Machine Learning Models Across Sectors

A practical mapping of 8 model types — definitions and real-world use cases in healthcare, education, and finance

Classic vs. generative models

Proprietary vs. open-source models

Language models vs. multi-modal models

Large vs. small language models

1. Classic (Discriminative) Models

Classic models learn to map inputs to outputs for tasks like classification, regression, or ranking. Rather than creating new content, they answer questions such as "what category does this belong to?" or "what number does this predict?" Examples include logistic regression, decision trees and standard CNNs used for image classification.

Sector	Example Use Case
Healthcare	Predicting readmission risk from patient records using logistic regression or random forests; tumour classification (benign/malignant) from imaging features
Education	Predicting student dropout risk from attendance and grades; automated grading of multiple-choice or short numeric answers
Finance	Credit scoring and loan default prediction; fraud detection flagging anomalous transactions

2. Generative Models

Generative models learn the underlying distribution of data and can produce new content, including text, images, and audio. Examples include GPT-style transformers, diffusion models ... They answer the question "what would a plausible example of this look like?"

Sector	Example Use Case
Healthcare	Generating synthetic patient data to train other models without privacy risk; drafting clinical notes from doctor-patient conversations
Education	Auto-generating practice questions, essay feedback, or personalised study material
Finance	Drafting financial reports or earnings summaries; generating synthetic transaction data to stress-test fraud models

Comparison: Classic vs. generative models

Classic models are typically simpler, cheaper to train, and easier to interpret since they're solving a narrower task. Generative models need to model much more complex distributions, so they require more data, compute, and parameters - but they can create rather than just categorize. In practice, generative models are increasingly built on classic techniques internally (e.g., a diffusion model still uses discriminative-style noise prediction under the hood).

3. Proprietary Models

Proprietary models are closed-weight systems accessible only through an API or product, controlled by a company. Examples include GPT-4/5-class models, Claude, and Gemini. They often lead on raw capability and come with hosted infrastructure and support, but offer no transparency into weights or training data.

Sector	Example Use Case
Healthcare	Hospitals using GPT-4 or Claude via API (with appropriate data agreements in place) for clinical documentation assistance
Education	Ed-tech platforms built on GPT-4, for tutoring
Finance	Banks using proprietary models via secure enterprise API for internal research summarisation or client report drafting

4. Open-Source Models

Open source (open-weight) models have publicly downloadable weights that anyone can run, fine-tune, or modify. Examples include Llama, Mistral, DeepSeek, and Qwen. They trade some cutting-edge performance for transparency, customisability, and the ability to run locally without ongoing API costs.

Sector	Example Use Case
Healthcare	Hospitals fine-tuning Llama or Mistral on-premises to keep patient data in-house for regulatory compliance (e.g., HIPAA)
Education	Universities running open-weight models locally for research without per-query costs
Finance	Fintech firms fine-tuning open models on proprietary trading data without exposing it to a third-party API

Comparison: Proprietary vs. open-source models

Proprietary models often lead on raw capability and come with hosted infrastructure, support, and safety tuning, but offer no transparency into weights or training data and lock users into a vendor. Open-source models trade some cutting-edge performance for transparency, customizability, and the ability to run locally or fine-tune for niche use cases without ongoing API costs. The gap between the two has been narrowing over time.

5. Small Models

Small models, roughly in the range of millions to a few billion parameters, are lightweight, fast, and cheap to run, often deployable on a phone or laptop. Examples include small Llama or Gemma variants. They excel at narrow, well-defined task, where speed and efficiency matter most.

Sector	Example Use Case
Healthcare	On-device models in wearables detecting irregular heartbeat in real time
Education	Lightweight models running offline on school tablets for basic language practice in low-connectivity areas
Finance	Real-time fraud scoring at point-of-sale terminals requiring millisecond latency

6. Large Models

Large models, ranging from tens of billions to trillions of parameters, capture broader and more nuanced patterns and generalise well across diverse tasks. Examples include GPT-4-class models and large Llama variants. They require significant compute to train and serve, but excel at general reasoning and open-ended tasks.

Sector	Example Use Case
Healthcare	Complex diagnostic reasoning across multiple symptoms, history, and literature (e.g., research-stage diagnostic assistants)
Education	Full tutoring systems that reason across an entire curriculum and adapt explanations to a student's level
Finance	Large-scale market analysis integrating macroeconomic data, news, and filings to generate investment insights

Comparison: Language models vs. multi-modal models

Small models win on latency, cost, privacy (on-device), and energy efficiency id-eal for narrow, well-defined tasks. Large models win on general reasoning, few-shot learning, and handling ambiguous or open-ended tasks, at the cost of speed and expense. A common strategy is using a large model to generate training signal that distils down into a small, efficient one.

7. Language-Only Models

Language-only models process and generate text (or code as a form of text) exclusively. Examples include early GPT-3. They are more focused and computationally lighter than multimodal systems since they work with a single modality's structure.

Sector	Example Use Case
Healthcare	Summarizing electronic health records into concise clinical summaries
Education	Essay feedback and writing tutors; text-based Q&A chatbots for homework help
Finance	Summarizing earnings call transcripts; chatbots answering account questions in text form

8. Multimodal Models

Multimodal models handle multiple input or output types together, such as text, images, audio, and sometimes video. Examples include GPT-4V, Gemini, and Claude with vision. Aligning representations across different data types is architecturally harder and more data-hungry, but unlocks tasks language alone cannot handle, such as describing an image or reasoning about a diagram alongside text.

Sector	Example Use Case
Healthcare	Reading X-rays or MRIs alongside patient notes to assist radiologists (image + text reasoning)
Education	Tools that read a student's handwritten math work (image) and explain errors in text
Finance	Analysing scanned invoices, checks, or ID documents combined with text data for KYC/compliance checks

Cross-Cutting Example

A hospital might combine several of these model types at once: a small, classic model triages incoming patients by risk score in real time; a large, multimodal, proprietary model helps radiologists interpret scans alongside notes; and an open-source, generative model fine-tuned locally drafts discharge summaries to keep sensitive data in-house.

Comparison: Large vs. small language models

Language-only models are more focused and computationally lighter since they work with a single modality's structure. Multimodal models require aligning representations across different data types (e.g., mapping pixels and words into a shared space), which is architecturally harder and data-hungrier, but they unlock tasks language alone can't - like describing an image, reading a chart, or reasoning about a diagram alongside text.