What is AI, Anyway?

redvespa

AI Type ›	Reactive Machines	Rule-Based Systems	Traditional Machine Learning	Deep Learning / Foundation	Multimodal / Agentic AI
Examples ›	IBM Deep Blue chess program's evaluation heuristics.	Expert systems in medical diagnosis (early decision-support), compliance checks, finance (e.g. SAP current), industrial process control.	Spam filtering, customer segmentation, predictive maintenance, image classification with traditional models.	.Image generation (GANs/Diffusion), language models (GPT-style), speech recognition.	Autonomous agents with tool use (Web browsing, IoT control), robotics, decision-support assistants with planning.
Definition ›	Systems that react to present input with no memory or learning from prior interactions.	Systems that combine procedural knowledge with hand-crafted rules; may use reasoning engines but no learned representations.	Systems learn from data using classical ML algorithms (SVM, k-NN, decision trees, logistic regression, clustering.	Deep neural networks with many parameters trained on large corpora; capable of rich representations and generation.	Systems that can plan, reason, and act across modalities and environments, often with external tool use and autonomy.
Strengths ›	Fast, deterministic, low resource usage, predictable.	Explainability, traceability, safety.	Interpretability (e.g., feature importances), well-understood training dynamics.	Strong representation learning, scalability, transfer learning.	Integrated sensing, planning, and action; can operate across domains.
Weaknesses ›	Inflexible, cannot adapt, no learning or improvement.	Maintenance of rules, brittle to edge cases, limited generalisation.	Requires quality labelled data, limited capability with very high-dimensional data.	Data-hungry, compute-intensive, opacity, bias and safety concerns.	Complex failure modes, safety and alignment challenges.
Opportunities ›	Reliable automation in constrained tasks; safety-critical deterministic workflows.	Compliance, safety-critical decision support with auditable decisions.	Routine analytics, automation, baselining for more complex models.	Content generation, automation, multimodal capabilities, pretraining transfer.	Autonomous assistance, dynamic decision support, industrial automation.
Threats ›	Obsolescence, brittle against unmodeled inputs.	Knowledge base decay, scalability concerns.	Data drift, bias, ethical concerns.	Hallucinations, misuse, environmental impact, governance challenges.	Misuse, safety-critical failures, regulatory concerns.
Tech Stack ›	C/C++/Python, rule-based engines, decision trees, finite-state machines.	Rule engines (Drools, CLIPS), Prolog, imperative languages, database backends.	Python, scikit-learn, Jupyter, ML pipelines, data storage (CSV/Parquet), compute (CPU/GPU).	CUDA-enabled GPUs, PyTorch/TensorFlow, distributed training, APIs, containers.	Large-scale DL systems, RL, policy networks, tool use frameworks, orchestration.
Hallucinations ›	Low to none; outputs strictly adhere to programmed rules.	Low to moderate; can produce plausible-sounding but incorrect outputs if rules insufficient.	Low to moderate for classification; may produce overconfident incorrect predictions if biased.	Moderate to high for generative tasks; plausible but incorrect or inconsistent.	High risk in uncertain contexts; can misinterpret instructions or misuse tools.
Error Mitigation ›	Rigorous edge-case testing; hardcoded logic verification.	Formal logic audits; regular manual updates to knowledge bases.	Data quality monitoring; bias detection; cross-validation.	RLHF (Reinforcement Learning From Human Feedback); RAG (Retrieval-Augmented Generation); guardrail layers.	Sandbox environments; Human-in-the-loop (HITL); safety alignment.
Era of Emergence & Dominance ›	1950s – Late 1990s	1980s – Early 2000s	1960s – Present	2012 – Present	2024 – Future
Sustainability Estimate ›	High (low power/compute).	High (runs on standard hardware).	Medium (requires periodic retraining).	Low (high energy/carbon footprint).	Low (constant high compute costs).