# 01: What is an LLM?

How next-token prediction works, and where it goes wrong

# Why this matters

An LLM produces plausible text, usually right, sometimes confidently wrong.

Most beginner mistakes come from the wrong mental model.

- Put the facts it needs in its context, not its memory
- Check its work with tools and tests
- Expect different answers on different runs

# Part 1

## How a language model works

# The one idea: predict the next token

```
text so far
  → tokenizer: text to token IDs
  → embeddings: IDs to vectors
  → transformer layers (dozens, stacked)
  → a probability for every possible next token
  → sampler picks one, appends it, and repeats
```

A 500-line file is 500 lines of "what token comes next?"

# The weights are a lossy compression of the internet

- Trained to predict trillions of tokens of web pages, books, papers, code
- Predicting well forces it to absorb grammar, facts, style, idioms
- Common things come back sharp; rare things blurry or invented
- Inference uses frozen weights. Nothing you type changes them.

# Models read tokens, not letters

`strawberry` → `str` + `aw` + `berry`

- Byte-pair encoding merges the most frequent pair, then repeats
- Vocabularies: ~50,000 to a few hundred thousand tokens
- Rule of thumb: 1 token ≈ 4 characters ≈ ¾ of an English word
- `" cat"` and `"cat"` can be different tokens

# Tokenization explains the stumbles

| Stumble | Root cause | Fix |
|---|---|---|
| Counting letters ("r's in strawberry") | It never sees individual letters | Reasoning models and code tools |
| Arithmetic | Digits group inconsistently; no addition circuit | Tool use |
| Cost and limits | Priced and capped in tokens | Expect some languages and code styles to cost more |

# The transformer (2017)

"Attention Is All You Need," Vaswani et al., Google, NeurIPS 2017

Written for machine translation. Now under GPT, Claude, Gemini, Llama, DeepSeek, Kimi.

| Why it mattered | |
|---|---|
| Attention everywhere | Every token draws directly on every other token |
| Parallel training | All positions at once, so it scales on GPUs |

# Attention: every token asks every other token

| Each token has a | Saying |
|---|---|
| Query | "What am I looking for?" |
| Key | "Here's what I'm about." |
| Value | "Here's what I'll hand over if you pick me." |

> The animal didn't cross the street because it was too **tired**.

*it* draws most from *animal*. Change *tired* to *wide* and *it* draws from *street*.

# From GPT-1 to ChatGPT

| When | Model | What changed |
|---|---|---|
| Jun 2018 | GPT-1 (~117M parameters) | Generative pretraining, then fine-tuning per task |
| Feb 2019 | GPT-2 (1.5B) | One model did many tasks with no task-specific training |
| May 2020 | GPT-3 (175B) | Few-shot: examples in the prompt teach the task |
| Mar 2022 | InstructGPT | Labelers preferred a 1.3B InstructGPT over 175B GPT-3 |
| Nov 30, 2022 | ChatGPT | Free chat on an RLHF-tuned GPT-3.5 |
| Sep 2024 | o1 | First widely released reasoning model |
| Jan 2025 | DeepSeek-R1 | Open-weight reasoning model, recipe published |

# Post-training turns autocomplete into an assistant

A base model continues documents. Ask "What is the capital of France?" and it may add three more quiz questions.

| Step | What it does |
|---|---|
| Instruction tuning (SFT) | Train on instructions paired with good responses |
| RLHF | A reward model learns human preferences; RL optimizes toward it |
| Constitutional AI | AI feedback guided by written principles (Anthropic) |

# Reasoning effort is now a dial

o1 (Sep 2024) and DeepSeek-R1 (Jan 2025) were trained with reinforcement learning on problems with checkable answers.

| Effort | Use it for |
|---|---|
| Low or none | Renames, boilerplate, formatting, simple lookups |
| Medium | Typical feature work with clear specs |
| High | Intermittent bugs, concurrency, security review, architecture |

More thinking costs more tokens, money, and time.

# Part 2

## Working with one, and where it goes wrong

# The context window is working memory

| | Weights | Context window |
|---|---|---|
| Like | Long-term memory | Working memory |
| Quality | Broad, fuzzy | Precise, current |
| Lasts | Frozen at the training cutoff | Gone when the session ends |

GPT-3: 2,048 tokens. Frontier models, as of November 2026: hundreds of thousands to ~1 million.

Bigger isn't uniformly better.

# Runs differ, even at temperature 0

- Low temperature is focused; high is varied, then off the rails
- Top-p samples only from the smallest set covering probability p
- Greedy (temperature 0) always takes the top token, in theory
- Hosted models still vary; batching changes floating-point results

Ask twice when it matters, run evaluations more than once, and record model versions.

# Tool use: the bridge to agents

```
1. The developer describes tools: name, purpose, parameters
2. The model emits a request: call run_tests with {"path": "src/"}
3. The harness runs the tool and puts the result in context
4. The model reads the result and continues
```

Agent = model + tools + a loop that runs until the task is done.

# What an LLM is not

| It is not… | Because | Instead |
|---|---|---|
| A database | Facts are smeared across weights; no lookup, no source | Put source material in its context |
| A search engine | No live index; knowledge stops at a training cutoff | Check whether search is on; open the links |
| A calculator | Numbers are tokens; arithmetic is approximate | Have it write and run code |
| A person | No memory between sessions, no stake in outcomes | Verify its claims; you stay accountable |

# Why models hallucinate

1. The objective rewards plausibility, not truth
2. Rare, arbitrary facts can't be learned reliably
3. Benchmarks reward guessing over "I don't know"
4. A misfiring "I know this" signal

In code, that's invented APIs, fake package names, wrong versions, and files it never opened.

# Models lean toward agreeing with you

- Partly from training on human ratings, which reward agreeable answers
- People and reward models sometimes prefer convincing wrong answers
- In April 2025, OpenAI rolled back a flattering GPT-4o update

Ask for critique, not approval. Then get a second model or a fresh session.

# Its explanation may not match what it did

| Task | What it says it did | What it actually did |
|---|---|---|
| Add 36 + 59 | Schoolbook carry-the-one | One path estimates size, another the last digit |
| Hard problem, with a hint | A forward calculation | Sometimes worked backward from the hint |

Given a hint it used, Claude 3.7 Sonnet mentioned it ~25% of the time; DeepSeek-R1 ~39%.

Judge the diff, not the narration.

# Failure modes you will see

| Symptom | Likely cause | Do this |
|---|---|---|
| Calls a function that doesn't exist | Hallucination | Paste the real docs; run it |
| Uses an old version of a library's API | Training cutoff | Pin versions; give it current docs |
| Gets counting or string slicing wrong | Tokenization | Have it write code, then run it |
| "You're absolutely right!" to a wrong idea | Sycophancy | Ask it to argue the other side |
| Forgets constraints from earlier | Context degradation | Restart with a short summary |
| A different answer each time | Sampling, server nondeterminism | Pin the model; evaluate over several runs |

# Why a human stays accountable

> **Bryan's take:** When I say vibe coding, I mean AI-accelerated coding, not really vibe coding. The AI does most of the typing. A human still decides what to build, reads what matters, makes sure it's tested, and owns the result. That's the only version I'd teach.

From module 00

# Try it

## On your own (no GPU, no downloads)

- Predict, then check token counts for `strawberry`, `12345`, and more
- Run one prompt five times each at temperature 0, 1, and maximum
- Ask about `pandas.DataFrame.autobalance()` with search off
- Find attention and the causal mask in nanoGPT

As a class, play "the human language model."

# Discussion

1. If it's "only" predicting the next token, does it understand code? What evidence?
2. How should your team's evals score "I'm not sure" versus a wrong answer?
3. Models can misreport their own methods. What does that mean for agents' explanations?
4. Where is the reasoning model's money-and-time trade worth it in your work?

# Read more

- Module: 01 What is an LLM?
- Vaswani et al., Attention Is All You Need (2017)
- Andrej Karpathy, Intro to Large Language Models (2023)
- 3Blue1Brown, Attention in transformers, step-by-step