Introduction to Artificial Intelligence

Introduction and history
Using AI for scientific writing and development
Responsible AI & EU legal context
Resources and evaluation

Sergey Ignatenko, PhD

LINKÖPING UNIVERSITY

Lecture 1

Introduction and History

2026-04-20

Introduction and History

- Foundations and early AI
- Shift to data-driven AI
- Transformer revolution
- Modern AI systems
- Capabilities, limits, and trends

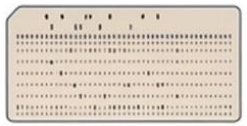
Material is available at: **https://www.itn.liu.se/~siaih22/6fitn80.html**

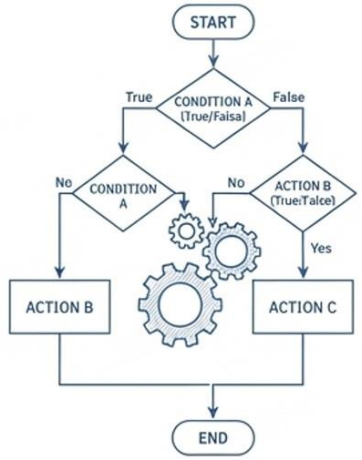
Seminars?

What Is Artificial Intelligence? Definitions Then and Now

Classical definitions (1950s–1980s)

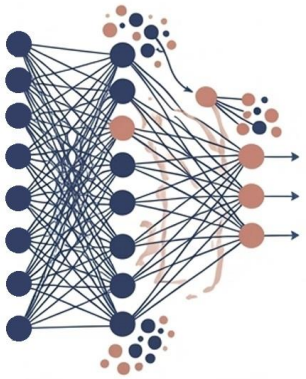
- AI as symbolic reasoning and rule-based problem solving
- Focus on logic, search, and knowledge representation
- Intelligence seen as manipulation of symbols





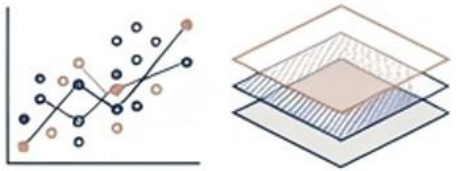
Statistical and Machine Learning era (1990s–2010s)

- Shift toward learning from data
- Emphasis on prediction, classification, and optimization
- Rise of probabilistic and statistical methods



Foundation model era (now)

- AI as scalable representation learning and generative modelling
- Systems trained on massive datasets with self-supervision
- Emergence of general-purpose models (LLMs, multimodal systems)



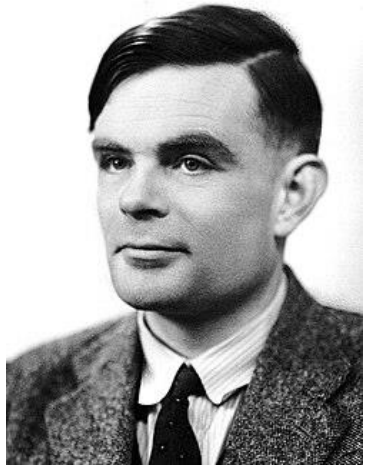
AI
- Machine Learning
- Deep Learning
- Large Language Models

Artificial Intelligence is not a fixed concept, but an evolving field shaped by changing paradigms–from symbolic reasoning to data-driven and large-scale generative systems

Source: S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach (2021)
IBM: The history of AI

Alan Turing and the question “Can Machines Think?”

- In 1950, Alan Turing published “Computing Machinery and Intelligence”
- The imitation game (Turing test)
  - Turing reframed the question into an operational test
  - A machine is considered intelligent if it can engage in conversation indistinguishable from a human
  - Focus shifts from internal thinking → observable behavior
- Key conceptual shift
  - Avoids defining “thinking” directly
  - Replaces philosophical debate with empirical evaluation
  - Establishes a behavioral criterion for intelligence
- Limitations of the Turing test
  - Measures human-like behavior, not true understanding
  - Can be “passed” via imitation, pattern matching
  - Does not assess reasoning transparency, factual correctness
- Lasting impact
  - Established a foundational research question in AI
  - Influenced chatbot development, evaluation of conversational agents
  - Still relevant in discussions of LLMs and generative AI



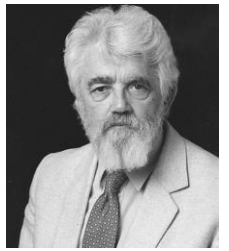
Alan Turing (1912-1954)

Turing transformed the question of machine intelligence into a testable, behavior-based framework, shaping how AI is evaluated to this day

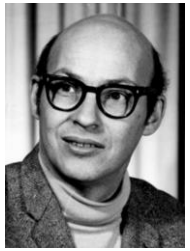
Source: A. Turing, "Computing Machinery and Intelligence". Mind. LIX (253): 433–460 (1950)

The Dartmouth workshop (1956)

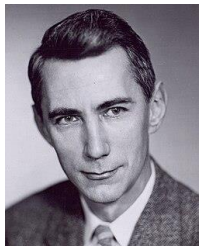
- Held in summer 1956 at Dartmouth College
- Organized by:
  - John McCarthy
  - Marvin Minsky
  - Claude Shannon
  - Nathaniel Rochester
- Foundational proposal
  - Introduced the term **"Artificial Intelligence"**
  - "*Every aspect of learning or intelligence can in principle be so precisely described that a machine can be made to simulate it.*"
- Research vision
  - Symbolic reasoning
  - Language understanding
  - Automated problem solving
- Paradigm established
  - Strong focus on symbolic AI (rule-based systems)
  - Intelligence viewed as manipulation of symbols
  - Laid foundation for decades of research in logic, search, knowledge representation
- Impact on AI as a discipline
  - Established AI as a distinct academic field
  - Created a shared research agenda and community
  - Led to early optimism and major funding initiatives



John McCarthy



Marvin Minsky



Claude Shannon

Nathaniel Rochester

The Dartmouth workshop formalized AI as a field, defining its core ambition: to computationally model intelligence, a goal that continues to evolve today

Sources: AIMA 1.3; History of artificial intelligence (Wikipedia); The history of AI (www.ibm.com)

Symbolic AI and the era of rule-based systems

Symbolic AI: intelligence as rule-based reasoning

- Dominant paradigm in early AI (1950s–1980s)
- Intelligence modeled as manipulation of symbols using formal rules
- Based on logics, search, knowledge representation

Intelligence = applying explicit rules to symbolic representations

Expert systems: practical implementation

- Systems designed to replicate decision-making of human experts
- Built using:
  - if–then rules
  - structured knowledge bases
  - inference engines

Examples:

- MYCIN

Capabilities:

- High accuracy in well-defined, narrow domains
- Transparent and explainable reasoning

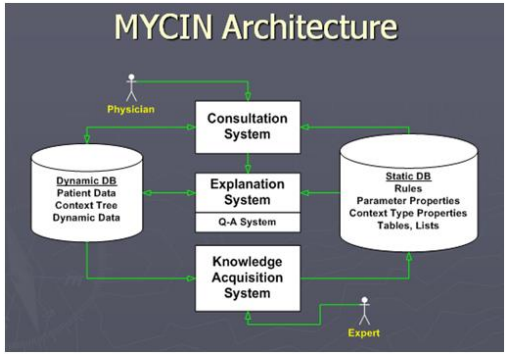
The knowledge engineering paradigm

- AI development focused on encoding expert knowledge manually
- Process involved:
  - interviewing domain experts
  - formalizing rules
  - maintaining knowledge bases

Knowledge acquisition became the main bottleneck

Limitations of rule-based systems

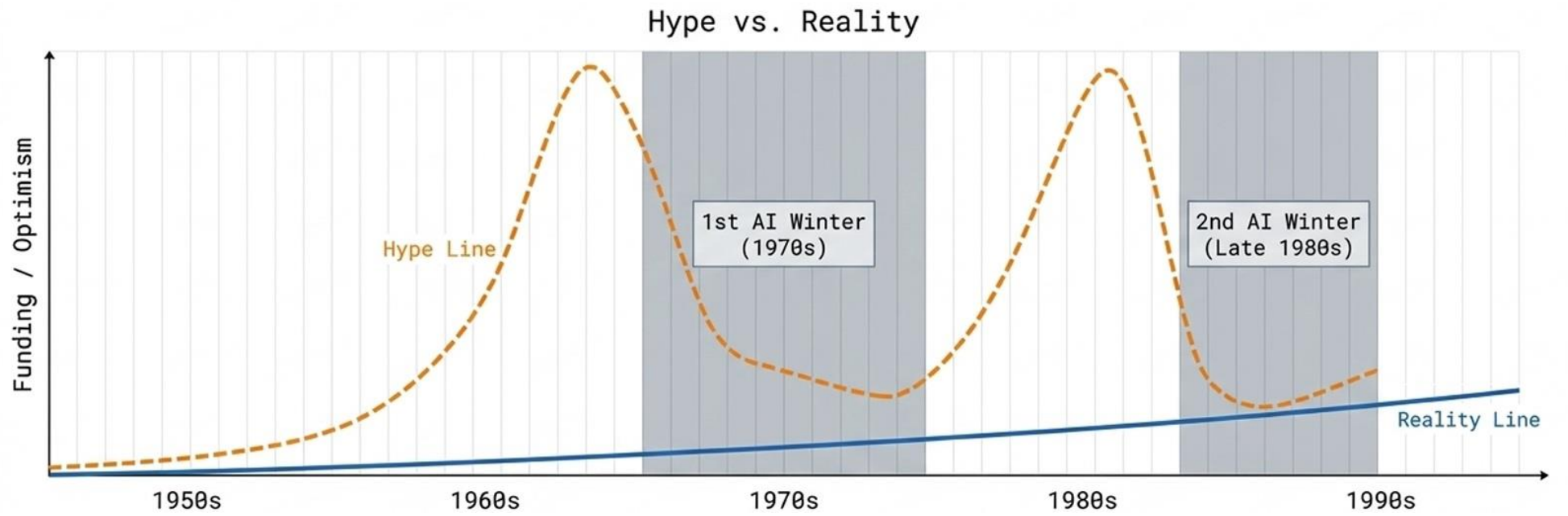
- Difficult to scale and maintain
- Brittle in the face of:
  - uncertainty
  - incomplete or noisy data
- Expensive and time-consuming to update



- Successes of expert systems demonstrated feasibility
- Limitations revealed need for more adaptive approaches

Symbolic AI achieved early success through expert systems and rule-based reasoning, but reliance on manual knowledge encoding limited scalability and adaptability

Rise and fall of early AI "boom" cycles (AI winters)



- Early successes in
  - symbolic reasoning
  - theorem proving
  - simple problem solving
- Strong optimism after Dartmouth Conference

- Lighthill report
- Key issues
  - lack of practical results
  - limited computational power
  - overestimated capabilities

- Rise of expert systems in industry
- Commercial success in narrow domains
- Increased investment and corporate adoption

- Expert systems proved
  - expensive to maintain
  - difficult to scale
- Collapse of AI-focused companies and funding

AI progress has historically been non-linear, shaped by cycles of hype and disappointment, highlighting the importance of rigorous evaluation and realistic expectations

Sources: AIMA 1.3; History of artificial intelligence (Wikipedia); The history of AI (www.ibm.com)

From symbolic AI to machine learning

- Transition from rule-based systems → data-driven approaches
- Move away from manually encoded knowledge toward learning from data

Limitations driving the shift

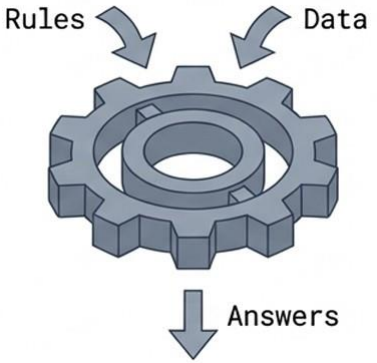
- Symbolic AI struggled with:
  - Scalability to complex, real-world problems
  - Handling uncertainty and noisy data
  - Knowledge engineering bottleneck

Emergence of Machine Learning

- Systems learn from examples rather than rules
- Use statistical methods to:
  - detect patterns
  - make predictions
  - generalize from data

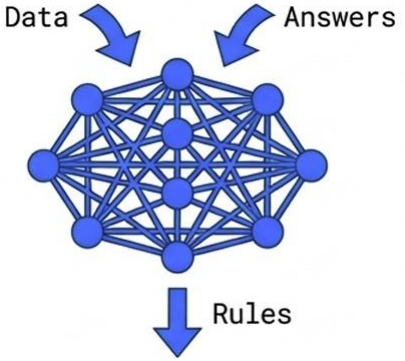
Key idea: Intelligence emerges from data + learning algorithms

Symbolic AI



- Rule-based
- Explicit knowledge
- Deterministic
- Manual design

Machine Learning



- Data-driven
- Learned patterns
- Probabilistic
- Automated learning

The shift from symbolic AI to machine learning replaced handcrafted rules with data-driven learning, enabling AI to scale to complex, real-world problems

Basics of supervised, unsupervised, and reinforcement learning

Supervised Learning
- Learning from labeled data (input → correct output)
- Goal: learn a mapping from inputs to outputs

Typical tasks:
- classification
- regression

Example: predicting labels from annotated datasets

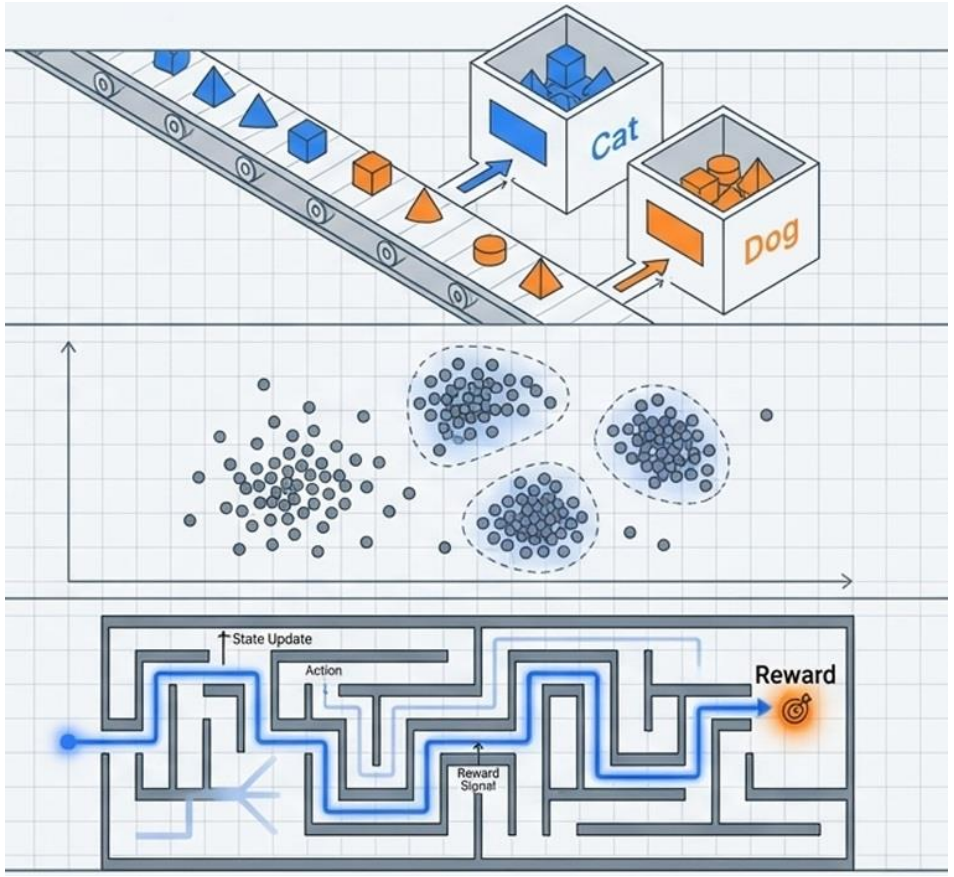
Unsupervised Learning
- Learning from unlabeled data
- Goal: discover structure or patterns in data

Typical tasks:
- clustering
- dimensionality reduction

Example: grouping similar data points without predefined labels

Reinforcement Learning
- Learning through interaction with an environment
- Agent receives rewards or penalties
- Goal: learn a policy that maximizes cumulative reward

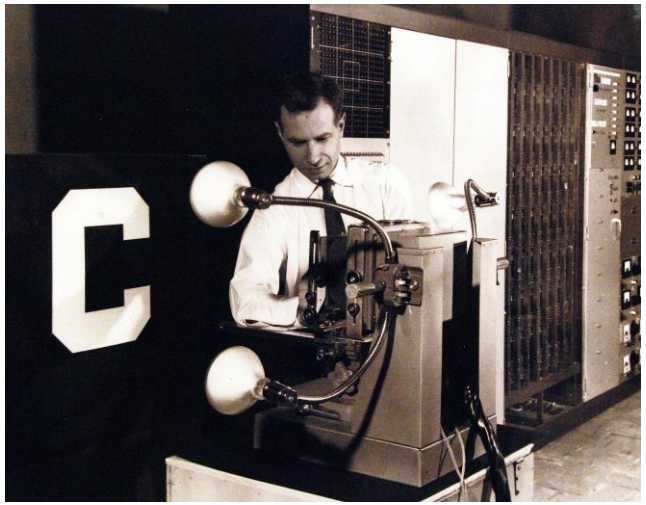
Applications:
- robotics
- game playing



These three paradigms define how AI systems learn: from labels, structure, or interaction, forming the foundation of modern machine learning

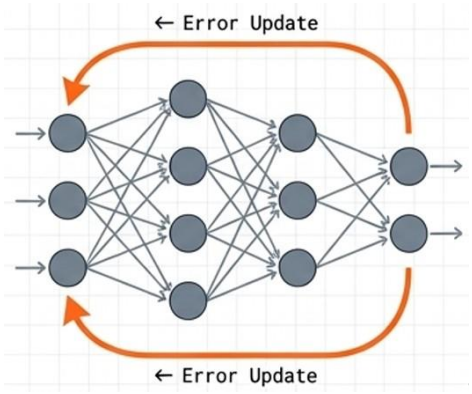
Neural networks: from perceptron to deep learning

Perceptron (1957)



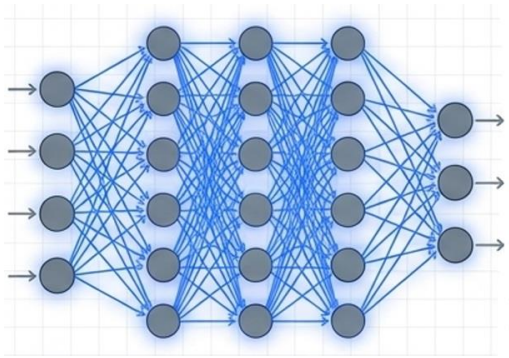
- Introduced by Frank Rosenblatt
- Simple model of a neuron:
  - weighted inputs
  - linear decision boundary
- Key idea: Learn a function that separates data into classes

Backpropagation (mid-1980s)



- Introduction of multi-layer neural networks
- Training enabled by backpropagation algorithm
- Key idea: Stacking layers allows learning of complex, non-linear functions

Deep Learning (2010s)



- Neural networks with many layers ("deep" architectures)
- Enabled by:
  - large datasets
  - increased computational power
  - improved training methods

Applications:

- computer vision
- speech recognition
- natural language processing

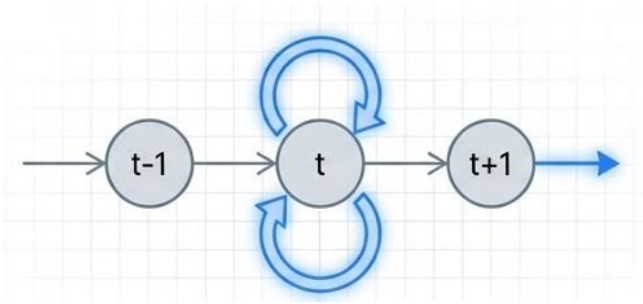
Neural networks evolved from simple linear models to deep architectures capable of learning complex representations, enabling the modern AI revolution

Source: Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain, (1958); Rumelhart, Hinton & Williams, Learning representations by back-propagating errors (1986)

Breakthroughs in computer vision and speech that revived deep learning

ImageNet breakthrough (2012)

- AlexNet achieved major improvement on ImageNet (15% error rate)
- Reduced image classification error significantly
- Key factors:
  - deep convolutional architecture
  - GPU acceleration
  - large-scale labeled dataset (ImageNet, 1.2M+ images)

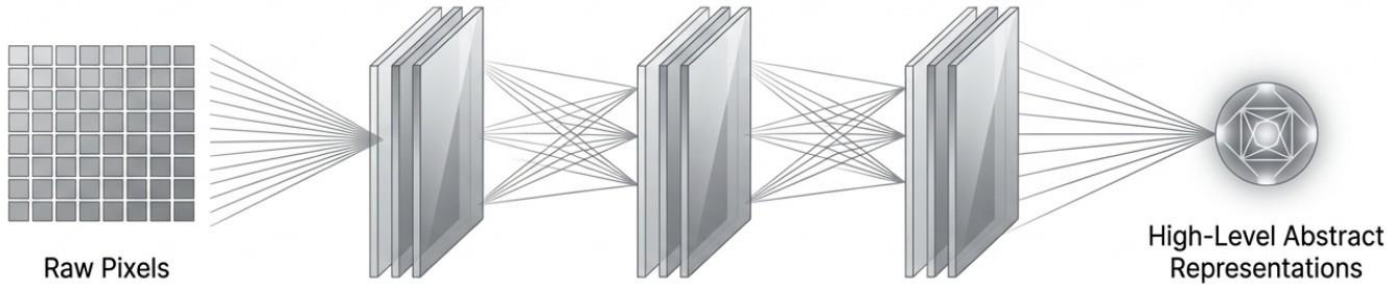


Breakthroughs in speech recognition

- Deep neural networks significantly improved speech recognition accuracy
- Replaced traditional models in major systems
- Key shift: from handcrafted features → learned representations

Why these breakthroughs mattered

- Demonstrated that deep learning:
  - scales with data and compute
  - outperforms traditional methods
- Triggered widespread adoption across AI fields
- Established deep learning as the dominant paradigm

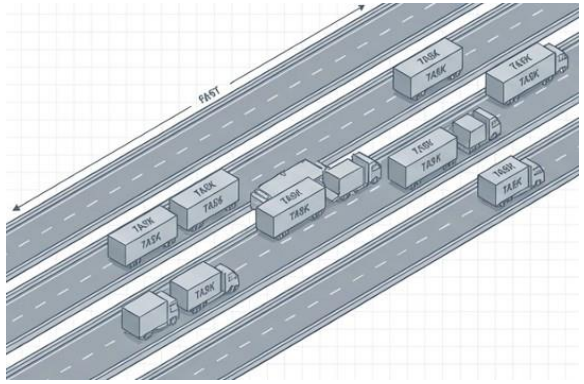


Breakthroughs in vision and speech showed that deep neural networks could outperform traditional methods at scale, triggering the modern resurgence of AI

Hardware revolution: GPUs and scalable computation

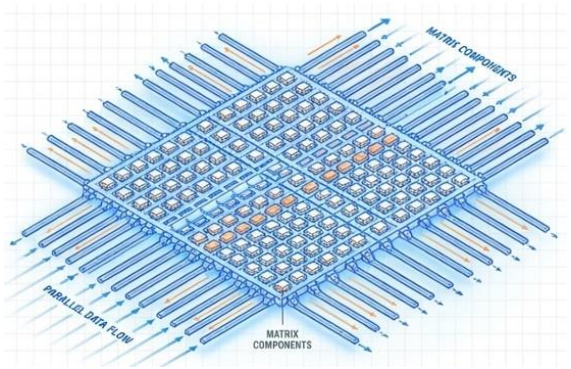
From CPUs to GPUs

- Traditional computing relied on CPUs (sequential processing)
- Shift to GPUs (massively parallel computation)
- Neural networks benefit from parallel operations (matrix multiplications)



Why GPUs enabled modern AI

- Thousands of cores allow efficient training of large models
- Accelerate key operations:
  - linear algebra
  - tensor computations
- Reduced training time from weeks → days or hours



Early adoption in DL

- Use of GPUs was critical in breakthroughs such as AlexNet

Scaling computation

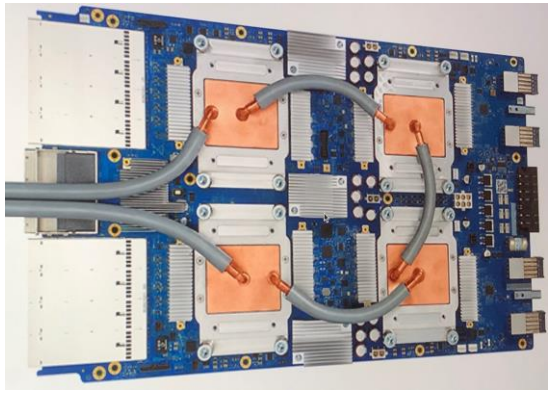
- Growth in:
  - model size (parameters)
  - dataset size
  - training compute
- Emergence of:
  - distributed training
  - cloud computing infrastructure

Specialized AI Hardware

- Development of AI-specific accelerators:
  - TPUs and other custom chips
- Optimized for:
  - deep learning workloads
  - large-scale training

Impact on AI Progress

- Enabled:
  - deep learning revolution
  - large-scale models (LLMs)
- Made AI research dependent on compute resources

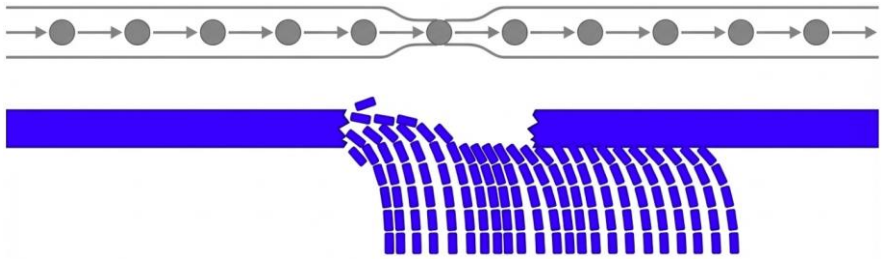


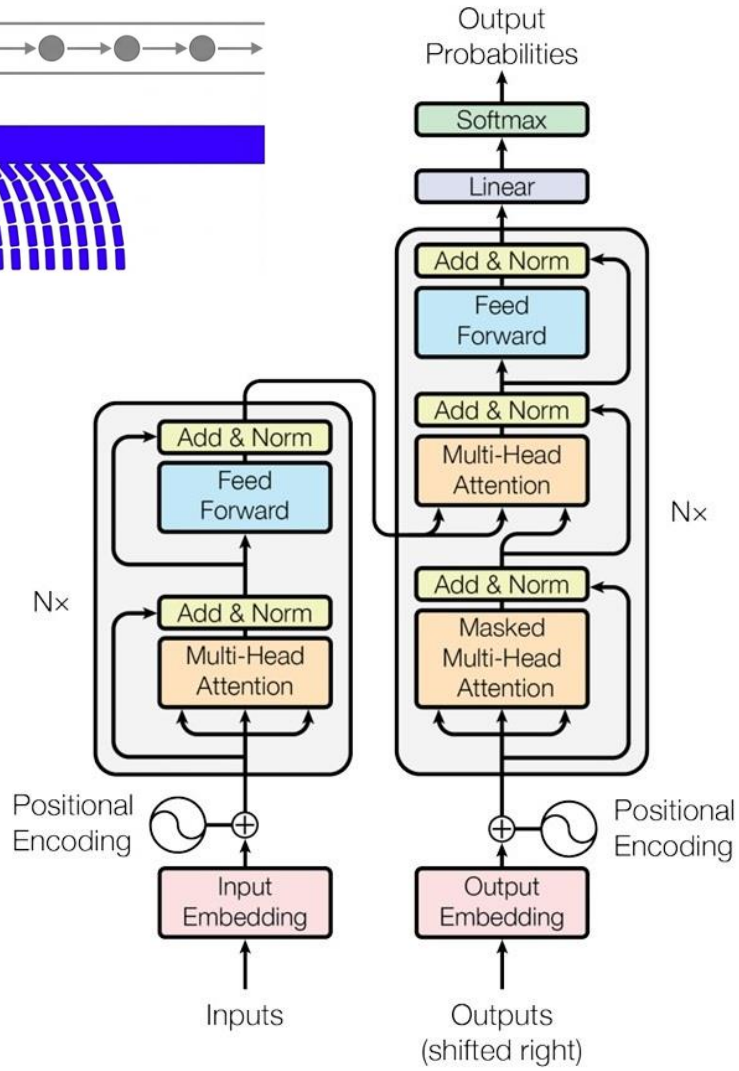
The shift to GPU-based and scalable computation enabled training of large neural networks, making modern AI progress possible but increasingly dependent on compute resources

2017 "Attention is all you need" milestone

Core Innovation: self-attention
- Each token attends to all other tokens in the sequence
- Enables:
  - contextual understanding
  - flexible information flow

Why it was revolutionary
- Eliminated need for:
  - recurrent neural networks (RNNs)
  - sequential processing
- Benefits:
  - Parallel computation (faster training)
  - Better handling of long-range dependencies
  - Improved scalability





2017

Published by Google researchers, introducing "Transformer" architecture

Immediate impact
- Rapid adoption in:
  - natural language processing
  - machine translation
- Led to models such as:
  - BERT
  - GPT

Long-term significance
- Foundation of modern LLMs
- Enabled:
  - scaling to billions of parameters
  - emergence of general-purpose AI systems

The Transformer introduced attention as the core mechanism for sequence modeling, enabling scalable, parallel, and context-aware AI systems that underpin modern LLMs

Source: AIMA 25.4; Vaswani etal., Attention is all you need (2017)

Key concepts: attention mechanism and self-attention

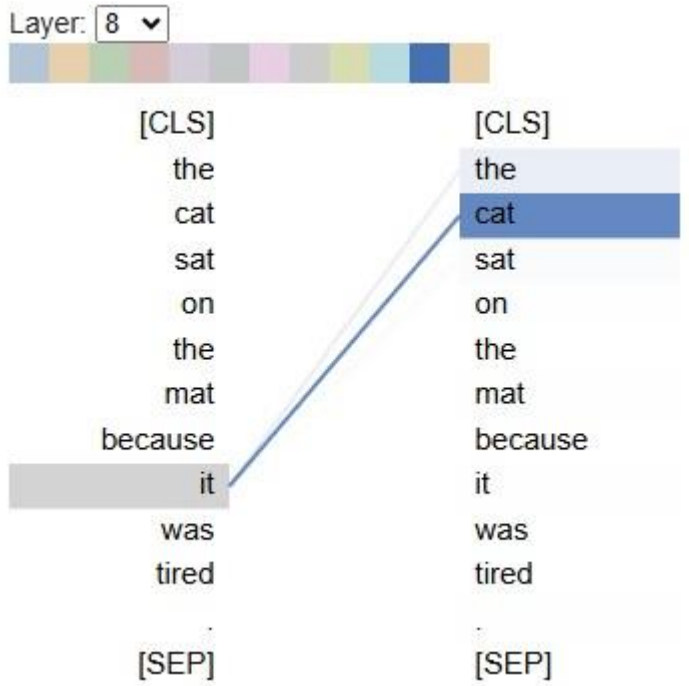
Why Attention?
- In language, meaning depends on context across the entire sequence
- Earlier models struggled with long-range dependencies

→ Models should dynamically focus on relevant parts of the input

Attention Mechanism (General Idea)
- Computes weights over input tokens
- Produces a weighted combination of representations

→ Attention weights indicate how much each token contributes to the current representation

Self-Attention
- Special case where a sequence attends to itself
- Each token:
  - queries all other tokens
  - aggregates relevant information
- Enables:
  - context-aware representations
  - flexible relationships between words
- Advantages
  - Captures long-range dependencies
  - Fully parallelizable
  - Scales efficiently with data and compute
- Interpretation and Limitations
  - Provides insight into information flow
  - But attention weights are:
    - not perfect explanations
    - not always causal indicators



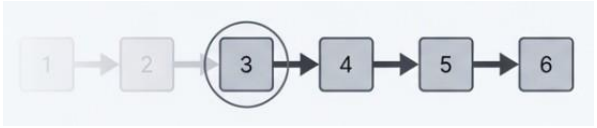
Attention = $\text{softmax}(QK^T)V$

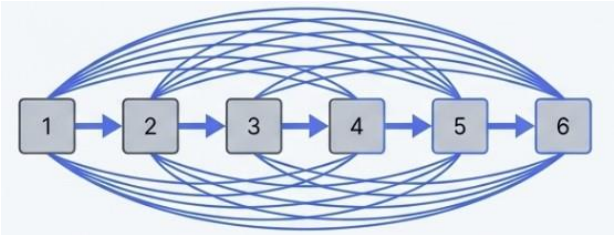
Query (Q): What I am looking for
Key (K): What I offer (index)
Value (V): What I actually contain

Attention allows models to dynamically weight context, while self-attention enables each token to build a context-aware representation from the entire sequence

Source: AIMA 25.4; Vaswani etal., Attention is all you need (2017)

Why transformers replaced RNNs





Parallelization

- Entire sequence processed at once
- Enables efficient use of GPU hardware

Long-range dependencies

- Any token can attend to any other token
- No need to propagate information step-by-step

Scalability

- Architecture scales effectively with:
  - data
  - model size
  - compute

Simpler training dynamics

- Avoids many issues of:
  - gradient instability
  - memory bottlenecks in long sequences

Conceptual shift

- From sequential processing → global context modelling
- From time-dependent memory → attention-based representation

Transformers replaced RNNs because they enable parallel computation, better handling of long-range dependencies, and scalable training, making them the foundation of modern AI systems

Sources: AIMA 25.2, 25.4; Hochreiter & Schmidhuber, LSTM paper (1997)

Visualization tools: BertViz and TensorFlow attention maps

Motivation: Making Attention Interpretable
- Transformer models are often seen as “black boxes”
- Visualization helps to:
  - understand internal behavior
  - inspect attention patterns
  - support debugging and research

BertViz

- Interactive tool for visualizing attention in transformer models
- Supports models like:
  - BERT
  - GPT-style architectures

Features:
- visualizes attention heads and layers
- shows token-to-token attention weights
- enables exploration of linguistic patterns

TensorFlow Attention Visualizations

- Provides tools for inspecting:
  - attention weights
  - model outputs
- Often used in:
  - tutorials
  - research experiments

Features:
- attention heatmaps
- integration with model training pipelines

What these tools reveal
- Different attention heads capture:
  - syntax (e.g., subject–verb relations)
  - coreference (e.g., pronouns)
  - positional patterns
- Help identify model biases, failure cases

Limitations of visualization
- Attention ≠ full explanation of model behavior
- Visual patterns can be:
  - misleading
  - difficult to interpret causally

Visualization tools like BertViz and TensorFlow attention maps help demystify transformer models by exposing attention patterns, but they provide insight into behavior–not complete explanations

Sources: Vig, A multiscale visualization of attention in the transformer model (2019)

Scaling laws and the rise of foundation models

Empirical scaling laws
- Performance of neural networks improves predictably with more data, larger models (parameters), increased compute
- Key idea: Loss decreases as a power-law function of scale

Compute as a driver of progress
- Advances in hardware + distributed training enabled: training of very large models
- Shift from algorithmic innovation → scaling paradigm

Emergence of Foundation Models
- Large models trained on broad, diverse datasets
- Can be adapted to many tasks with minimal changes

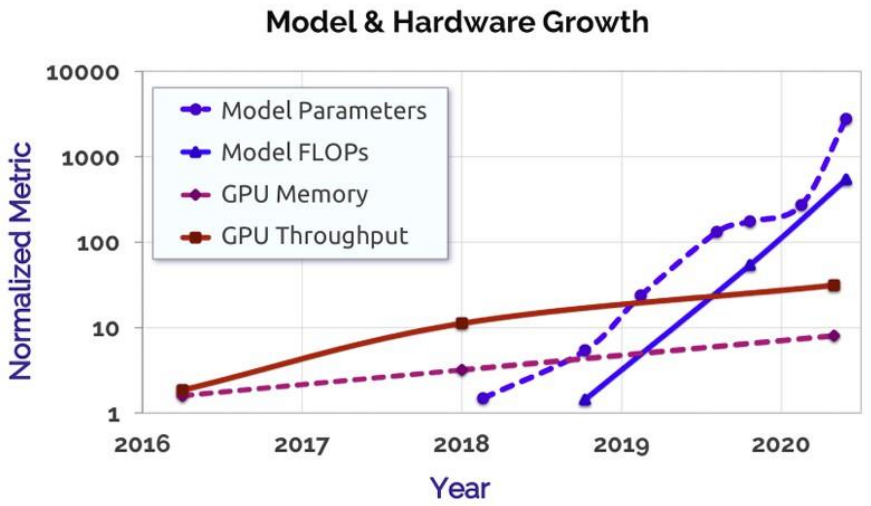
Key properties: general-purpose, transferable across domains, support zero-shot and few-shot learning

From Task-Specific to General Models
- Earlier AI: one model per task
- Foundation models: one model, many tasks
- Examples: GPT, BERT

Emergent Abilities

Implications for AI research: shift toward prompt-based interaction

Challenges: high computational cost, environmental impact, bias and data quality issues



Scaling laws revealed that increasing data, model size, and compute leads to predictable performance gains, enabling the rise of foundation models as general-purpose AI systems

Sources: Bommasani etal. On the Opportunities and Risks of Foundation Models (2021); Kaplan etal, Scaling laws for natural language models, 2001.08361v1

Trends: GPT, BERT, T5, LLaMA, Claude

From specialized to general-purpose models
- models differ in training objectives, architectures, use cases

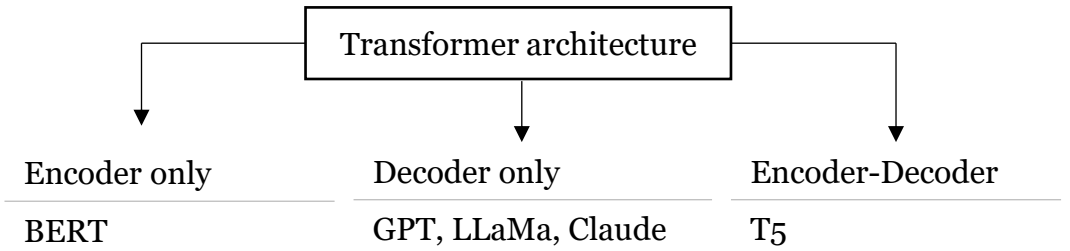
GPT (generative models)
- autoregressive: predict next token
- strong in text generation, reasoning, conversational tasks

BERT (bidirectional understanding)
- masked language modelling
- deep bidirectional context
- strong in classification, information extraction

T5 (unified framework)
- all tasks framed as text-to-text
- flexible and general-purpose

LLaMA (efficient open models)
- focus on efficiency, research accessibility
- strong performance with fewer parameters

Claude (aligned AI systems)
- emphasis on safety, alignment, helpfulness
- uses techniques like constitutional AI

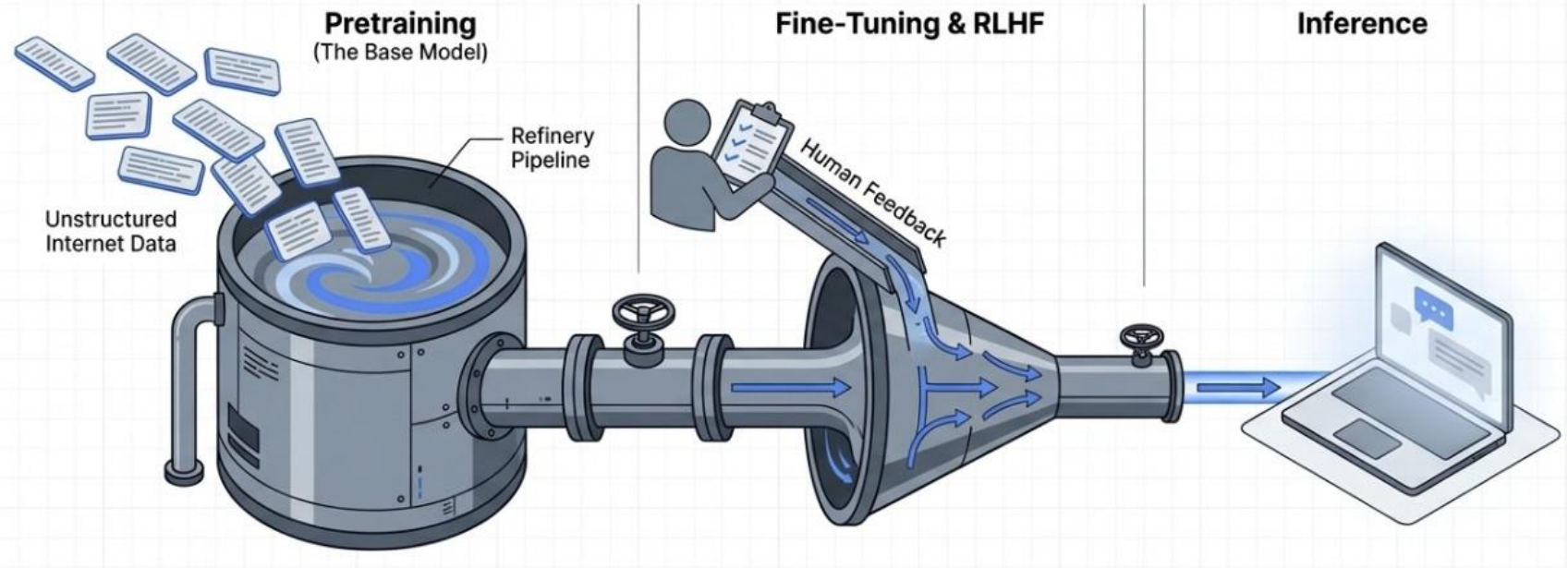


Key trends across models
- scaling to larger models and datasets
- shift toward instruction-following systems
- increasing focus on alignment, safety, usability

Modern LLMs represent different design choices around generation, understanding, efficiency, and alignment, reflecting the rapid evolution toward general-purpose AI systems

Sources: Brown etal, Language models are few-shot learners, 2005.14165v4; Delvin etal, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 1810.04805v2; Raffel etal, Exploring the limits of transfer learning with a unified text-to-text transformer, 1910.10683v4; Touvron etal, LLaMA Open and efficient foundation language models, 2302.13971v1

Pretraining, fine-tuning, and inference



- Train model on large-scale, general data
- Most capabilities emerge during pretraining
- Objective: learn general representations, capture structure of language/data
- Key characteristics: self-supervised learning (e.g., next-token prediction), computationally intensive
  - High cost

- Adapt pretrained model to specific tasks or domains
- Methods: supervised fine-tuning (labeled data), instruction tuning, Reinforcement Learning from Human Feedback (RLHF)
- Goal: align model behavior with task requirements or human preferences
  - Risk of overfitting

- Stage where model is applied to new inputs
- No learning occurs (parameters fixed)
- Includes: prompting, generation, decision-making
  - Inference efficiency and latency constraints

Modern AI systems are built through large-scale pretraining, adapted via fine-tuning, and deployed during inference–separating learning from usage

Source: AIMA 25.5

From traditional ML to LLMs: key differences

	Traditional machine learning	LLMs / Foundation models
Architecture	Task-specific algorithms	Unified, general-purpose transformer
Training	Trained from scratch for single, narrow task	Pretrained once on vast internet data, then adapted for many tasks
Data requirements	Requires 1000s of perfectly labeled examples to function	Zero-shot or few-shot learners; can infer intent from a simple text prompt
Capabilities	Only does exactly what it was trained to do	Exhibit emergent abilities, like translation, logical reasoning

The shift from traditional ML to LLMs represents a move from task-specific, feature-engineered systems to large, general-purpose models capable of flexible, multi-task behavior

Emergence, generative AI, and multimodality

From predictive to generative models
- traditional models: predict labels or values
- generative models: produce new content (text, images, audio)

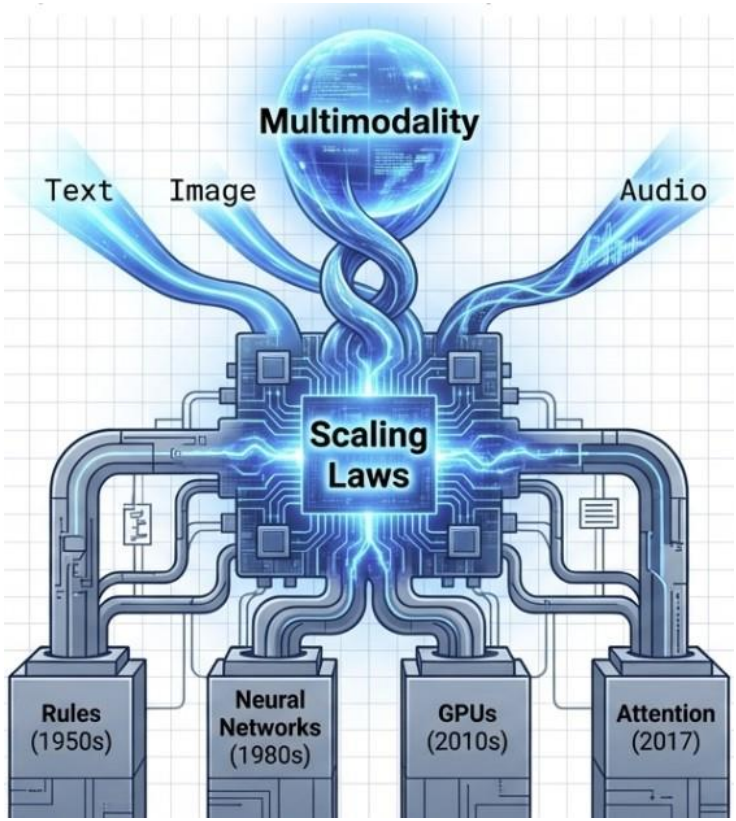
Foundation models and generative AI
- Large models trained on web-scale, diverse data
- Enable: text generation, reasoning-like behavior, cross-task generalization

Emergent abilities in large models
- Capabilities appear **only at scale**: in-context learning, reasoning-like behavior, few-shot generalization

Why generative AI accelerated after 2022
- Convergence of:
  - Transformer architecture (2017)
  - large-scale compute
  - massive datasets
- Rapid improvements in: quality, usability, accessibility

Multimodality
- Modern models process multiple data types: text, images, audio, video
- A single foundation model can now parse image, reason about it logically and generate a spoken response

Alignment techniques
- Improve usefulness and safety via instruction tuning, RLHF



The combination of scaling, transformers, and massive data led to generative AI systems with emergent abilities and multimodal capabilities, marking a fundamental shift in AI

Sources: Bommasani etal, On the opportunities and risks of foundation models (2021); Kaplan etal, Scaling laws for natural language models, 2001.08361v1; Ouyang etal, Training language models to follow instructions with human feedback, 2203.02155v1

Limitations and the new role of researcher

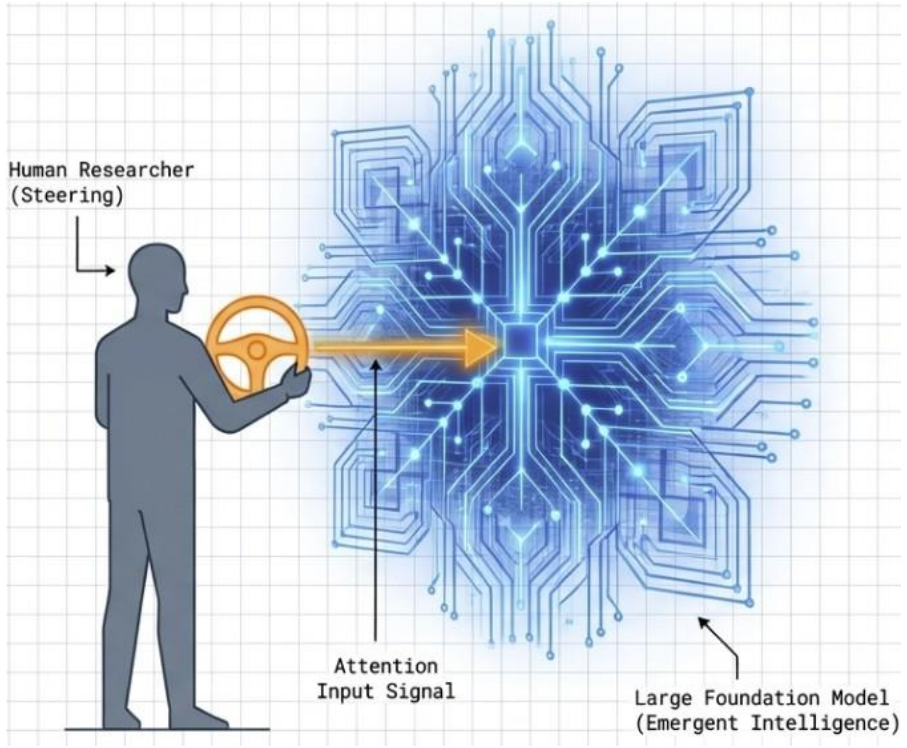
Key Limitations of modern AI systems
- Hallucinations: generate plausible but incorrect information
- Bias and fairness issues: reflect biases in training data
- Lack of true understanding: pattern recognition ≠ reasoning
- Uncertainty and overconfidence

Opacity and interpretability challenges
- Models are often black boxes
- Difficult to: explain decisions, trace reasoning
- Limiting usage in high-stakes domains

Data and dependency risks
- Dependence on: large-scale web data, proprietary datasets
- Issues: data quality and contamination, reproducibility challenges

Evaluation challenges
- Hard to measure: reasoning, factual accuracy, robustness
- Benchmarks may not reflect real-world performance

The new role of researcher
- critical evaluator of outputs
- designer of prompts and workflows
- architect of AI-augmented systems



AI does not replace researchers — it changes how research is conducted

Modern AI systems are powerful but imperfect, requiring researchers to act as critical architects who validate, guide, and augment AI-driven workflows

Sources: Bender etal, On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? (2021); Bommasani etal, On the opportunities and risks of foundation models (2021);