



COLUMBIA
UNIVERSITY

AI engineering apps

IEOR4574E001

Fall 2025



Nice to meet you

- | Co-founder and CEO at **Bauplan**.
- | Previously co-founder at Tooso (acquired by Coveo [TSX:CVO](#)).
- | VP AI at Coveo from scale-up to IPO.
- | Postdoc in computational stuff.

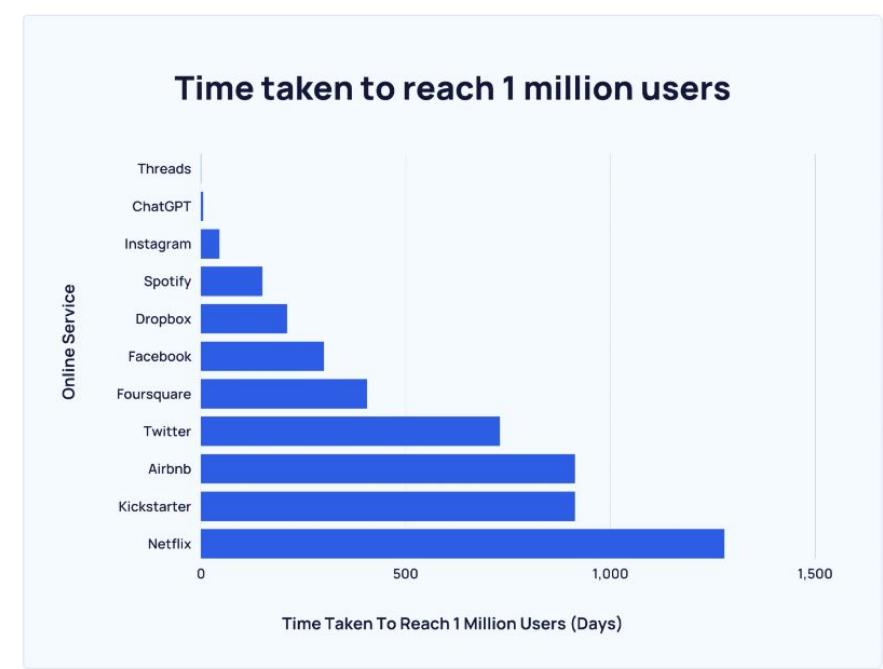


The golden era of AI



It's a fantastic moment to be in AI

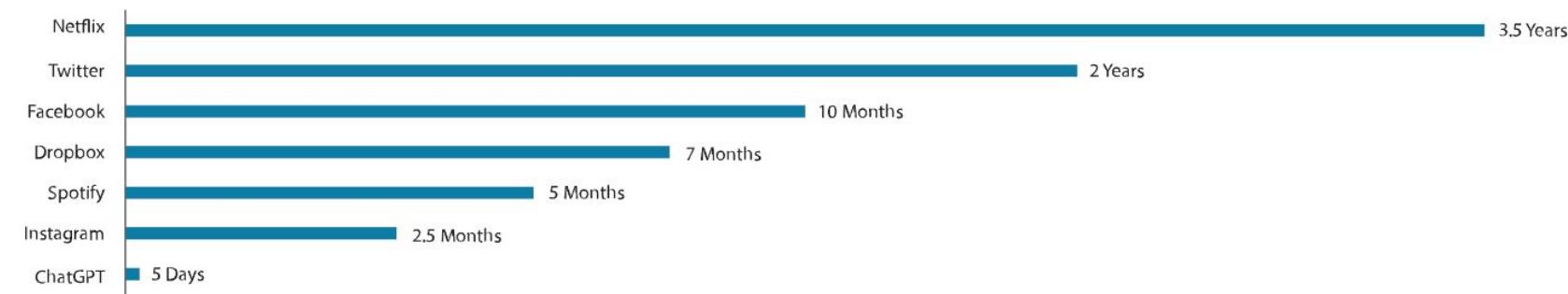
- ChatGPT usage: In April 2025, ChatGPT had approximately 800 million weekly users, with users sending 2.5 billion prompts per day by mid-2025.
- Generative AI impact: It's expected to generate US \$1.3 trillion in global economic impact annually by 2030



It's a fantastic moment to be in AI

- AI market growth
Expected to rise from US \$294 billion (2025) to US \$1.77 trillion by 2032, with a 29.2% CAGR.
- AI market forecast
Another credible estimate projects AI growing from US \$197 billion (2023) to around US \$1.81 trillion by 2030, with a 36.6% CAGR.

Time taken for reaching 1Bn mark



Source: Company announcements via Business Insider/LinkedIn

It's a fantastic moment to be in AI

- GDP boost potential: AI could increase global GDP by up to 14% by 2030, representing an additional ~US \$15.7 trillion.

3x

Higher growth in revenue per worker in industries more exposed to AI

100%

Of industries are increasing AI usage including industries less obviously exposed to AI such as mining and agriculture

66%

Faster skill change in AI-exposed jobs up from 25% last year. Change is fastest in automatable jobs

Applications

- **Code generation & IDE copilots (B2B)**
GitHub Copilot: 20M all-time users; used by 90% of the Fortune 100.
[TechCrunch](#)
Cursor (AI IDE): raised at ~\$10B valuation; company says >\$500M ARR.
[Reuters](#).
- **General-purpose chat assistants (B2C)**
ChatGPT: ~800M weekly users (Oct 2025) and ~\$12B annualized revenue run-rate (mid-2025). [Business Insider](#)
Google Gemini app: 650M monthly active users (Q3 2025). [Business Insider](#)
Perplexity: ~22M MAU and ~\$80M ARR in 2025. [businessofapps.com](#)
- **Image generation & editing**
Midjourney: ~21M community members and ~\$500M ARR by mid-2025. [CB Insights](#).
- **Video generation**
Runway: targeting ~\$300M annualized revenue in 2025. [TechCrunch](#)
OpenAI Sora app: >1M downloads within five days of launch (Sep/Oct 2025).
[The Verge](#)
- **Enterprise productivity.**
Microsoft 365 Copilot: used by “nearly 70%” of the Fortune 500; sold at ~\$30 per user/month. [Source](#)
- **Customer support AI agents**
Zendesk’s new AI Agent: vendor claims up to **80%** issue resolution.
[TechCrunch](#)
- **Design/marketing content creation**
Canva (with Magic Studio/AI tools): ~170M monthly active users.

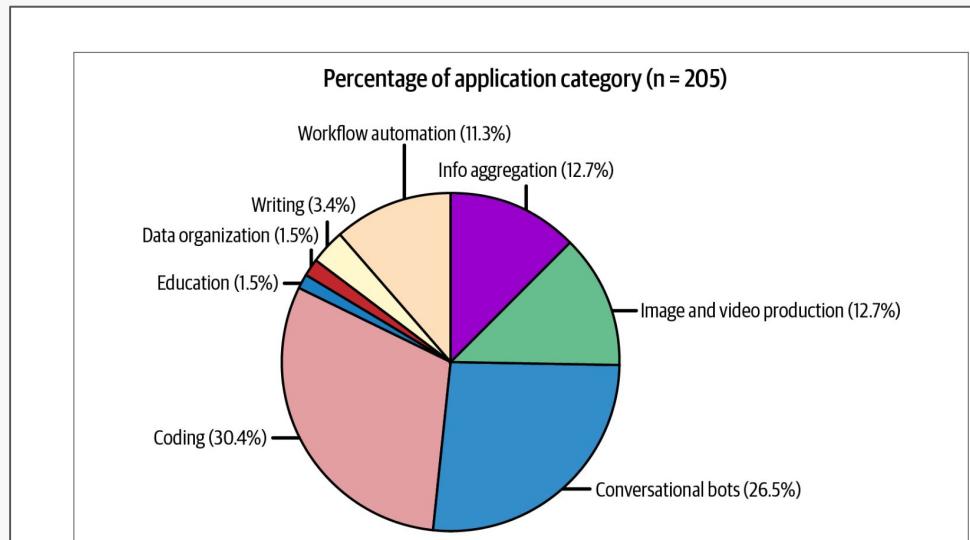


Figure 1-7. Distribution of use cases in the 205 open source repositories on GitHub.

Image from Huyen 2025

The future is... we don't really know, actually

- AI 2027 projects explosive AI progress, culminating in AGI by late 2027:
 - Mid-2025: “Stumbling agents” begin performing simple digital tasks (scheduling, web automation)
 - 2026-2027: Emergence of “Neuralese” (latent reasoning language), enabling rapid self-improvement and shared knowledge across AI agents
 - Late 2027: Deployment of superhuman agents (e.g., “Agent-3-mini”) widely disrupts labor markets; AGI capabilities surpass human cognitive reach
- However, [AI researchers estimate](#) only a 10% chance of machines outperforming humans in all tasks by 2027, and 50% by 2047.

AI 2027¹

Daniel Kokotajlo, Scott Alexander, Thomas Larsen, Eli Lifland, Romeo Dean

We predict that the impact of superhuman AI over the next decade will be enormous, exceeding that of the Industrial Revolution.

We wrote a scenario that represents our best guess about what that might look like. It's informed by trend extrapolations, wargames, expert feedback, experience at OpenAI, and previous forecasting successes.²

[What is this? | How did we write it? | Why is it valuable? | Who are we?]

Published April 3rd 2025 | PDF | Listen | Watch

Mid

The w

Advertis

you can

my bu

with yo

more a

widesp

Meanw

beginn

Unreliable Agent

Apr 2025 Dec 2024

Hacking Coding Robotics Forecasting

AI Concepts

Bioweapons Compute

2,000 Unreliable Agent copies thinking at 8x human speed

Approval Revenue Valuation Importance Datacenters Timeline

The New York Times

OPINION GUEST ESSAY

Silicon Valley Is Drifting Out of Touch With the Rest of America

Aug. 19, 2025



Large Language Models

- LLMs = very large deep learning models that are pre-trained on vast amounts of data to predict next tokens in a sequence.
- Based on transformer architecture (Vaswani et al., 2017).
- Capabilities: reasoning, summarization, translation, code generation, tool use.
- Examples: GPT-4, Claude, Gemini, LLaMA.



Transformers introduced
"Attention is All You Need"

17

BERT
(Bidirectional Transformer)

18

19

20

- GPT-2 (1.5B parameters)
- RoBERTa

- CLIP & DALL-E
- Codex
- Google Pathways

21

- GPT-3 (175B parameters)
- Hugging Face's Transformers
- Google T5

22

- DALL-E 2
- Stable Diffusion
- ChatGPT
- Open-source LLMs (OPT, BLOOM)

23

- GPT-4 (multimodal)
- Claude
- LLaMA
- AWS Bedrock
- Runway's Gen-1 & Gen-2

24-25

Jan 25

- DeepSeek R1
- OpenAI & SoftBank's \$500B AI project

Feb 25

- Nvidia Blackwell GPUs
- Meta AI chatbot
- AI Action Summit

Mar 25

- GM appoints Chief AI Officer

Dec 24

- DeepSeek V3 matches GPT-4o & Claude 3.5

Feb 24

- Claude 3
- Sora
- Gemini 1.5

Mar 24

- DeepMind SIMA

Oct 24

- Nobel Prize for AlphaFold

Jun 24

- Apple Intelligence

The rise of AI engineering

- Shift from AI research to AI engineering: building reliable, scalable applications
- Focus areas: prompt/data pipelines, cloud deployment, monitoring (cost, latency, safety)
- AI engineering is to ML research what software engineering is to computer science

From LMs to LLMs



Language models are completion machines

My favorite book is _?

tokens

Language models are completion machines

My favorite book is _?

tokens

Moby Dick	4.1%
War and Peace	3.9%
the one	1.4%
written	2.2%



Pick one of these, but with
a bit of randomness

argmax ----- sampling



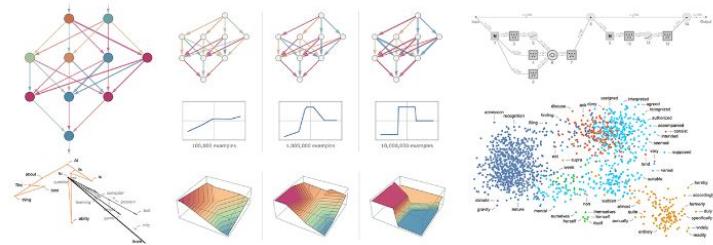
temperature

Language models are completion machines

- Both traditional LMs and today's LLMs do the same core thing: **predict the next token given context.**
- More specifically: decode by **sampling** or **taking argmax** to complete the sequence.

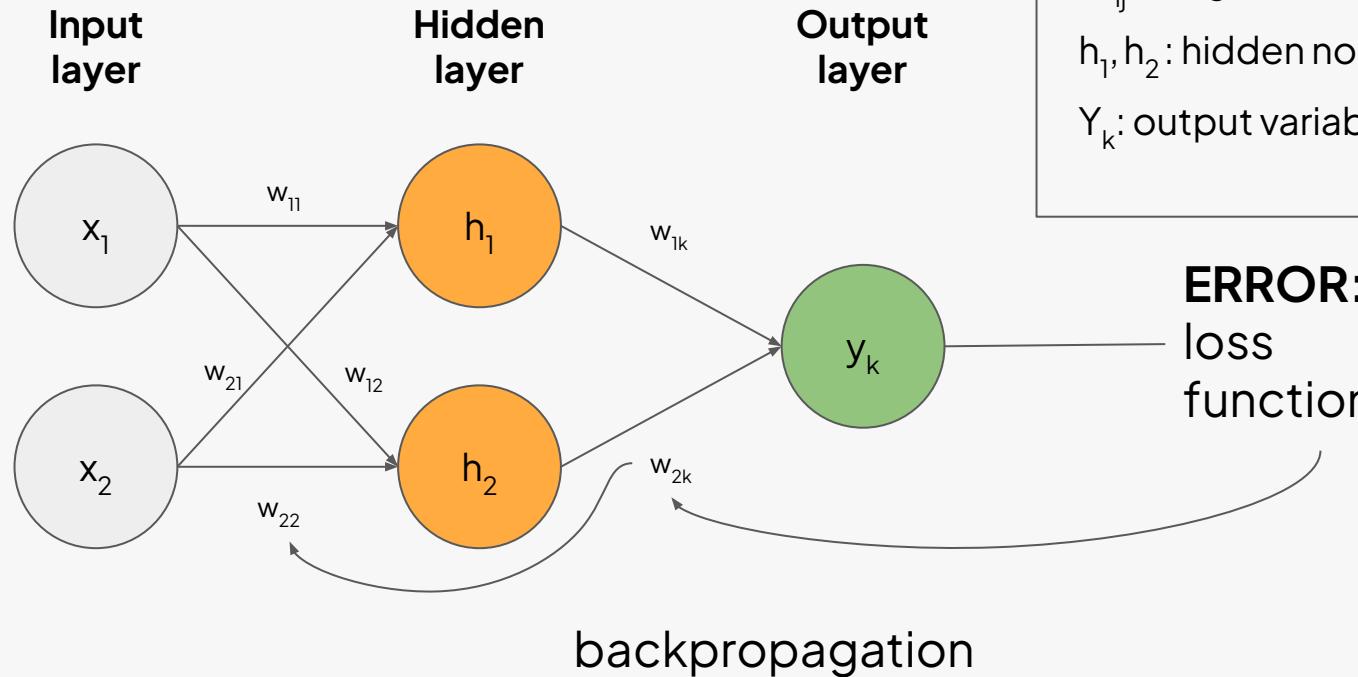
What Is ChatGPT Doing ... and Why Does It Work?

February 14, 2023



It's Just Adding One Word at a Time

Neural networks



x_1, x_2 : input data features

W_{ij} : weights of the network

h_1, h_2 : hidden nodes of the network

Y_k : output variable

ERROR:
loss
function

Find the best parameters:

- We want the parameters that maximize the likelihood of the training data.
- Equivalent to “make the model assign high probability to real text.”

Why log? Because probabilities are small and multiplying them becomes unstable. Log turns products into sums, making optimization tractable.

$$\text{LM/LLM objective: } \max_{\theta} \sum_t \log p_{\theta}(x_t | x_{<t})$$

The parameters of the model:

- Billions of numbers (weights) in a neural network.
- Training = adjusting theta to make predictions better.

Sum over all positions in the text sequence:

- Every word/token in the training data contributes to the training objective.
- The model learns from predicting each next token correctly.

This is the heart of the LM: predict the probability that the model with parameters θ assigns to token x_t given the context – that is the token that came before $x_{<t}$.



From language models to large language models

- **Masked LM** (BERT, 2018): predicts hidden tokens. Great for understanding and classification tasks.
- **Autoregressive LM** (GPT, 2018+): predicts the next token. Better for generation.
- Today's LLMs mostly **autoregressive** → fluent, flexible outputs.

Figure 1-2 shows these two types of language models.

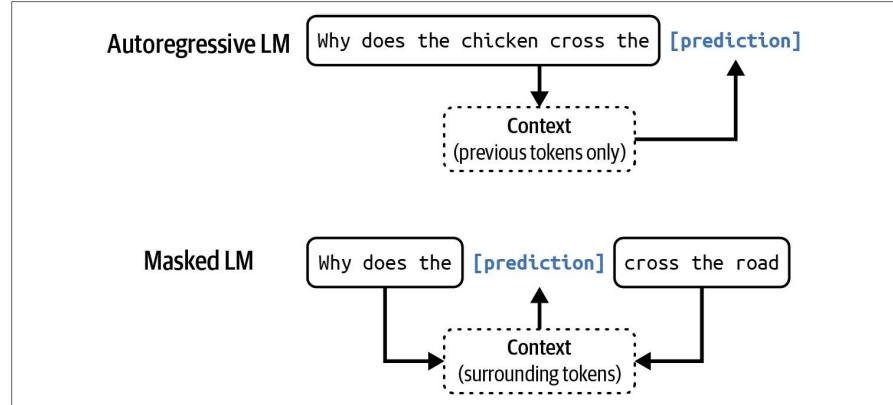


Figure 1-2. Autoregressive language model and masked language model.



So what's different with LLMs?

- **Scale**

Much larger parameters, data, and compute → richer learned distributions, emergent capabilities.

- **Architecture**

From n-grams and RNN/LSTM to transformers with self-attention and long context windows.

- **Training recipe**

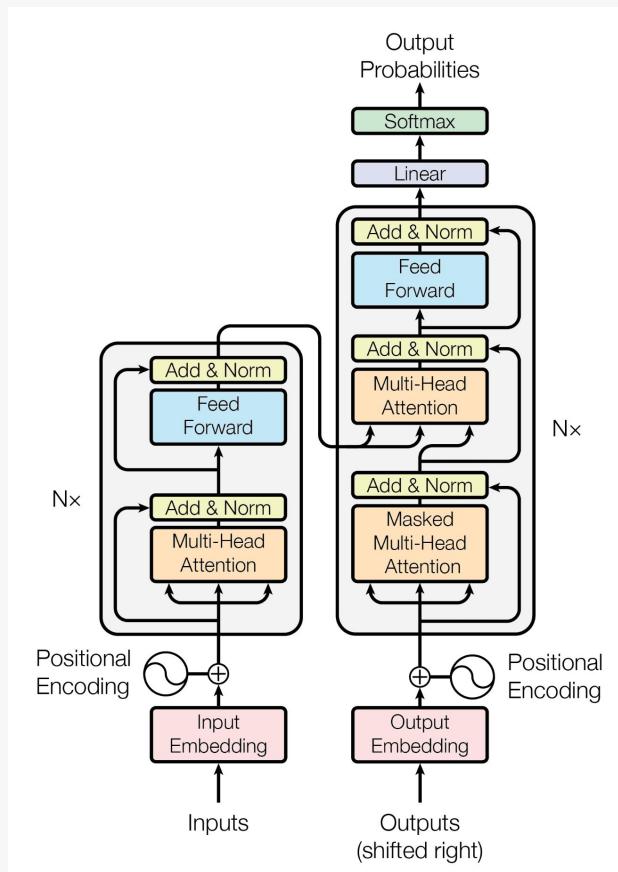
Self-supervised pretraining + instruction tuning + preference optimization (RLHF/DPO) → chat/task behavior.





From language models to large language models

- **N-grams (1990s–2000s):** statistical models with limited context.
- **RNNs & LSTMs (2010s):** introduced memory, but struggled with long sequences.
- Breakthrough: **transformer architecture (2017)** → attention mechanism scales to long context.



Training recipe: self-supervision

- **Self-supervised learning:** training on massive unlabeled corpora.
- **Cost is exponentially lower:** no human labels → unprecedented scaling becomes possible.
- Data = internet text, code, images.
- Changed how we think about datasets:
quantity and diversity > handcrafted labels.

Input: “The cat sat on the [MASK]”

Label: “mat” (comes directly from the original sentence).

<BOS>	the
<BOS>, the,	cat
<BOS>, the, cat	sat
<BOS>, the, cat, sat	on
<BOS>, the, cat, sat, on	the
<BOS>, the, cat, sat, on, the	mat

Training set

Is all we need just a bigger boat?

- **Size does matter in terms of data AND model**
 - Rapid growth in size: From 117M parameters (GPT-1, 2018) to 175B (GPT-3, 2020) to estimates of 1.8T (GPT-4, 2023).
 - Scaling laws (Kaplan et al., 2020): Model performance improves predictably with more data, parameters, and compute.
 - Emergent abilities (Wei et al., 2022): Beyond certain thresholds, qualitatively new capabilities appear (e.g., reasoning, coding).
- **But scaling is costly:**
 - Compute- and data-optimal scaling (Chinchilla, 2022) shows we need ~20 tokens per parameter to train efficiently.
 - Bottlenecks ahead: data scarcity and energy consumption.
 - **Industry reality:** Small, efficient open-source models (Mistral, LLaMA) are gaining adoption because they are cheaper to train and run, even if they don't always hit state-of-the-art accuracy.



Large language models

Traditional Language Models	Large Language Models (LLMs)
Trained on small datasets with limited compute → narrow capabilities	Trained on massive datasets with huge compute → qualitatively new capabilities
Task-specific (e.g., translation, classification, spam detection, etc.)	General-purpose (translation, reasoning, summarization, coding, multi-domain)
Scale = incremental improvements	Scale = emergent abilities (reasoning, tool use, code generation)
Each model trained for a single task	Foundation models: one model adapted to many tasks

Take a look at:

Wei et al., *Emergent Abilities of Large Language Models* (2022) arXiv

Kaplan et al., *Scaling Laws for Neural Language Models* (2020) arXiv

Multi-modal models

- Foundation models encapsulate knowledge about the world
- And data comes in several modalities
- Historically we don't do a lot with unstructured data
- Foundation models will increasingly be multi-modal

An example: from CLIP to Fashion CLIP

Download model:

<https://huggingface.co/patrickjohnc/yh/fashion-clip>

scientific reports

Explore content ▾ About the journal ▾ Publish with us ▾

nature > scientific reports > articles > article

Article | Open access | Published: 08 November 2022

Contrastive language and vision learning of general fashion concepts

Patrick John Chia, Giuseppe Attanasio, Federico Bianchi, Silvia Terragni, Ana Rita Magalhães, Diogo Goncalves, Ciro Greco & Jacopo Tagliabue

Scientific Reports 12, Article number: 18958 (2022) | [Cite this article](#)

21k Accesses | 32 Citations | 14 Altmetric | [Metrics](#)

A Publisher Correction to this article was published on 23 January 2023

This article has been [updated](#)

	Image	Textual Data
Farfetch		Short Description: Cropped knit cardigan Gender: Women Brand: M Missoni Highlights: light brown/multicolour; mix print; ribbed-knit edge; V-neck; three-quarter length sleeves; ribbed hem; cropped
		Product Name: Nike Women As Element Ja Pink
		Category: Topwear
		Sub-category: T-Shirts
KAGL		Category: T-shirt / Top
F-MNIST		Product Description: Breton Stripe Henley Category: Henley Attributes: Long sleeve; Sleeve
DEEP		

An example: from CLIP to Fashion CLIP

Download model:

<https://huggingface.co/patrickjohncyh/fashion-clip>

scientific reports

Explore content ▾ About the journal ▾ Publish with us ▾

[nature](#) > [scientific reports](#) > [articles](#) > [article](#)

Article | [Open access](#) | Published: 08 November 2022

Contrastive language and vision learning of general fashion concepts

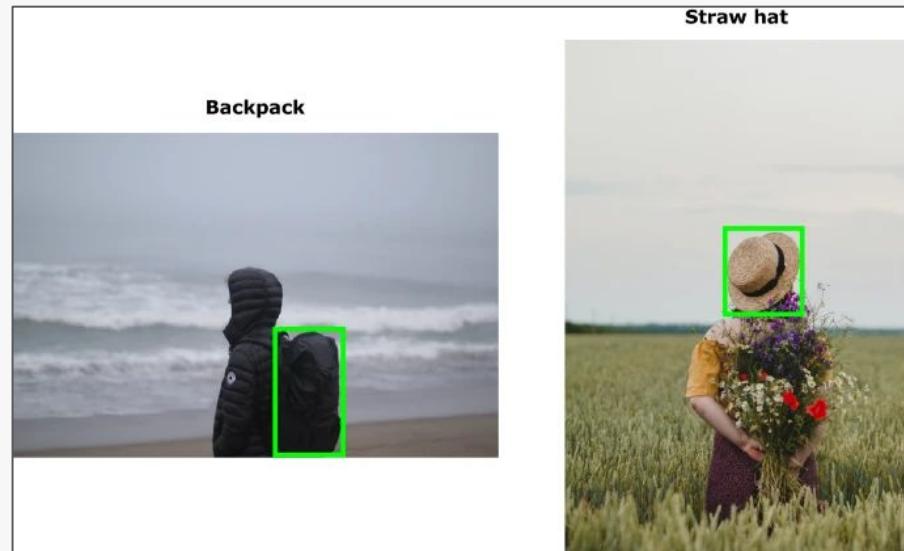
Patrick John Chia , Giuseppe Attanasio, Federico Bianchi, Silvia Terragni, Ana Rita Magalhães, Diogo Goncalves, Ciro Greco & Jacopo Tagliabue

[Scientific Reports](#) 12, Article number: 18958 (2022) | [Cite this article](#)

21k Accesses | 32 Citations | 14 Altmetric | [Metrics](#)

 A Publisher Correction to this article was published on 23 January 2023

 This article has been [updated](#)



AI engineering



The AI engineer

Three layers of responsibility

1. **Application Development** – prompt design, context management, user interfaces, evaluation loops.
2. **Model Development** – fine-tuning, dataset engineering, inference optimization.
3. **Infrastructure and LLM Ops** – serving models, managing compute, monitoring, scaling.

Model adaptation and application development

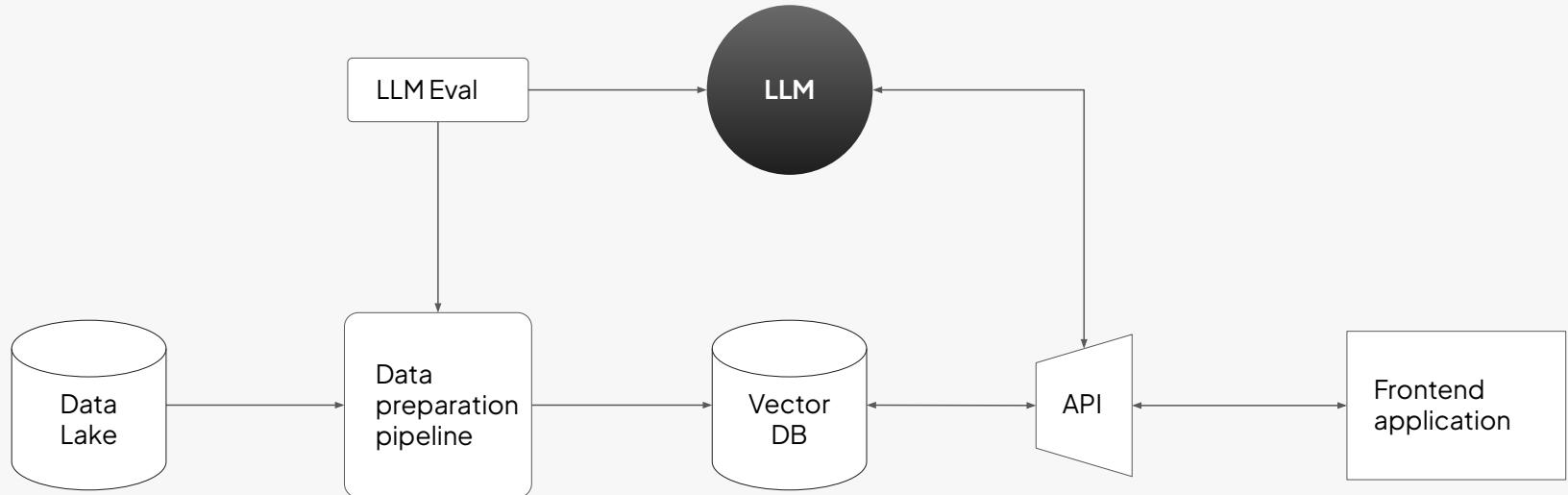
Key differences from ML engineering:

- Work with foundation models trained elsewhere. → Prompt engineering, context engineering, sampling, fine-tuning
- Handle larger, more compute-intensive models (GPU scaling, efficiency). → LLMOps and DataOps
- Manage open-ended outputs → much harder evaluation, testing and guardrails. → Eval, Eval, Eval!

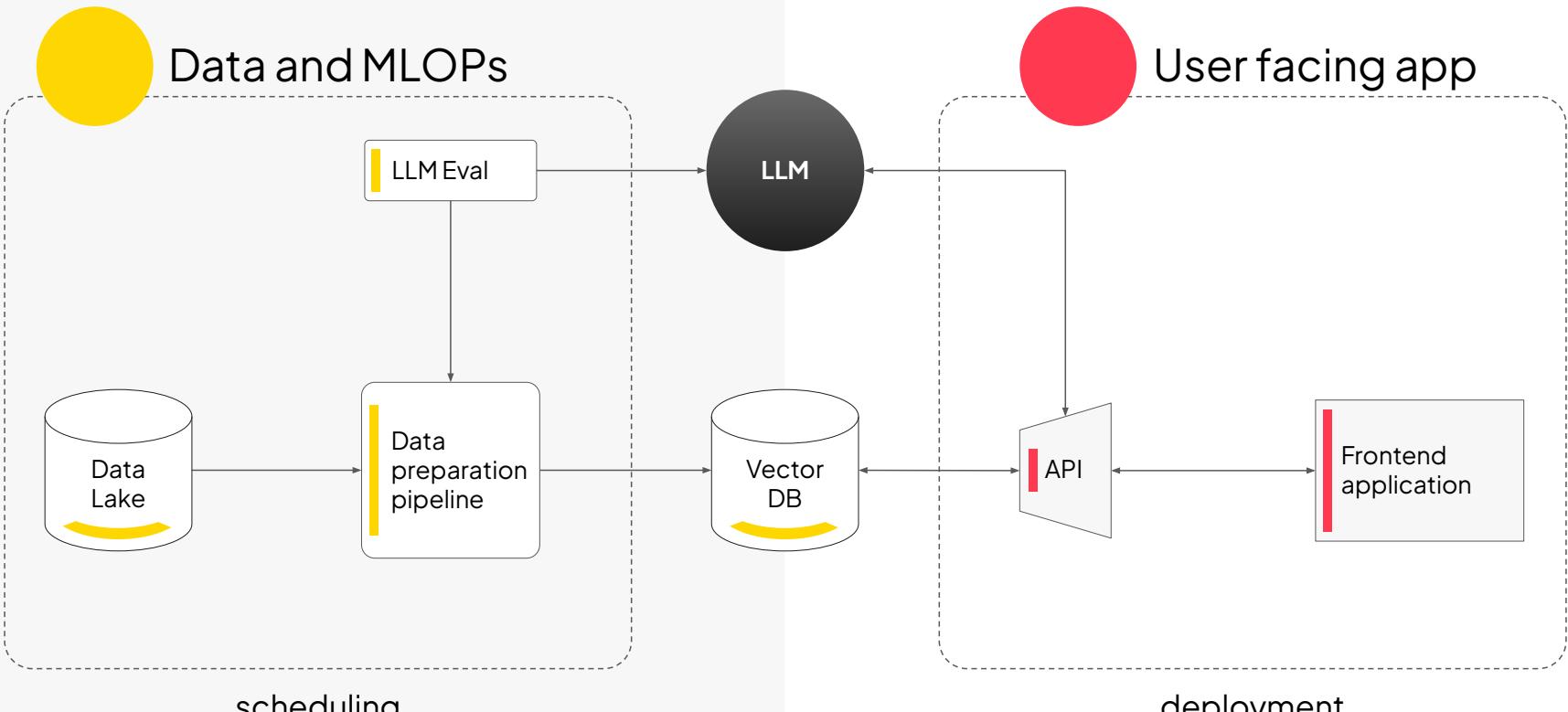
Model adaptation and application development

Closer to product & full-stack work:

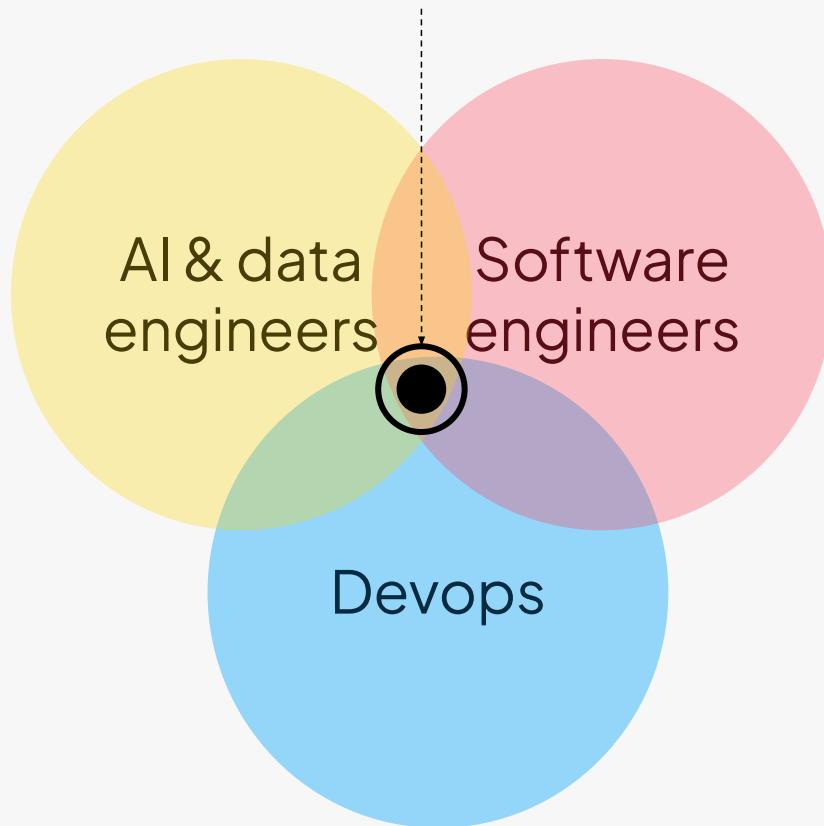
- AI engineers increasingly collaborate on UX, feedback loops, and product decisions.
- Many come from web dev / full-stack backgrounds, not only ML.



offline online



Your AI
application!



Historically, productionizing ML has been pretty hard



I am a data scientist, I work with csv files and notebooks

Data

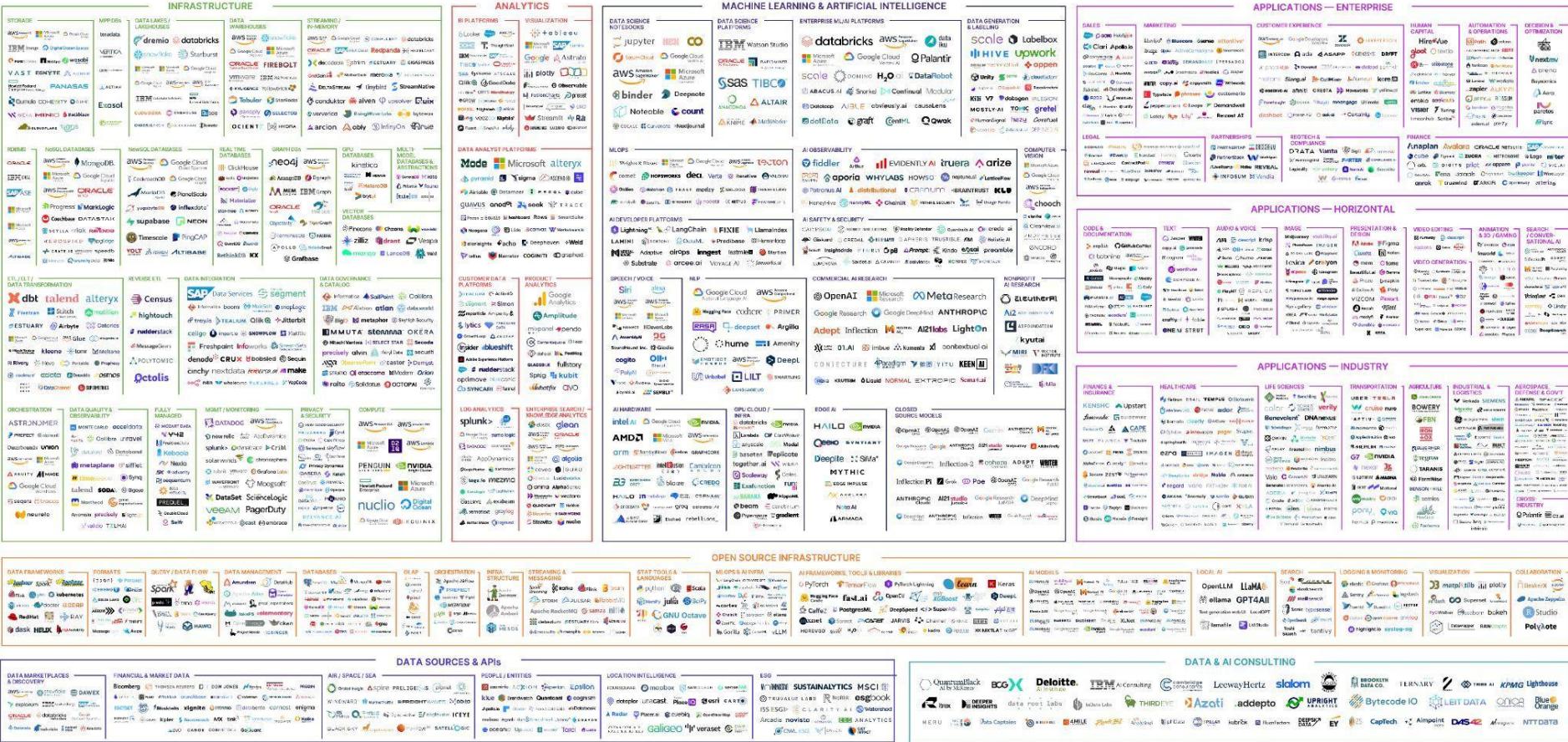


Notebooks on my laptop and Python scripts



DevOps

THE 2024 MAD (MACHINE LEARNING, ARTIFICIAL INTELLIGENCE & DATA) LANDSCAPE



It still is, but LLMs made it much easier to get started

- Product mentality is a must-have.
- IT mentality is a big no.
- So:
- Invest in FinOps! Make your costs easy to monitor and control.
- Invest in Developer Experience! Make your infrastructure simple so your product developers can participate.

Paradigm shift in product development

Circa 2015–2023*

- Invest in data
- Invest in Analytics
- Invest in Data Science
- Invest in ML and AI
- Deploy AI applications
- Measure ROI

Today

- Put together a prototype
- Measure rough ROI
- Invest in data
- Invest in Analytics
- Invest in Data Science
- Invest in ML and AI

Application development

**Data science + model development +
MLOps**

Data

Application development

LLM APIs

Data

AI will not fix your data problem

- **Data strategy** - your data touch engineering, product, security compliance and legal. You might want to think about it thoroughly.
- **Data quality** - if AI has high ROI for you and you have some special that only you have, using OpenAI won't cut it.
- **Data infrastructure** - data infrastructure is historically super hard and an IT domain. Simplify and put it in the hands of application builders.

Data centric AI

- The models are a commodity.
- Textual public data also is.
- Your data is what makes you special.

Hazy Research

People | Blog

July 19, 2021

The Road to Software 2.0 or Data-Centric AI

by Chris Ré



This article gives a brief, biased overview of our road to data-centric AI (AKA Software 2.0). The hope is to provide entry points for people interested in this area, which has been scattered in nooks and crannies of the overall AI picture—even while it drives some of our favorite products, advancements, and benchmark improvements. Our plan is to collect pointers to these resources on GitHub, write a few more articles about exciting directions, and engage with folks who are excited about it.

Maybe you?

[Data Centric AI Resource](#) [Mailing List](#) [Interest Form](#)

The curtain opens...

Why should you care about data?

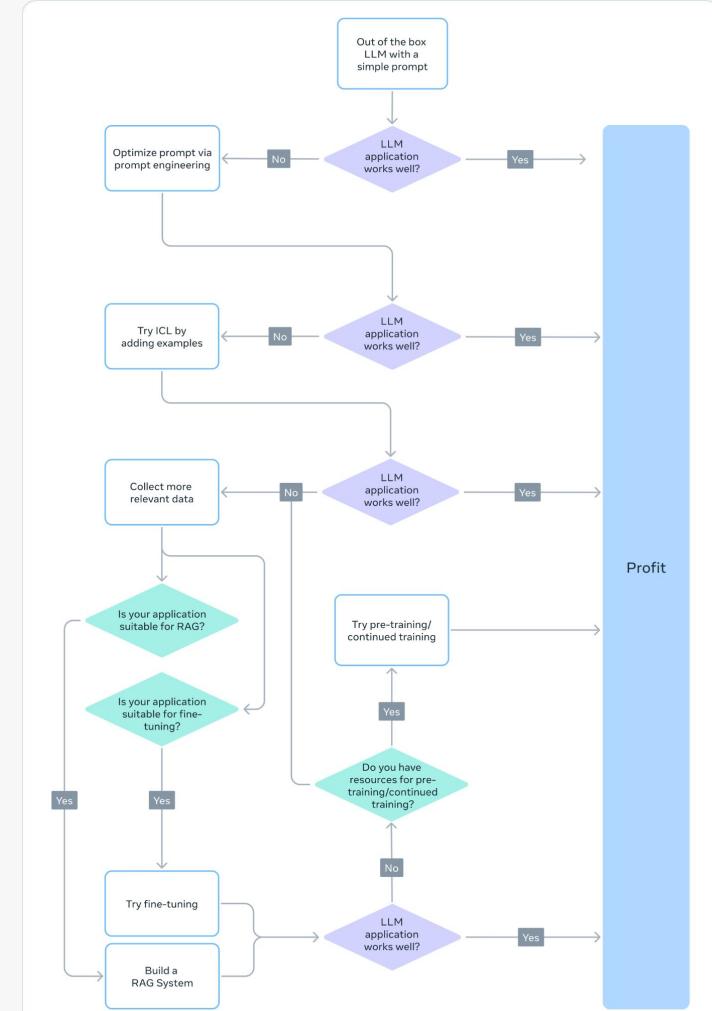
- **Pretraining is expensive and hard. Nobody really needs that, except those who build foundational models.**
 - 50B to ~300B parameters, \$5 to \$50M for end-to-end training, there are probably 2000 people in the world that really know how to do this.
 - Done yearly.
 - Quantity over quality
 - Kind of the same for everybody.
- **Prompt engineering, post-training, fine-tuning, context engineering is where most of the improvement is.**
 - Fine-tuning on labelled data - 100k to 500k tokens.
 - Done daily or weekly.
 - Quality over quantity.
 - Your specific data matter!

Why should you care about data?

- **Differentiate** – Use data specific to your business.
- **Integrate** – Use accurate, up-to-date information.
- **Govern** – Ensure auditability, compliance, and trust.
- **Improve** – Use feedback data to improve AI systems.

Buy vs. build

- Spend as much time as you can understanding the use case.
- Identify weak vs. strong constraints.
- Start simple and determine ROI.
- Do not - I repeat **do not** - do what Google does.
- Assume you'll operate at a reasonable scale.



Source: <https://ai.meta.com/blog/adapting-large-language-models-langs/>

Problem Domain	Why not LLMs
Recommender Systems	Sensitive to prompt ordering → unstable, high variance (arXiv)
Time Series Forecasting	No better than simpler models; lack temporal structure and are computationally expensive (arXiv , NeurIPS , ResearchGate)
Anomaly Detection	Specialized models outperform LLMs; LLMs require workaround and are less reliable (ResearchGate , ScienceDirect , MIT News)
Common Challenges	Structured data, precision needs, temporal consistency, cost constraints, and stability requirements .

Recommender systems

- Most of ML applications out there are actually recommender systems.
- Recsys are a big deal as they are ubiquitous in our life:
 - Movies
 - Books
 - News
 - Shopping
 - People
- The Recsys market is predicted to reach around 15 BN by 2026.

