How AI Actually Works

> **Plain English:** When you type a message to an AI, it doesn't "think" or "know" things. It predicts the most likely next words based on patterns it learned from enormous amounts of text. Understanding this one idea changes how you use it.

The simplest explanation that's still true

Here's what happens when you send a message to ChatGPT, Claude, Gemini, or any other AI chat tool:

1. **Your message gets broken into tokens.** The AI doesn't read words the way you do. It splits your text into small chunks called tokens — sometimes whole words, sometimes pieces of words. The word "tokenization" might become "token" + "ization." This is why AI tools have limits on how much text you can send — they're counting tokens, not words. (For a deeper look at this, see Tokenization & Context Windows.)
2. **Each token gets converted into numbers.** The AI represents every token as a list of numbers (called an embedding) that captures its meaning and relationships. Words with similar meanings end up with similar numbers — "happy" and "joyful" are close together, "happy" and "wrench" are far apart.
3. **The model predicts what comes next.** This is the core of it. The AI looks at all the tokens in your message, weighs how they relate to each other (this is the "attention" mechanism you might have heard about), and predicts the most likely next token. Then it predicts the next one. Then the next. One token at a time, until it has a complete response.

That's it. No understanding. No reasoning in the human sense. Pattern matching at a scale and sophistication that produces remarkably useful output — but pattern matching nonetheless.

Why "trained on text" matters

Modern AI models like GPT-4, Claude, and Gemini were trained on enormous amounts of text — books, websites, code, conversations, research papers. During training, the model was repeatedly shown text with a word removed and asked to predict what goes there. Billions of times. Across billions of examples.

This is why AI can write in any style, answer questions about almost any topic, and generate code in dozens of languages. It's seen patterns in all of those domains.

But it also means:

- **AI doesn't "know" facts.** It learned that certain words tend to follow other words in certain contexts. When it tells you that Paris is the capital of France, it's not retrieving a fact from a database — it's producing the statistically likely continuation of your question.

  This is also why it can confidently state things that are wrong. (See: Why AI Gets Things Wrong)
- **Training has a cutoff date.** The model learned from text up to a certain point. It doesn't know what happened after that unless it has access to search tools or uploaded documents.
- **It reflects the patterns in its training data.** Including the biases, the common phrasings, and the popular opinions. AI doesn't have its own perspective — it has a statistical average of human text.

The three types you'll encounter

You don't need to memorize the full AI taxonomy, but understanding three distinctions helps:

AI is the broadest term — any system that performs tasks typically requiring human intelligence. Your email spam filter is AI. Siri is AI. A chess engine is AI.

Machine learning is a subset — systems that learn from data rather than following pre-written rules. Instead of programming "if the email contains 'Nigerian prince,' mark as spam," you show the system thousands of spam and non-spam emails and let it figure out the patterns.

Large Language Models (LLMs) are what you interact with when you use ChatGPT, Claude, or Gemini. They're a specific type of deep learning model trained on text, using an architecture called a "transformer." When people say "AI" in 2026, they usually mean this.

For the purposes of this playbook, when we say "AI," we mean the tools you actually use — the chat interfaces, the built-in AI features in your apps, the models you can give instructions to. The engineering details are fascinating (and if you want them, XueCodex goes deep), but you don't need them to build AI fluency.

What this means for how you use AI

Understanding that AI predicts rather than knows changes your approach in practical ways:

Give it more context, not less. The model predicts based on what's in front of it. A vague prompt gives it less to work with, so predictions are more generic. A detailed prompt with context, examples, and constraints gives it much better patterns to follow. This is why prompt engineering matters.

Verify, don't trust. Since the model is generating plausible text rather than retrieving verified facts, you need to check outputs — especially for specific claims, numbers, dates, and citations. The Fact-Check Habit exercise builds this skill.

It's a collaborator, not an oracle. AI is most useful when you bring judgment and it brings speed and breadth. You know your context, your goals, your stakeholders. It can process information, generate options, and stress-test your thinking faster than you can alone.

Different models have different strengths. GPT, Claude, and Gemini are trained differently, on different data, with different priorities. A prompt that works brilliantly in one may flop in another. This is why the playbook is tool-agnostic — we teach capabilities, not product-specific tricks.

Where to go next

- Why AI Gets Things Wrong — what hallucinations are and why they happen
- The Fact-Check Habit — your first exercise in working with AI critically

---

Revision #1
Created 2026-03-16 15:47:55 UTC by Admin
Updated 2026-03-16 15:47:55 UTC by Admin