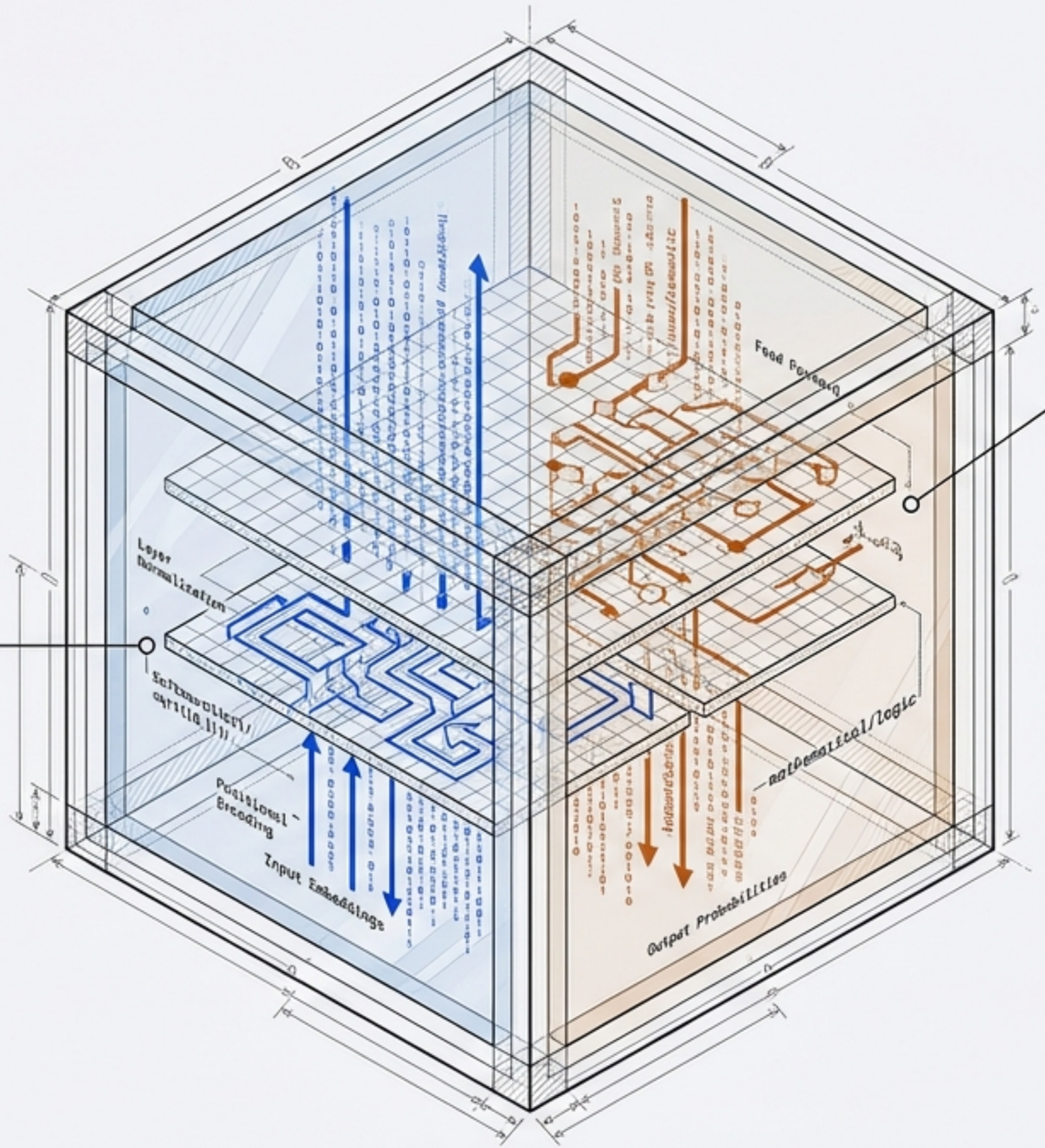
Inside the Black Box.

The Mechanics of Large Language Models: From Vector Space to Generative AI.



Feed Forward

$$\mathrm{ReLU}(xW_1+b_1)W_2+b_2$$

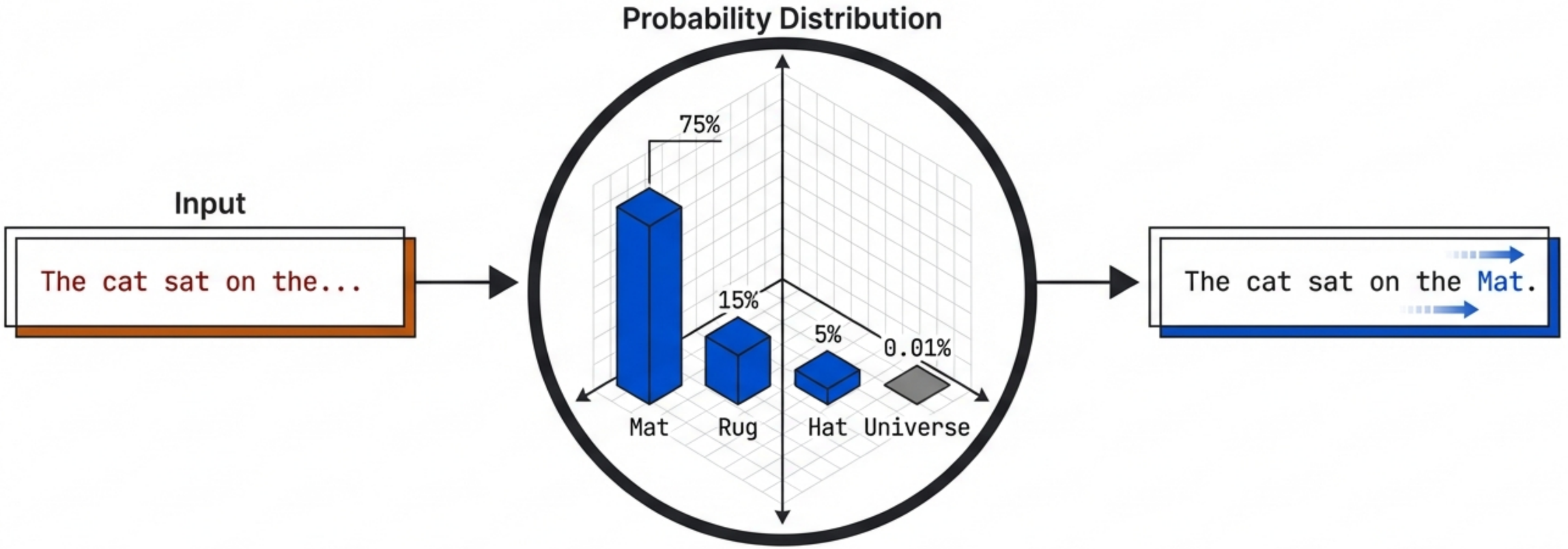
Attention

$$\mathrm{Softmax}(QK^T/\sqrt{d_k})V$$

Synthesis of insights from Piyush Garg, 3Blue1Brown, IBM, and Elastic.

It’s Just Math, Not Magic.

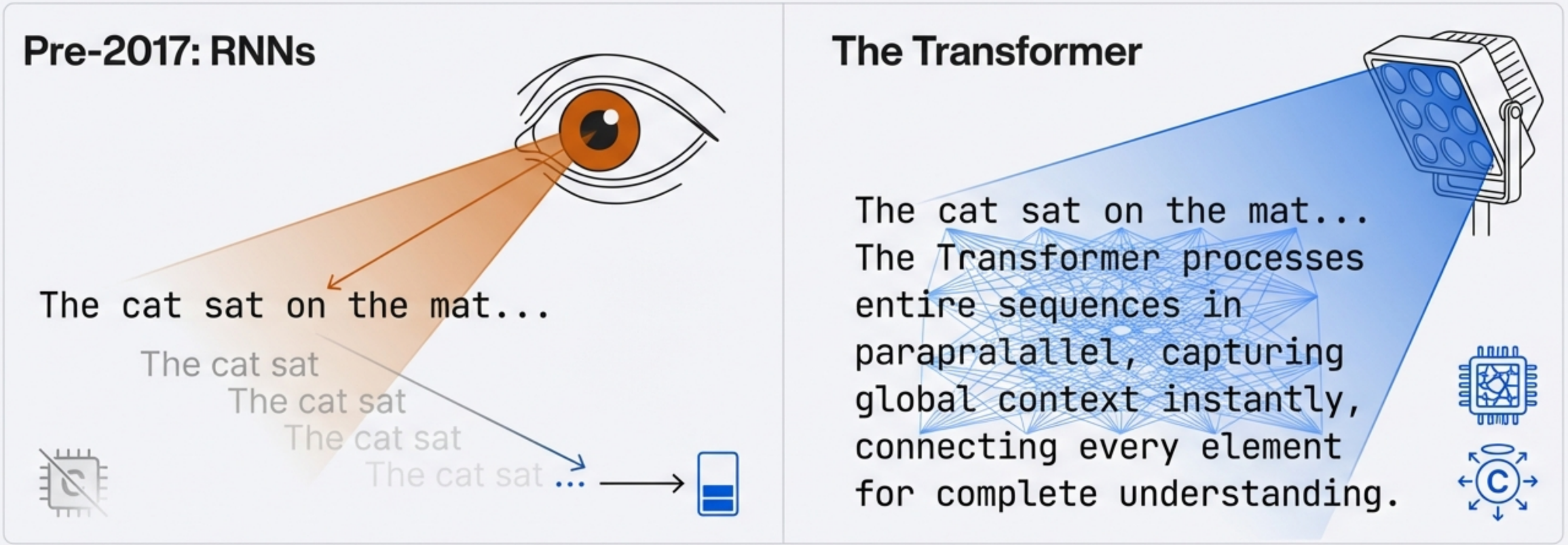
The Next-Word Prediction Engine.



An LLM is a deterministic statistical function. It does not ‘know’ facts; it calculates the probability of the next token based on training data distribution.

Attention Is All You Need

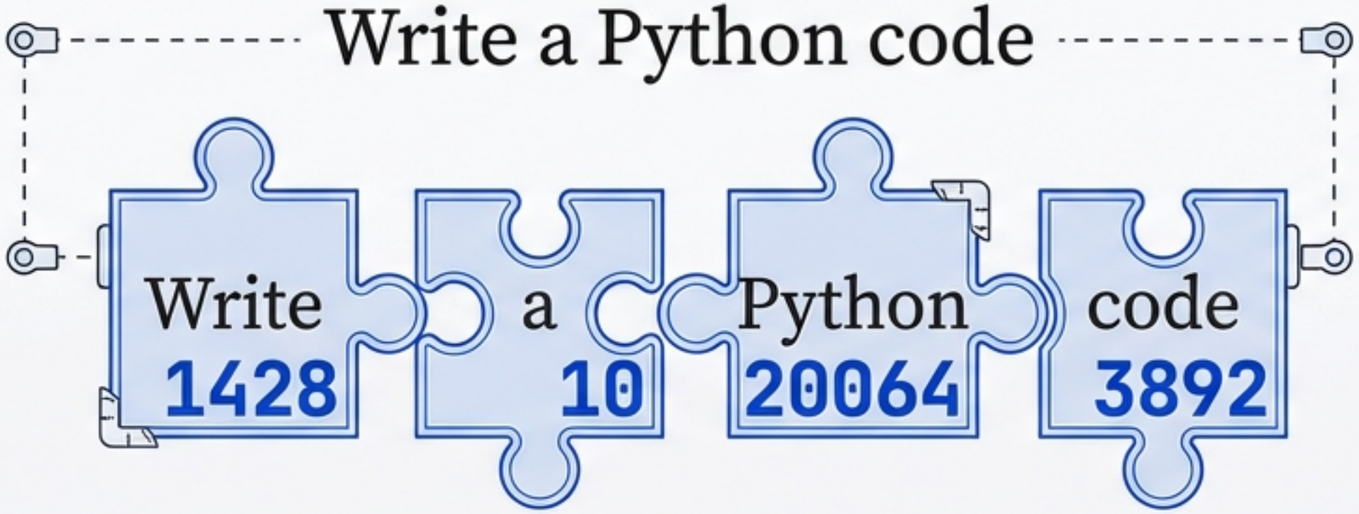
The shift from Sequential to Parallel Processing (2017).



Before Transformers, models read sequentially, losing context over long distances. Transformers process entire sequences in parallel, capturing global context instantly.

Step 1: Tokenization

Computers Speak Integers, Not English.

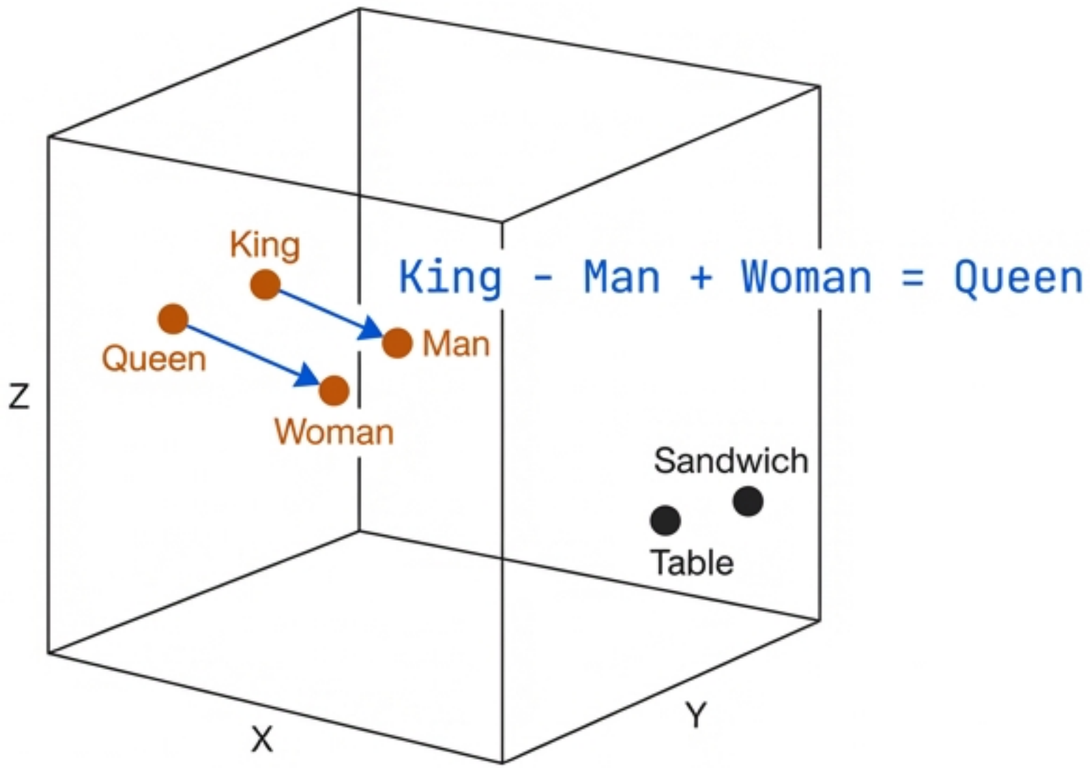



```
tokenizer = AutoTokenizer.from_pretrained('gpt-4')
tokens = tokenizer.encode('Hey there')
# Output: [213, 1823]
```

Text is broken into chunks called 'Tokens'. Every unique token is mapped to a specific ID number in a fixed vocabulary.

Step 2: Embeddings

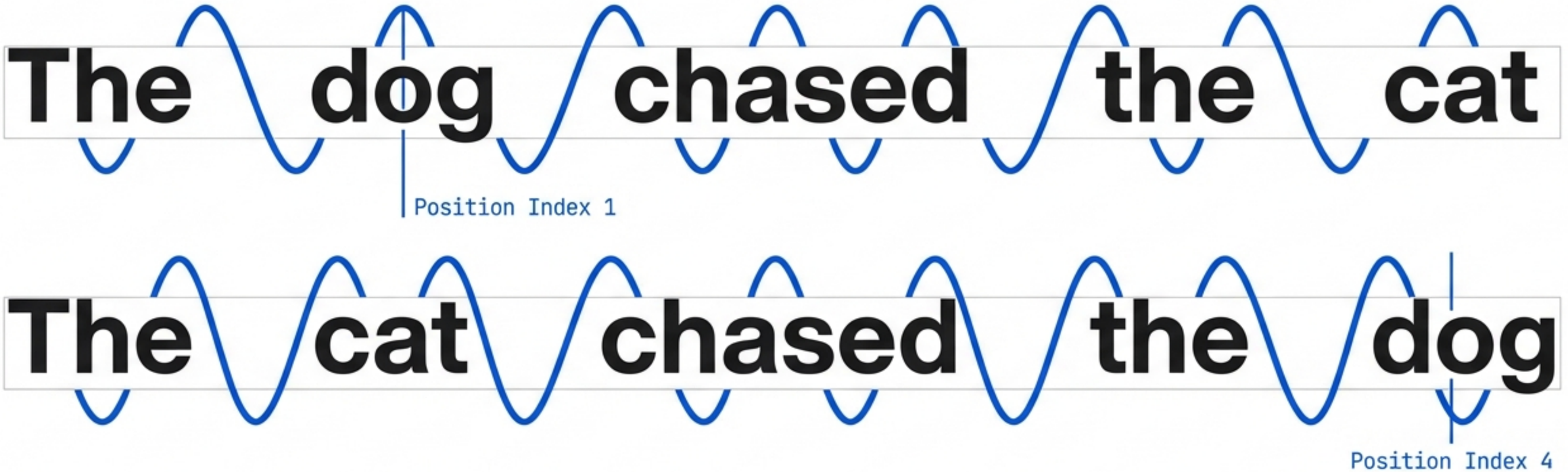
Converting Numbers to Meaning in Vector Space.



Tokens are converted into Vectors—lists of numbers representing coordinates. Words with similar meanings appear physically closer in this multi-dimensional mathematical space.

Step 3: Positional Encoding

Injecting Order into Chaos.

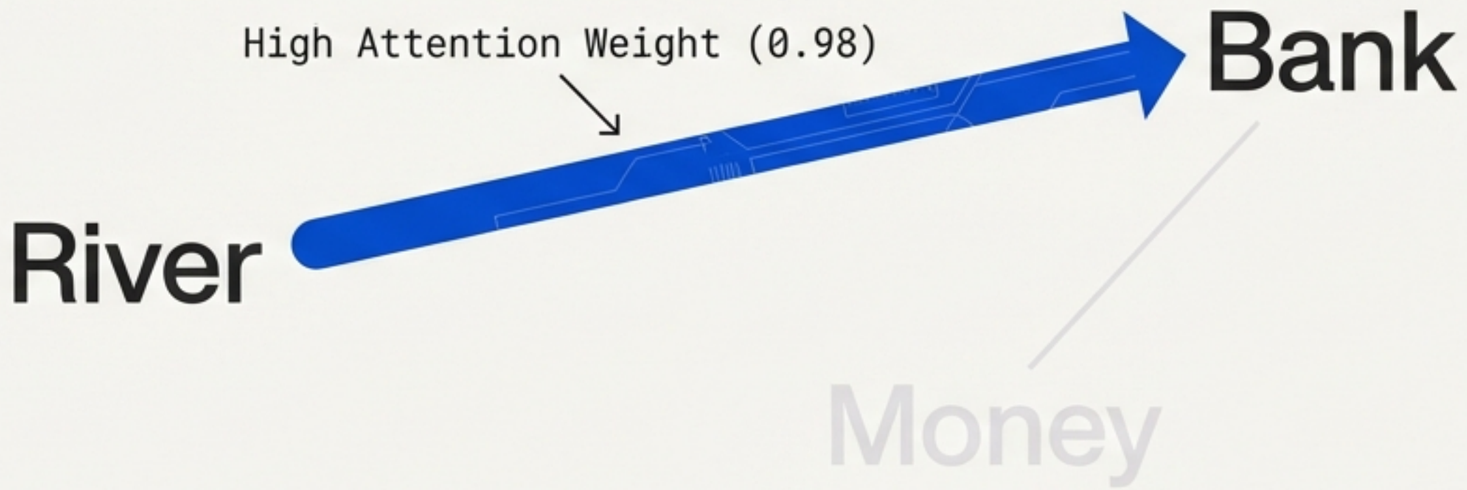


Because Transformers read all words at once, they lack an inherent sense of order. We add a unique "time stamp" signal (Positional Encoding) to the math of each word to define the sequence.

Step 4: The Attention Mechanism

How Words Talk to Each Other to Resolve Ambiguity

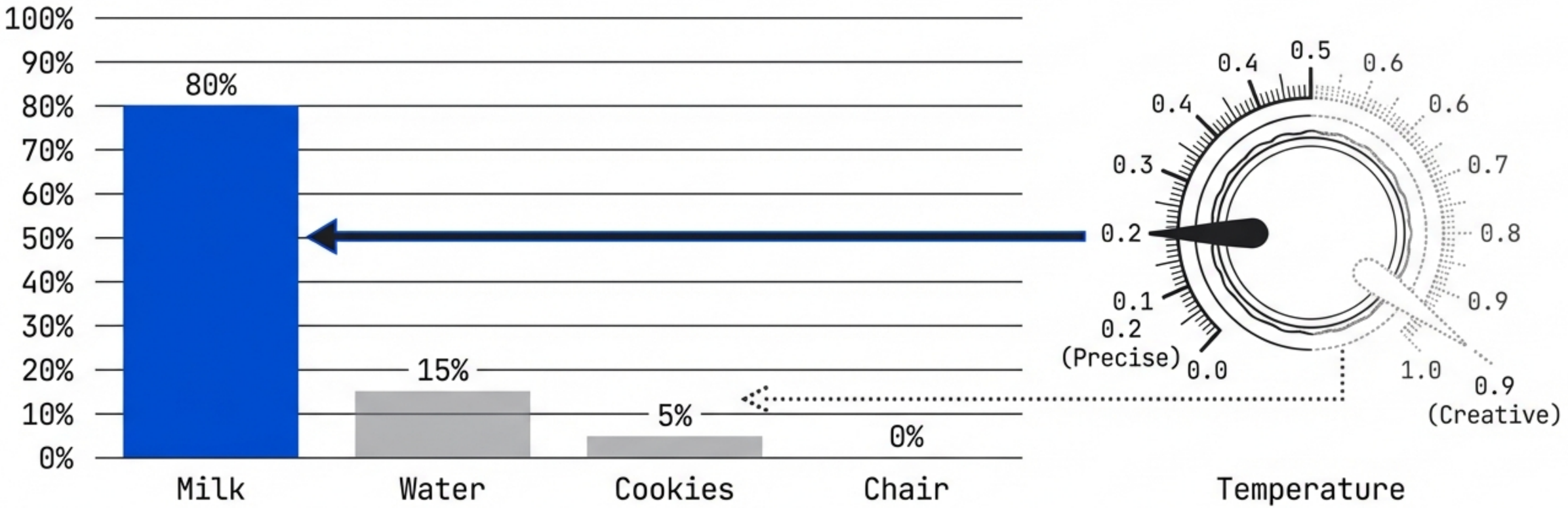
I went down to the river bank.



Self-attention allows every token to weigh its relevance against every other token. Here, the model updates the meaning of ‘Bank’ to be geological rather than financial, based purely on the presence of the word ‘River’.

Step 5: The Output & Softmax

Converting Math Back to Words



The final layer produces a list of probabilities for all possible words. The ‘Temperature’ setting determines whether the model plays it safe (High Probability) or gets creative (Low Probability).

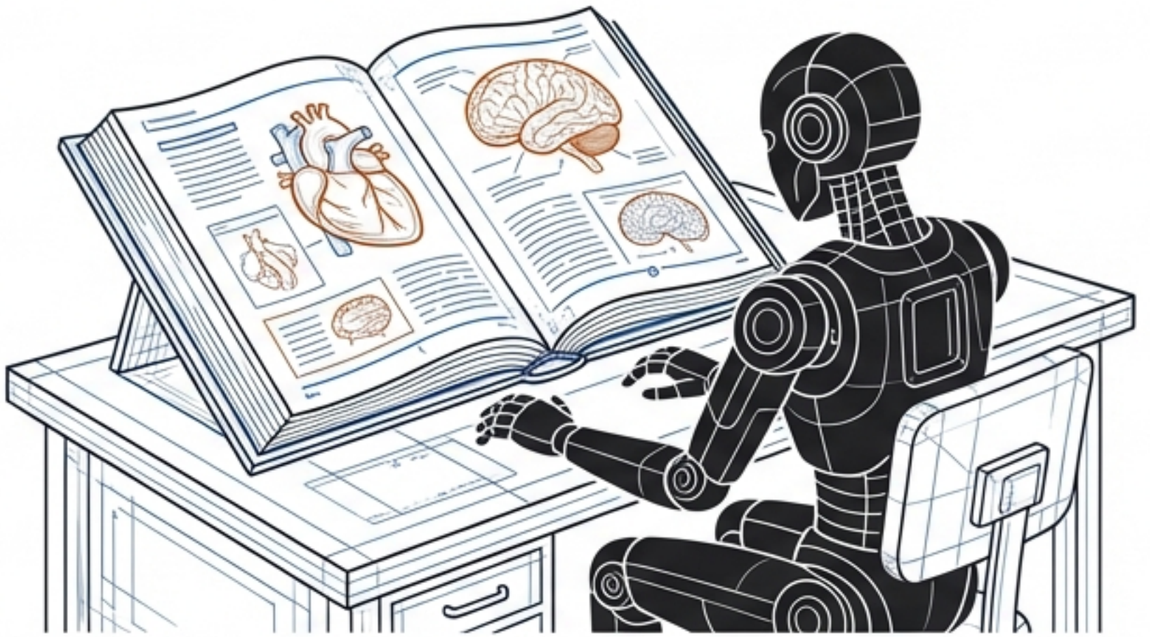
Training the Model

Pre-Training vs. Fine-Tuning.

Pre-Training: The Generalist

Unsupervised Learning on Trillions of Tokens
(Wikipedia, GitHub, Common Crawl).

Fine-Tuning: The Specialist

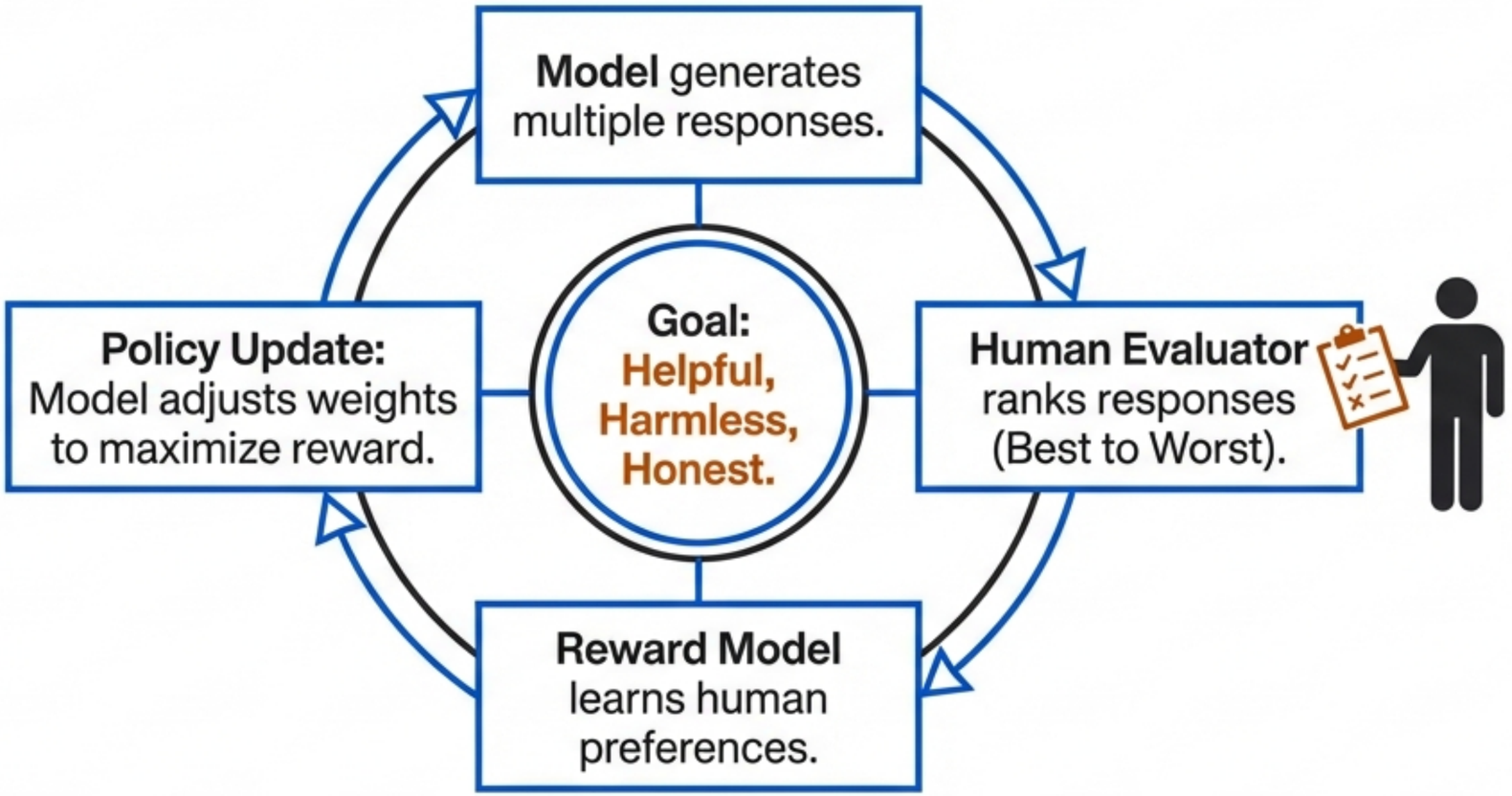


Supervised Learning on Labeled Datasets
(Medical, Legal, Coding).

Pre-training creates the base model by reading the internet (learning grammar and facts).
Fine-tuning specializes the model for specific tasks using curated, high-quality data.

RLHF: Alignment

Reinforcement Learning from **Human Feedback.**



Raw models can be toxic or unhelpful. RLHF aligns the mathematical output with human values by rewarding the model when it behaves correctly.

Applications & Future Use

Leveraging LLMs for Business and Knowledge.

Beyond the Chatbot

Core Business Applications of LLMs.

Generative

Content creation, marketing copy, email drafting.

Coding

Writing functions, debugging, legacy translation (COBOL to Python).

Analytical

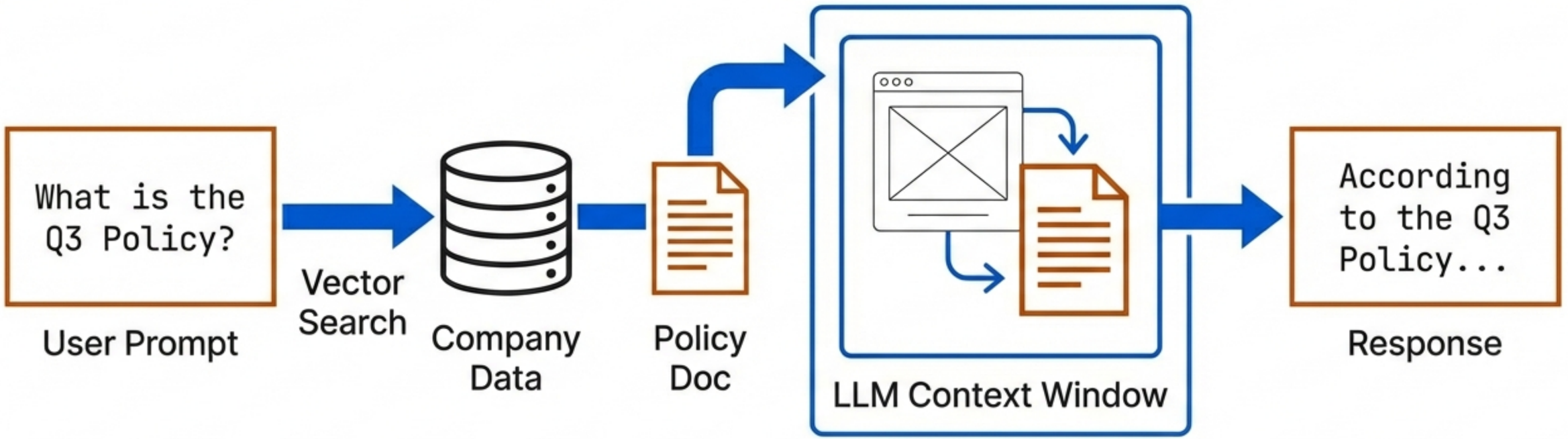
Sentiment analysis, summarization of meetings, entity extraction.

Retrieval

Semantic search over internal company knowledge bases.

Solving the Knowledge Gap

RAG (Retrieval Augmented Generation).



Pre-trained models don’t know your private data. RAG connects the LLM to live enterprise data, reducing hallucinations and ensuring answers are factual.

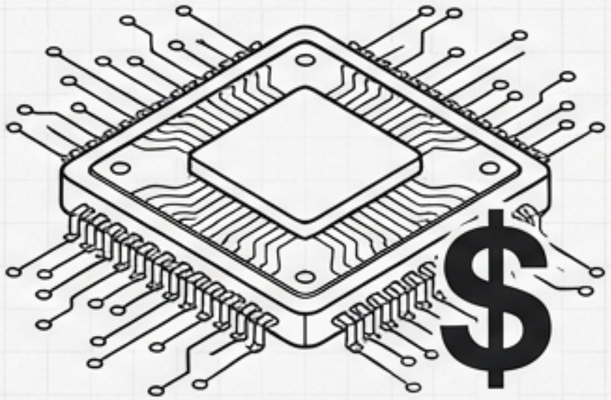
The Cost of Intelligence

Limitations and Governance.

Hallucinations

Models are probabilistic, not factual. They can confidently state falsehoods.

Compute Cost



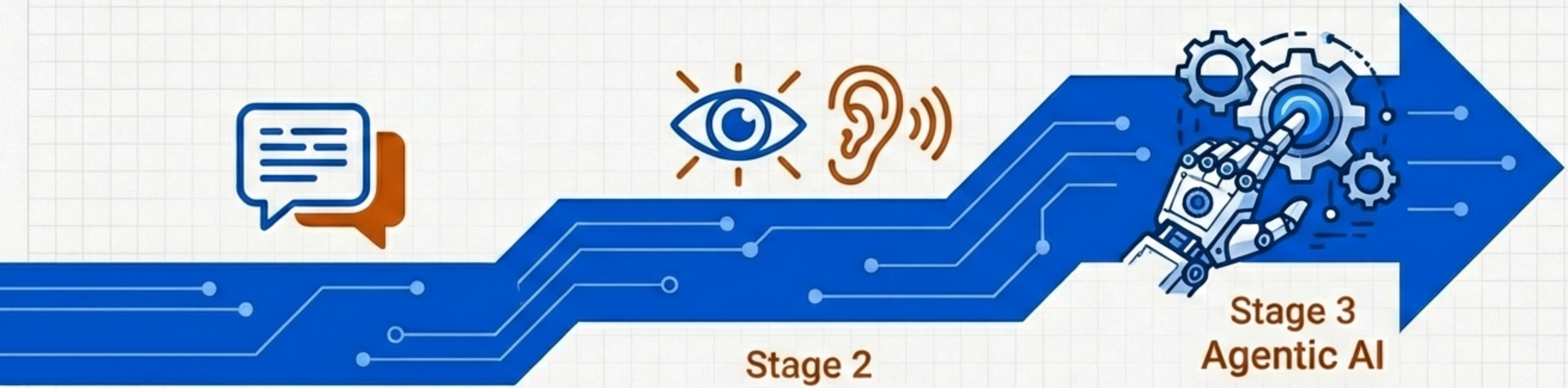
Training and inference require massive energy and GPU resources.

Bias & Safety

Risk of data leakage and propagation of training data biases.

The Next Horizon

From Chatbots to Agents.



Stage 1
Text-to-Text

Stage 2
Multimodal (Audio/Vision)

Stage 3
Agentic AI

Reasoning models that can plan, use tools, and execute actions autonomously.

The future of LLMs is not just about generating text, but taking action.