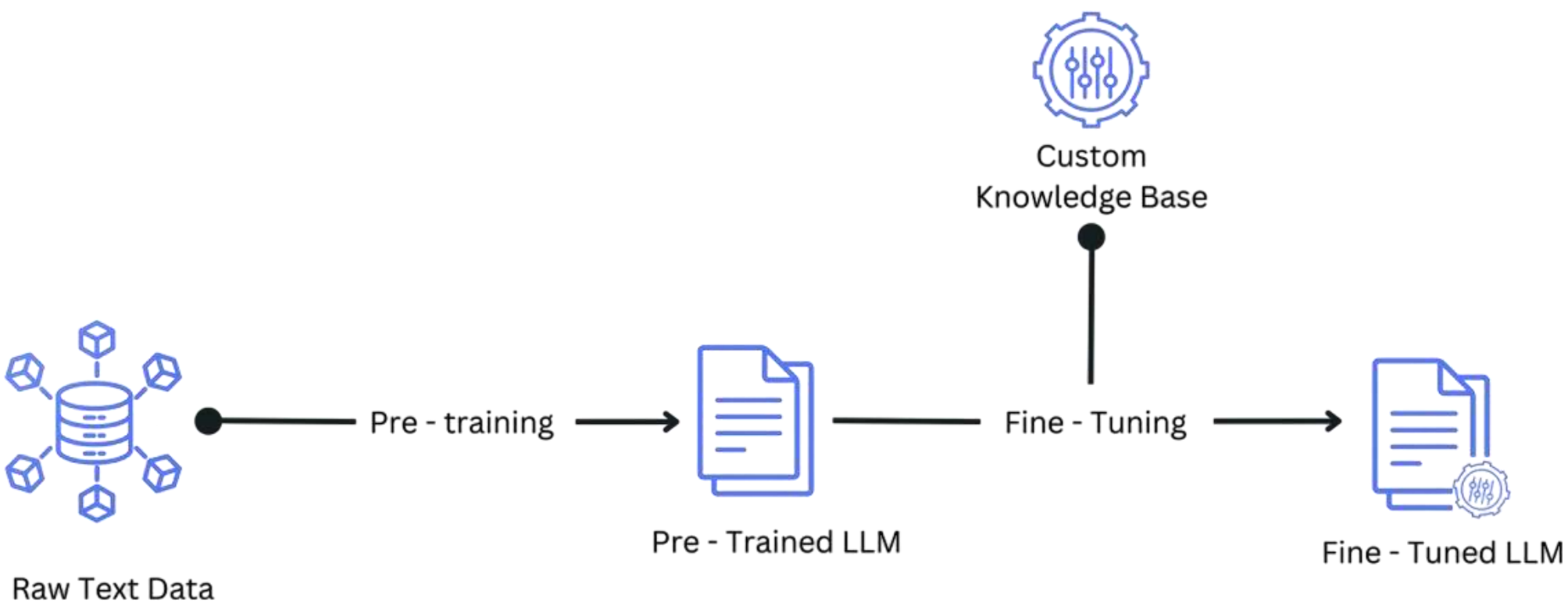


4

4. Fine-Tuning

Adapting a pre-trained model for specific tasks.

- Customizes LLMs for industries.
- Uses labeled datasets.
- Improves accuracy in niche areas.

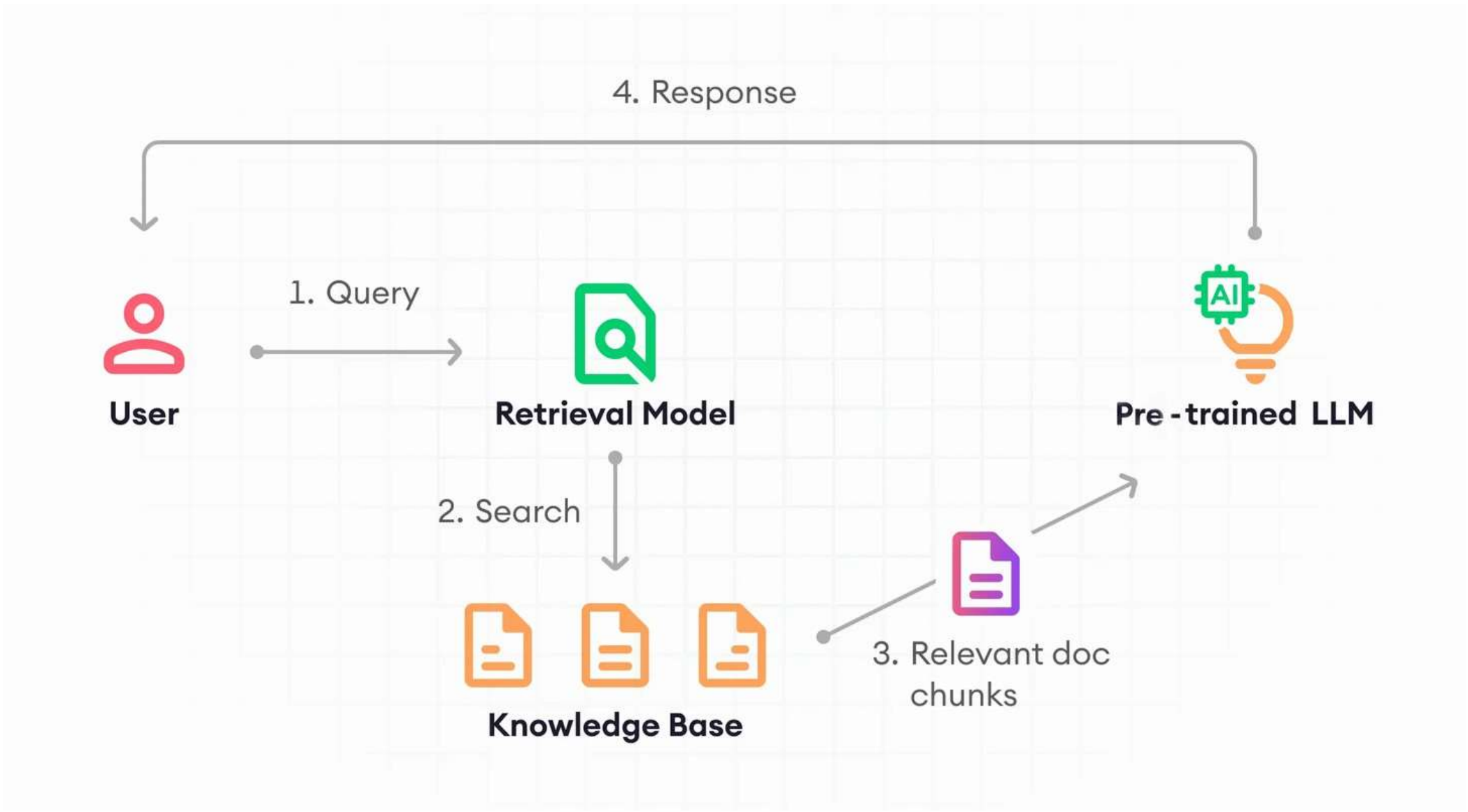


5

5. RAG (Retrieval-Augmented Generation)

Enhancing LLMs with external knowledge.

- Combines retrieval + generation.
- Accesses real-time or proprietary data.
- Used in chatbots & enterprise AI.

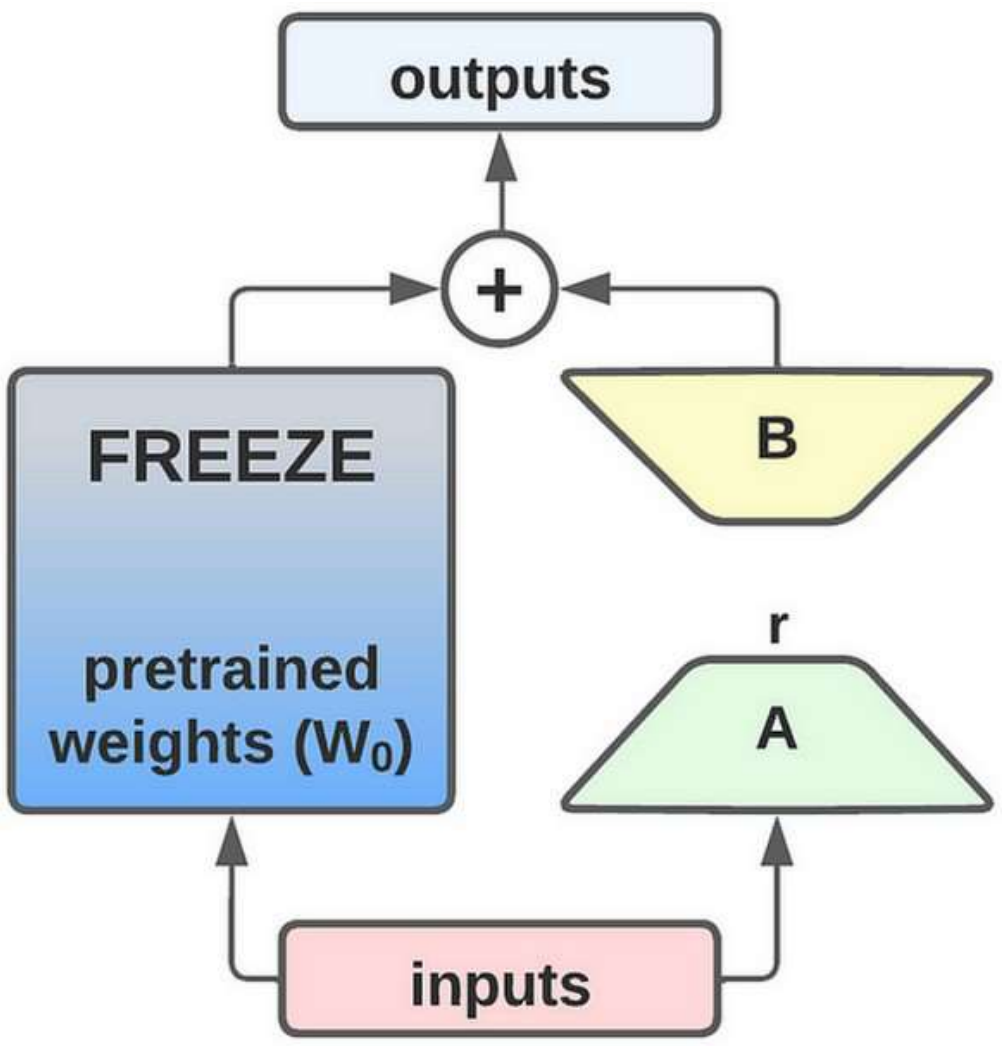


6

6. LoRA (Low-Rank Adaptation)

Efficient fine-tuning method for LLMs.

- Reduces computational cost.
- Adds small adapters to models.
- Used in lightweight model customization.

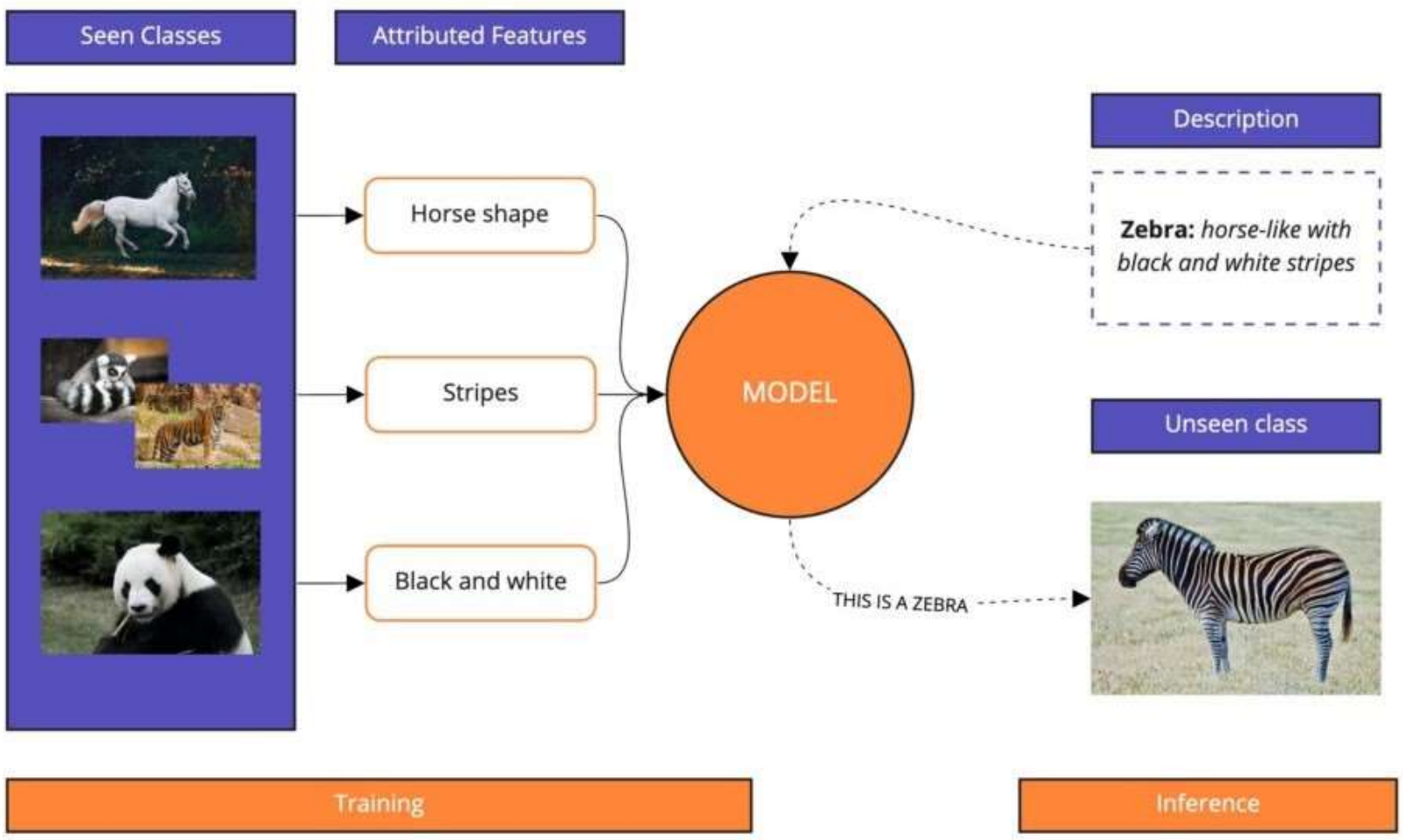


7

7. Zero-Shot & Few-Shot Learning

How AI learns with minimal examples.

- Zero-shot: No prior examples.
- Few-shot: Learns from few samples.
- Used in LLM inference & NLP tasks.

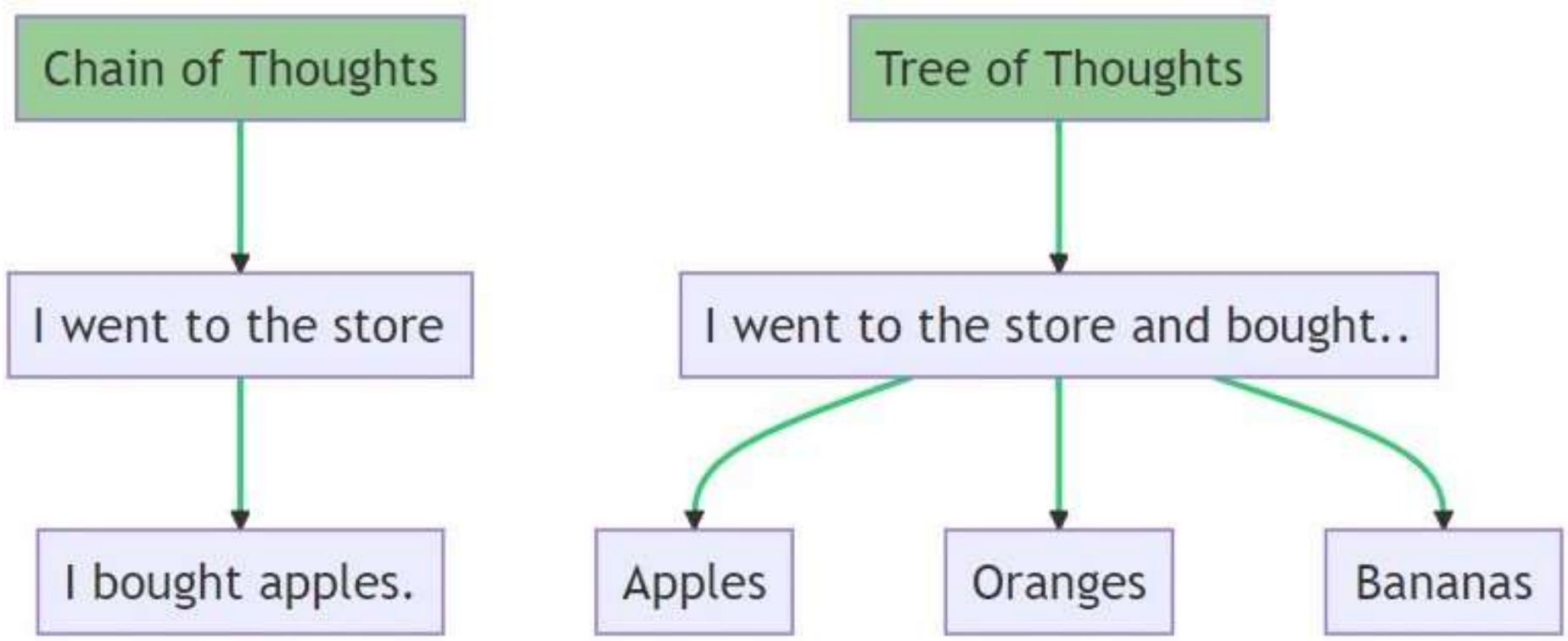


8

8. Chain-of-Thought (CoT) Prompting

Guiding AI through step-by-step reasoning.

- Helps LLMs think logically.
- Improves accuracy in complex tasks.
- Used in math, coding, and problem-solving.

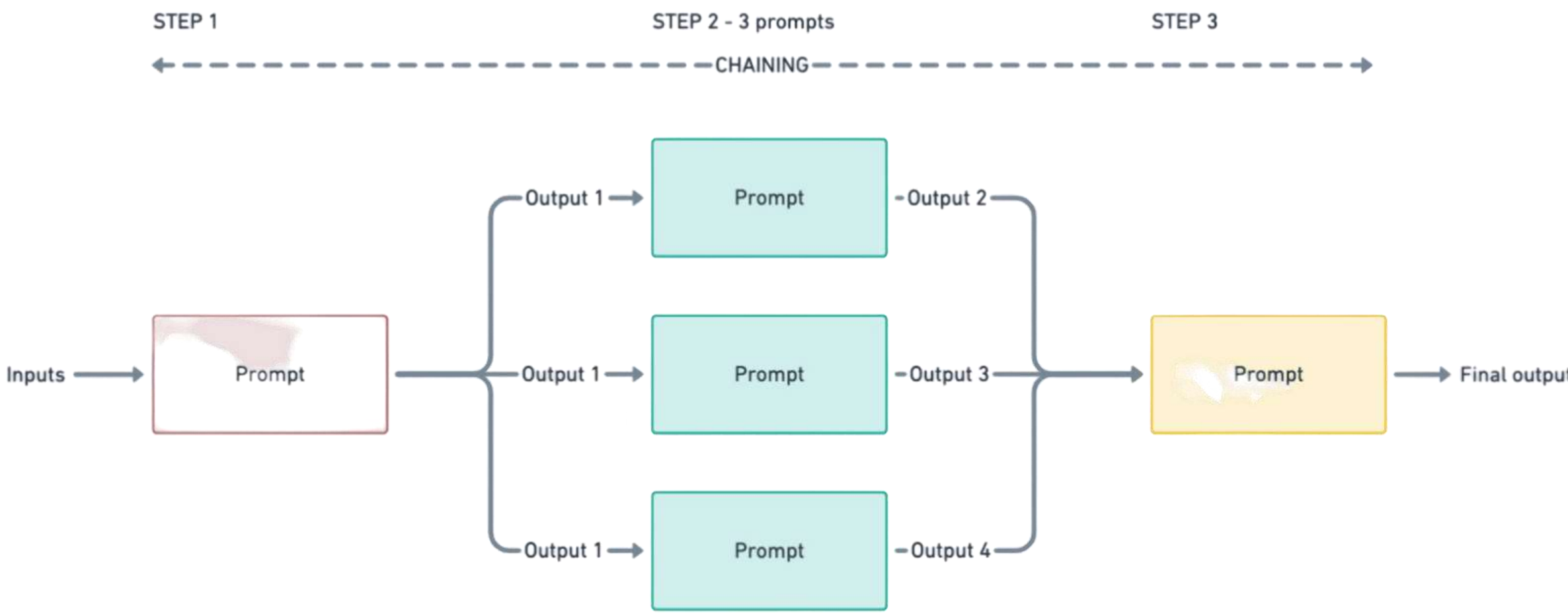


9

9. Prompt Engineering

Crafting inputs to optimize AI responses.

- Directly impacts LLM output.
- Uses context, examples, and formatting.
- Essential for developers & AI engineers.



10

10. Model Inference

Generating responses in real time.

- Runs LLMs on trained data.
- Used in chatbots, search, AI apps.
- Requires GPU/TPU acceleration.