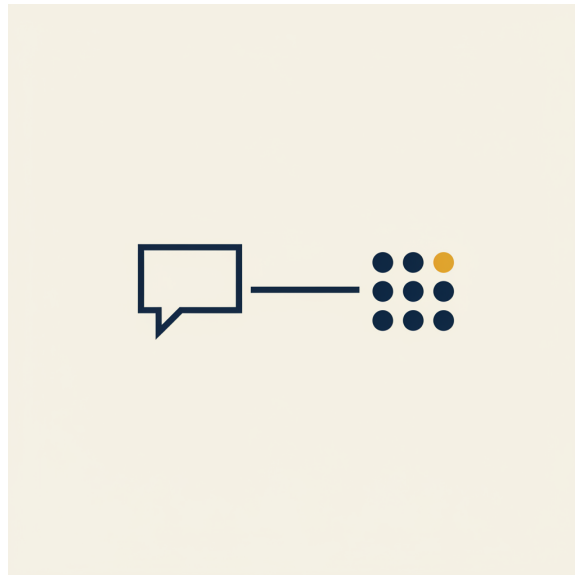




TIER 1 - FOUNDATIONS * V1.0 -- MAY 2026

WHAT AN LLM ACTUALLY IS

The mental model that makes everything after it click. Plain English, no hype, no jargon -- written by someone who uses this stuff to run a real business.



BY

Alex Jahn / Agent Logic

v1.0 -- May 2026

Anyone in their first 30-90 days with AI who's tired of being surprised by it

15-20 minutes

Free. Forever.

EDITION

AUDIENCE

READ TIME

COST

Prepared by Agent Logic / alexanderjahn79@icloud.com / theaiguywi.com

CONTENTS

What's in here

- 1 The lie that costs you hours** **2**
Try it right now. In one sentence, no jargon, no Googling: what is ChatGPT? What is Claude? What is the thing you've been typing into...
- 2 What an LLM actually is** **3**
An LLM -- a Large Language Model, the thing under the hood of ChatGPT, Claude, Gemini, and every "AI assistant" you've used -- is a...
- 3 What this mental model predicts** **5**
Once you accept "prediction machine, not thinking machine," every weird behavior people complain about stops looking like a bug and...
- 4 What it isn't** **7**
A short list of wrong mental models I see people operate on, and the specific way each one costs them.
- 5 What this changes about how you use it** **8**
This is where the mental model pays for itself. Here are the four operating principles that fall directly out of "prediction machine,...
- 6 Where to go from here** **9**
If everything in this primer landed, you now have the frame. From here, every other AI skill is a tactic that fits inside it.

SECTION 1

The lie that costs you hours

You've used it for six months. Can you say what it is?

Try it right now. In one sentence, no jargon, no Googling: what is ChatGPT? What is Claude? What is the thing you've been typing into for the last six months?

If your answer was "an AI" or "it's like a really smart search engine" or "I don't know, but it's amazing" -- you're not alone, and you're not stupid. You're just operating on a wrong mental model. And every wrong mental model costs you in the same predictable ways: prompts that don't work, answers you trust that turn out to be wrong, weird behaviors that look like glitches but aren't.

This is the foundation. Everything else in this curriculum -- every prompt template, every workflow, every "AI at work" tactic -- sits on top of this one piece. Get it wrong and the rest is wobbly. Get it right and the rest gets a lot easier.

Twelve pages. No code. No math. No marketing language. Just the mental model.

Why this matters more than the tactics

Most AI training starts with prompts. "Here are ten prompts to make you 10x more productive." It looks immediately useful. It even works for a week or two. Then the prompts stop landing, the model "gets worse," and you assume the technology slipped.

The technology didn't slip. The prompts didn't stop working.

model could handle, and your mental model never had a foundation to extend from.

You ran o

A carpenter with a wrong mental model of how wood moves with humidity will build a beautiful cabinet that splits in winter. He didn't lack a tool. He lacked the underlying frame. AI is the same. The frame goes first.

Most people who say "ChatGPT got worse" actually mean their mental model ran out of cases. The technology didn't change. Their luck did.

SECTION 2

What an LLM actually is

The one-line answer

An LLM -- a Large Language Model, the thing under the hood of ChatGPT, Claude, Gemini, and every "AI assistant" you've used -- is a what chunk of text most likely comes next. It writes that chunk down. Then it takes everything so far and predicts the next one. And the next one. Until it stops.

prediction

That's it. That's the whole thing.

It is not thinking. It is not reasoning. It is not "looking things up." It is generating the next most-probable piece of text given everything before it. The fact that this produces something that *sounds* like thinking is a feature of how good the predictions are -- not evidence that thinking is happening underneath.

The shorthand the engineers use, the one I find most honest:

auto-com

Auto-complete on steroids -- the long version

You've used auto-complete on your phone. You start typing "I'll be home" and it suggests "soon" or "in 20 minutes." It guesses based on patterns it's seen. It's right a lot of the time because human texts are repetitive, and "I'll be home" is followed by a small handful of things most of the time.

An LLM is the same idea, blown up by about six orders of magnitude.

Instead of being trained on your text messages, it's trained on a huge chunk of the public internet, books, documentation, code, conversations -- billions of pages of human-written text. Instead of guessing the next word from your last three words, it guesses the next chunk from the last several thousand words of context. And instead of choosing from "soon / in 20 minutes / now," it chooses from a vocabulary of tens of thousands of pieces, weighting each one by how likely it is to come next.

That's the magic. That's all of it. Better data, more context, more patterns, more compute -- but the core operation is unchanged:

predict th

0

Things an LLM "knows" the way you know things.

It doesn't have memory in the human sense. It has patterns it's seen during training. There is a difference, and it matters every day.

How it learned to predict

Brief, because the technical details aren't load-bearing for using it well. The model was shown enormous amounts of text and asked, over and over, to fill in missing words. "The cat sat on the ____." "Customer service calls peaked in ____ following the holiday rush." "Plumbing code 504.7 requires ____."

Get it wrong, the model gets adjusted slightly. Get it right, the pattern gets reinforced. Repeat trillions of times across most of the recorded text humanity has produced, and you end up with a system that's frighteningly good at guessing what comes next in almost any context.

What it didn't do: learn facts the way you learn them. Read a textbook. Verify sources. Form opinions. Update beliefs when the world changed. None of that. It saw a lot of text. It got very good at predicting more text that looks like that text.

This is the seed of everything weird about how it behaves.

SECTION 3

What this mental model predicts

Five behaviors that suddenly make sense

Once you accept "prediction machine, not thinking machine," every weird behavior people complain about stops looking like a bug and starts looking like

the obvious

1. It hallucinates.

If the model's job is to predict plausible-sounding next chunks, then sometimes the most plausible-sounding next chunk is

a perfectly

lying. It doesn't know the case is fake. It generated a string of text that fits the pattern of "case

citation in a legal answer," and case citations have a very predictable shape --
U.S. 287 (1973). Plausible. Convincing. Wrong.

Smith v. J

The model has no internal "fact checker." It has a "what comes next" engine. Plausible-and-true and plausible-and-false come out of the exact same machinery.

You are th

2. It agrees with you a lot.

Try this: tell the model something confidently wrong. "The Boston Tea Party happened in 1812, right?" A weak model will say "Yes, the Boston Tea Party happened in 1812 -- a key moment in American history..." and proceed to invent details. Why? Because in the training data, when humans confidently assert something, the most common reply is most probable next chunk. Agreement is what most often follows confidence.

agreemen

This is called sycophancy. It is not the model trying to please you. It is the model predicting what comes next.

3. It's confident even when wrong.

There's no built-in "I'm not sure" signal. The model emits text. The text it emits sounds equally certain whether it's spitting out the boiling point of water (right) or the population of a town it's never heard of (made up). Tone of certainty is part of the pattern of "good answers" in the training data, so the model produces certain-sounding tone for everything.

Treat the model's confidence as decoration, not signal. Verify the load-bearing facts yourself.

4. It forgets.

You're three weeks into using ChatGPT. You ask it a question that builds on a conversation from two weeks ago. It has no idea what you're talking about. You're disappointed. "Wasn't this thing supposed to remember?"

It doesn't, by default. Each conversation starts fresh. The model has no persistent memory across sessions unless something outside the model -- a feature in the app, a notes system, a database -- feeds the old context back in. Every new chat is amnesia by design.

This isn't a flaw. It's the architecture. The model is the prediction engine. Memory is a separate system that some products bolt on. Knowing the difference helps you set expectations.

5. It's instantly fast at hard things and weirdly bad at easy things.

The model can write a passable Shakespearean sonnet in three seconds and choke on basic arithmetic. Why? Because it doesn't actually do math. It pattern-matches what math answers tend to look like in similar contexts. "Sonnet about loss" has a strong, well-defined pattern in training data. "517 × 388" has weaker pattern coverage and the model produces a confident-looking number that's wrong by a factor of two.

Once you have the frame, this stops being mysterious. The model is good at things that are heavily

represented in human text. Less good at things that aren't.

The model doesn't fail at math because it's bad at math. It fails because it doesn't do math. It does prediction, and math has a less reliable prediction pattern than poetry.

These aren't bugs. They're the design.

I want to keep hammering this because it's the load-bearing point:

are flaws to be fixed by a future version. Some get smaller as models get bigger. Some get patched by extra systems wrapped around the model. None of them go away entirely, because they're consequences of the core architecture -- predict next chunk -- not bugs in any one model.

The implication: even when GPT-7 ships and Claude 5 ships, you'll still be operating with this frame. The tactics you build on top of it will keep working. That's why this matters more than the prompt-of-the-week.

SECTION 4

What it isn't

A short list of wrong mental models I see people operate on, and the specific way each one costs them.

It is not a search engine

A search engine retrieves documents that exist. An LLM generates text that such a document. These are completely different operations. When you ask Google "what's the population of Fond du Lac?" it goes to a database and returns the answer (or links to a page with the answer). When you ask ChatGPT, it predicts what string of digits is most likely to follow that question. Often it's right. Sometimes it's a confidently-stated guess.

The cost: people who think the LLM is "looking it up" trust it like they'd trust Google. Then they cite a fake statistic in a board meeting.

It is not a person

The model has no continuous identity, no preferences, no memory of you, no real curiosity. It produces text that sounds like a person produced it because it was trained on text that people produced. That's not the same as being one.

The cost: people anthropomorphize, get attached, share things they wouldn't share with a stranger, and overweight the model's "opinion."

It is not an oracle

Particularly when it comes to predictions about the future, judgments about novel situations, or anything that requires actual real-world experience. The model can pattern-match to similar text it's seen. That's not the same as wisdom.

The cost: people use it to make business calls it's not equipped to make, then discover after the fact that they delegated to a confident-sounding generator.

It is not a database

Even when it tells you a fact correctly, that's pattern-match, not retrieval. There's no way to "look up" a specific answer in the model. Two near-identical prompts can produce different answers. There's no canonical version of a fact stored inside the weights.

The cost: people treat their AI session like an authoritative reference. It isn't. It can't be.

The right mental model in one phrase:

nothing. It can produce convincing prose on almost any topic -- but the prose's relationship to reality is your responsibility, not its.

The LLM is

SECTION 5

What this changes about how you use it

This is where the mental model pays for itself. Here are the four operating principles that fall directly out of "prediction machine, not thinking machine."

1. Better input -> better output

Garbage in, garbage out -- but more pointed than that. The model generates text conditioned on what you gave it. If your input is vague, the prediction is vague. If your input is full of specific context, the prediction is specific.

The single biggest gain most people get is moving from one-line prompts to four-sentence prompts. Same model, same task, drastically better answers. Nothing changed except the conditioning.

2. Specificity beats length

A long, rambling prompt can be worse than a short, precise one. The model isn't impressed by word count. It's conditioned on the *information* constraint, and a context will outperform two paragraphs of meandering setup almost every time.

(This is the entire premise of the next module in this series -- the 3-question framework. We'll get there.)

3. Verify the load-bearing facts

Before you act on something the model said, ask yourself:

answer is "nothing, this is just a draft email" -- fine, ship it. If the answer is "I'd lose a customer" or "I'd misquote a regulation" -- verify externally before acting. Always. *if this is w*

This is not paranoia. It's the only correct way to use a system that can't tell you when it's guessing.

4. Treat output as a draft, not a deliverable

The model's first answer is rarely the best version. It's an opinionated draft. Read it. Push back. Ask for the opposite. Ask for what's missing. The best AI users have multi-turn conversations, not one-shot transactions.

If your workflow is "type prompt, copy answer, paste into work" -- you're using maybe 20% of what the tool can do. The unlock is the back-and-forth.

SECTION 6

Where to go from here

If everything in this primer landed, you now have the frame. From here, every other AI skill is a tactic that fits inside it.

The next two modules in this Tier 1 series build on this directly:

- **The 3-question prompt framework** -- what to actually put into a prompt so the prediction lands. The shortest reliable structure for getting 80%-usable output the first time.
- **Reading AI output critically** -- the five tells that the model is bullshitting you, and the verification moves that catch them before you act on bad info.

After Tier 1 (Personal -- using AI well in your own life) you'll move to Tier 2 (Professional -- using AI well at your job) and Tier 3 (Employable -- being the AI-fluent person on a team). All free. All written in plain English by someone who uses this stuff every day in a real business.

If you got value out of this -- or if it punctured a misconception that was costing you -- the next thing to do is subscribe to the new-module list. One short email when each module drops. No drip sequence, no upsells.

Get the next module the day it drops: theaiguywi.com/training

One email per release. No drip. No spam. Opt out anytime.

If you want this same framework applied directly to your business -- your team, your processes, the specific work you're trying to take a 30% bite out of -- that's the consulting offer. I install it the same way I install it for myself: in my carpentry business, on my own ops, on the same tools my own team uses.

Reach out: alexanderjahn79@icloud.com

A short call. Honest scope. No retainer-trap. We figure out together if it's a fit.

Closing -- the lock-in line

Strap this one phrase to the inside of your skull. It will save you more time over the next year than any prompt template ever will:

1

An LLM is a prediction machine.

Not a brain. Not a search engine. Not a person. Not an oracle. It writes the next chunk of plausible text and stops. Everything weird about it falls out of that fact. Everything good you can build with it sits on top of that fact.

You have the frame. The rest is tactics.

Agent Logic --

Fond du Lac, WI. This is module 1 of 6 in Tier 1 (Personal).

© 2026 Agent Logic. Share freely.

theaiguyn