AI LITERACY SERIES

What is an AI model, really?

A plain-English guide to what is actually happening when you use AI

Ryan Bishop | April 2026 | ryanbishop.co.uk

1. AI is not one thing

Before asking what an AI model is, it helps to be clear about what AI is — because the word covers an enormous range of systems that work in very different ways.

The spam filter that keeps junk out of your inbox is AI. So is the algorithm that decides which posts appear in your social media feed. The system that flags suspicious transactions on your bank account. The tool that recommends your next Netflix series. The software that reads an X-ray and flags areas of concern for a radiologist to review.

None of these systems work the same way. They were built for different purposes, trained on different data, and operate through different mechanisms. What they share is that they involve systems which learn from data and improve through experience, rather than simply following fixed instructions written by a developer.

This episode focuses on the type of AI most people encounter most often today: **large language models** — the technology behind ChatGPT, Claude, Gemini, Copilot, and similar tools. But it is worth holding onto the broader picture. When we talk about AI in healthcare, in hiring, in content moderation, we are often talking about very different systems with very different implications. The large generative model you use to draft an email is not the same kind of system as the one your hospital might use to assist in diagnosis.

2. Start with a recipe

The philosopher Virginia Dignum offers an analogy that cuts through the complexity better than most technical explanations: an algorithm is a recipe.

A recipe is a set of precise instructions designed to produce a specific result. Every time you add two numbers together, you are following an algorithm — a recipe for addition. When you bake an apple pie, you are following an algorithm — a sequence of steps applied to a set of ingredients to produce an output.

But here is the crucial observation Dignum makes: **by itself, a recipe has never turned into an apple pie.** The result depends on your ingredients — their quality, their freshness, their proportions. And it depends on the skill and choices of the person who prepared them.

The same applies to an AI model. But the recipe analogy goes further than just ingredients. It also captures the technique — the skill and the choices made at every stage of preparation. A model does not simply absorb data and emerge ready to use. It develops through a process of exposure, adjustment, and feedback, shaped by deliberate human decisions throughout.

How a model actually learns

A traditional computer program is given rules by a developer: if X happens, do Y. Every outcome is deliberate and traceable. The developer controls the logic explicitly.

A large language model works differently. Instead of being given a rulebook, it is exposed to enormous quantities of text and develops internal parameters — numerical values that get adjusted throughout training as the model learns what good responses look like. Nobody wrote rules telling the model how to structure a paragraph or answer a question. It developed patterns from billions of examples.

Human choices shape this process at every stage. In supervised learning, human-labelled examples guide the model toward correct outputs. Through a process called Reinforcement Learning from Human Feedback, human raters evaluate the model's responses and those preferences are used to further refine its behaviour. The model's parameters are not random, and they are not a rulebook — they are the accumulated result of the training process and the human judgements woven through it.

This matters because it means the model's behaviour reflects the choices made during training — who labelled the data, what they considered a good response, what was included and what was left out. As Dignum observes, the result depends on the ingredients and the technique, not just the recipe itself.

The recipe in plain English

Algorithm = the recipe. The mathematical structure the model learns through.

Training data = the ingredients. The text, images, and other data the model learned from.

Human feedback and choices = the technique. The skill and decisions applied throughout preparation.

The model = the finished pie. The result of applying recipe, ingredients, and technique.

Your prompt = a slice request. The model uses what it learned to serve up a response.

3. Pattern matching — the engine behind the response

When you type a prompt into a large language model, what happens is best understood as pattern matching at extraordinary scale.

The model processes your input and draws on the statistical patterns it developed during training to produce a response. Each part of its output reflects what it has learned tends to follow — given everything that came before in the conversation, and everything it absorbed during training.

This is not retrieval of stored facts. It is not step-by-step reasoning in the way a person works through a problem. It is the model completing patterns based on what it learned.

That distinction matters — and it is worth being honest about what it means. At sufficient scale and sophistication, pattern matching can produce outputs that feel like reasoning. A well-trained model can work through complex problems, explain its thinking, and arrive at correct conclusions. The outputs can be genuinely impressive. But the mechanism is still fundamentally one of pattern completion, not understanding.

Why does this matter? Because pattern matching has a characteristic failure mode: the model can produce fluent, confident, entirely wrong responses when the pattern leads somewhere plausible but inaccurate. The response sounds right. It reads as authoritative. But the model has no way of knowing it is wrong, because it is not checking against truth — it is completing a pattern. That is the subject of Episode 5.

Pattern matching vs reasoning — why the distinction matters

A person who does not know the answer to a question can recognise their uncertainty and say so. A large language model completing a pattern has no equivalent check. It produces the most plausible-

sounding continuation of the text — whether or not that continuation is accurate. Understanding this is the most practically important thing anyone using these tools can know.

4. When AI searches — the retrieval layer

The pattern matching description above applies to a base large language model operating on its training data alone. But many AI tools now do something additional before generating a response: they retrieve information.

ChatGPT can search the web. Copilot can search your documents. Claude can be connected to knowledge bases. Enterprise deployments routinely combine a language model with a document library, database, or live data source. This approach — connecting a model to an external information source — is known as retrieval-augmented generation, or RAG.

The retrieval layer changes what the model has access to. Instead of relying solely on patterns from its training data, it can draw on current, specific, or proprietary information that was not part of training.

But it is important to understand that retrieval does not replace pattern matching — it sits alongside it. The model still uses pattern matching to process the retrieved information, synthesise it, and generate a response. The quality of that synthesis still depends on the model's training and the choices made during it. Retrieving accurate information does not guarantee an accurate response if the model misrepresents, misinterprets, or inappropriately combines what it found.

Two layers, working together

Retrieval layer: finds relevant information from an external source — a web search, a document library, a database.

Model layer: processes and presents that information using pattern matching. This is where synthesis, summarisation, and generation happen.

Understanding which layer you are depending on in any given interaction helps you calibrate how much to trust the output — and what to verify.

5. The model and the product — they are not the same thing

One of the most practically important distinctions that most people miss: the model and the product you use are different things.

Claude is a model, developed by Anthropic. Claude.ai is a product built on that model — with a specific interface, configuration, guardrails, and deployment layer applied on top. ChatGPT is a product built on OpenAI's GPT models. Gemini is Google's product built on their Gemini model family.

The same underlying model can behave very differently depending on how it has been configured. A model accessed directly with no instructions will behave differently to the same model deployed inside a consumer product with safety filters, persona instructions, and usage policies applied. A model deployed internally by a pharmaceutical company with domain-specific guidance will behave differently again.

This matters in two practical ways:

- When comparing AI tools, you are often comparing products and configurations, not just models. The model is one input into the experience you get — the deployment layer is another.
- When an organisation deploys AI — building a tool for internal use, a customer-facing application, a document assistant — they are making choices about that deployment layer. Those choices carry responsibility, because they shape how the model behaves for that context.

The training process encodes human choices into the model. The deployment layer encodes further human choices on top. At every level, the behaviour of an AI system reflects decisions made by people. That is not a flaw in the technology — it is the nature of it.

6. For families — explaining it to your children (and yourself)

The recipe analogy works well for children. Most children understand that a recipe does not cook itself, and that the same recipe produces different results depending on the ingredients and the cook. That intuition transfers directly to how AI models work.

A few things worth explaining to younger users of AI tools:

AI is a family of very different things

The AI your child encounters ranges from the recommendation algorithm on YouTube, to the spell-checker in their word processor, to a large language model like ChatGPT. These are not the same kind of system. Helping children understand that AI is not one thing — and that different AI systems work differently and should be trusted differently — is a foundation for critical thinking.

It learned, it did not look it up — mostly

When your child asks an AI a question and gets an answer, it is usually producing what it learned to produce based on patterns — not checking a trusted database. Some tools do retrieve web information first, but even then the model is processing and presenting that information through pattern matching. Verification matters regardless. The answer can be fluent, confident, and wrong.

The same AI can behave differently in different places

ChatGPT on a phone, Copilot inside Microsoft Word, and a customer service chatbot on a retail website may all be built on similar underlying models — but they behave differently because they have been configured differently. This is worth explaining to children who may assume AI is one consistent thing with one consistent level of reliability.

Human choices are inside every AI system

The people who built the model made choices about what data to use, how to train it, and what kind of responses to encourage. Those choices are baked into every response the model produces. AI is not neutral, and it is not objective — it reflects the choices of the people who made it.

A conversation starter for families

Ask your child: if you followed the same recipe but used different ingredients, would the pie taste the same? Now ask: if an AI model learned from different information, or if the people training it made different choices, do you think it would give different answers?

That conversation builds the critical thinking habit that matters most when using AI: the outputs reflect the inputs and the choices made during training, and those inputs and choices are not always visible to the user.

7. For organisations — what this means in practice

The framework above has direct implications for how organisations should think about AI procurement, deployment, and accountability.

Which AI are you using is often the wrong question

When a team asks which AI tool to use, the model name is less important than how the model is configured and what it is being asked to do. A well-configured deployment of a mid-tier model will outperform a poorly configured deployment of a frontier model for a specific use case. Procurement conversations need to go beyond the headline model name to the deployment layer — what instructions has the model been given, what data can it access, what guardrails are in place.

The training data shapes the outputs

Because models learn from their training data, the characteristics of that data are encoded in the model's patterns. A model trained predominantly on one language will perform differently on others. A model with a training cutoff date will not know about events after that date — even if the product has a retrieval layer for some queries. A model trained on data that contains bias will reproduce that bias in its outputs, not because anyone designed it to, but because the patterns were there.

Understanding this is foundational to responsible AI use in any professional context. The question is not just what the model can do — it is what it learned, from what, and when.

Configuration is accountability

When your organisation deploys an AI tool — configuring it, giving it instructions, connecting it to your data, applying it to a specific use case — you are shaping how it behaves for your context. The choices you make in that deployment layer carry responsibility.

This connects directly to the EU AI Act's provider and deployer distinction explored in Episode 1. Understanding what a model is — and recognising that human choices at every layer of its development and deployment determine how it behaves — is the foundation for responsible AI governance. As Dignum puts it, AI systems are artifacts created by people, and the responsibility for their behaviour rests with the humans and organisations involved at every stage.

Coming next — why AI makes things up

If a model works by completing patterns based on what it learned — rather than reasoning through problems or retrieving verified facts — what happens when the pattern leads somewhere that sounds right but is wrong?

That is the question Episode 5 addresses. AI hallucinations are not a mystery or a malfunction. They are a logical consequence of how models work, and understanding that mechanism is one of the most practically important things anyone using these tools can know.

Episode 5: ***AI hallucinations — why does it make things up?***

Sources & Reference

This episode draws on academic literature in AI ethics and responsible AI, combined with original analysis by Ryan Bishop.

Recipe and algorithm analogy: Dignum, V. (2020). 'Responsibility and Artificial Intelligence.' In Dubber, M.D. et al. (eds.), *The Oxford Handbook of Ethics of AI.* Oxford University Press. https://doi.org/10.1093/oxfordhb/9780190067397.013.12. Pages 214–231.

Key quotation: "By itself, a recipe has never turned into an apple pie. The end result of your pie has more to do with your baking skills and your choice of ingredients. The same applies to AI algorithms: in large part the result depends on the input data and on the ability of those that trained it." — Dignum (2020), p. 228.

AI systems as artifacts: Dignum (2020) argues that AI systems are artifacts designed by people, and responsibility for their behaviour rests with the humans and organisations involved in their design, deployment, and use.

Want to discuss what responsible AI implementation looks like for your organisation? Get in touch: ryan@ryanbishop.co.uk

RAI Disclosure: *This content was produced with AI assistance, reviewed and contextualised by Ryan Bishop. The process is intentional — this is augmentation in action.*