Large Language Models in Healthcare

Day 1: Introduction to LLMs

UMC Utrecht

Ruurd Kuiper – Julius Center, UMCU

Getting to know me

Ruurd Kuiper

Assistant professor in NLP/AI for Healthcare
@ University Medical Center Utrecht (UMCU)

- PhD in AI for Medical Image Analysis
- MSc in Biomedical Engineering
- BSc in Biochemistry

Currently working (not exhaustive):
- Validation of ChatGPT for anesthesiological consultations
- LLMs as a radiology assistant
- Review on validation methods for LLMs in healthcare
- Diffusion Language Models (DLMs)
- GENAISIS (establishment of LLM lifecycle in healthcare)

Email: R.J.A.Kuiper@UMCUtrecht.nl

Organizational

Lecture days are from 8:30 to 16:30

First three days, the practical sessions will be individual (although helping is encouraged)

Four days of lectures, last day dedicated to groupwork + presentation

Practical sessions will be done in Python (free to use anything else for groupwork)

Slides will be made available online

Contents

1. Course overview
2. Preliminary assessment
3. Introduction to language models

SHORT BREAK

4. Core technical concepts
5. Types of LLMs
6. Current capabilities and risks
7. Quick quiz

LUNCH BREAK

6. Assignment
7. Presentation/discussion

Course overview

Day 1: A general introduction to Large Language Models
A broad introduction to LLMs, their foundational concepts, and current capabilities and risks.

Day 2: Deep dive into LLM mechanics & training from scratch
An in-depth exploration of the internal workings of LLMs and a hands-on experience in building a small model.

Day 3: Multimodal/foundation models in healthcare
Exploring models that integrate multiple data types and hands-on experience with currently available (multimodal) models.

Day 4: The life-cycle of training medical LLMs
Understanding the process of developing and deploying LLMs for medical applications, in a detailed project plan.

Day 5: A first attempt at implementing your own LLM project
Translating what has been learned into initial, tangible implementation steps.

Day 1: Introduction to LLMs in Healthcare

Theme: A broad introduction to LLMs, their foundational concepts, and their current capabilities.

Learning Objectives:

1. Define what a Large Language Model (LLM) is and its significance.
2. Understand the general architecture and working principles of LLMs.
3. Identify different types of LLMs and their core functionalities.
4. Explore common applications and current limitations of LLMs in healthcare

5. Ideate and research possible applications of LLMs in your field

Getting to know you

Please note:

- **There are no wrong answers**
- **This is not meant to test you, but to inform me**
- **This course is designed to teach you something regardless of your current level**

CODE: 1463 6379

https://www.menti.com/alirfe27b4kh

Introduction to LLMs in healthcare

Contents

1. Historical context: From traditional NLP to deep learning and transformers.
2. What are Large Language Models?
3. The "largeness" factor: Parameters, data, and computational scale.

Introduction to LLMs in healthcare

The rise of LLMs

The New York Times

Art World Takes On A.I. | Putting A.I. in Charge | A.I. and Hollywood | Microsoft-OpenAI Partnership

A.I. Chatbots Defeated Doctors at Diagnosing Illness

A small study found ChatGPT outdid human physicians when assessing medical case his
using a chatbot.

nature

Explore content | About the journal | Publish with us | Subscribe

nature > news > article

NEWS | 12 January 2024

Google AI has better bedside manner than human doctors — and better diagnoses

their artificial-intelligence system could help to democratize medicine.

NEWS | 17 September 2025

Which diseases will you have in 20 years? This AI accurately predicts your risks

A modified large language model called Delphi-2M analyses a person's medical records and lifestyle to provide risk estimates for more than 1,000 diseases.

Health Tech World

Latest | Awa

AI

ChatGPT can produce medical notes 10x faster than doctors

By News Editor Published On: March 28, 2024 Last Updated: March 27, 2024

About | Centers | Research | Education | Policy

Healthcare, Language Processing

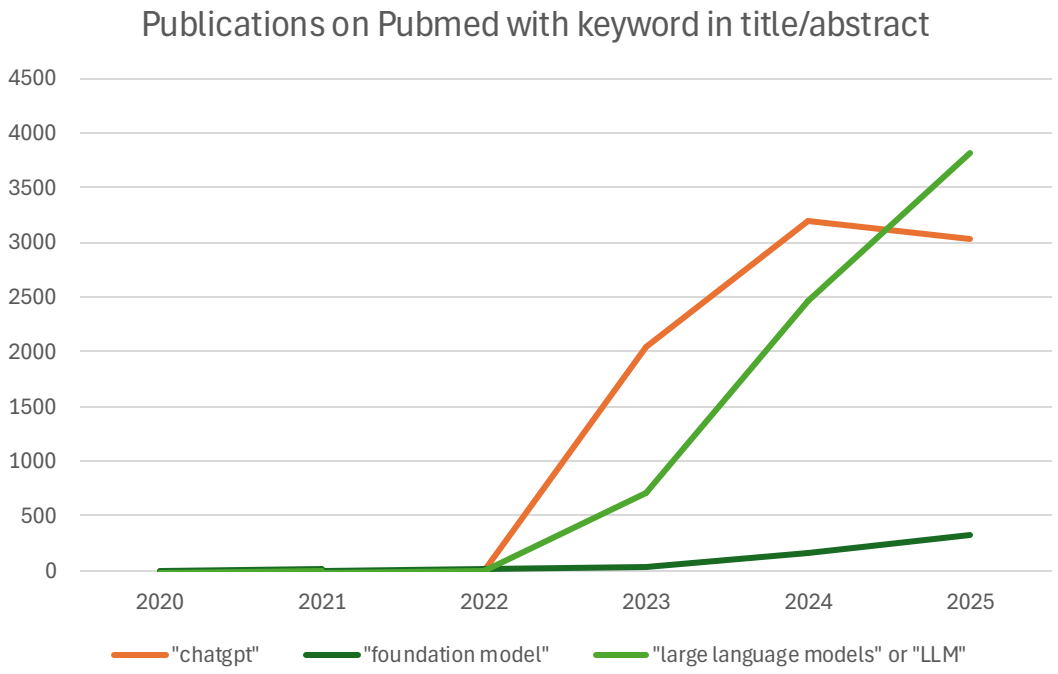
AI can Outperform Humans in Writing Medical Summaries

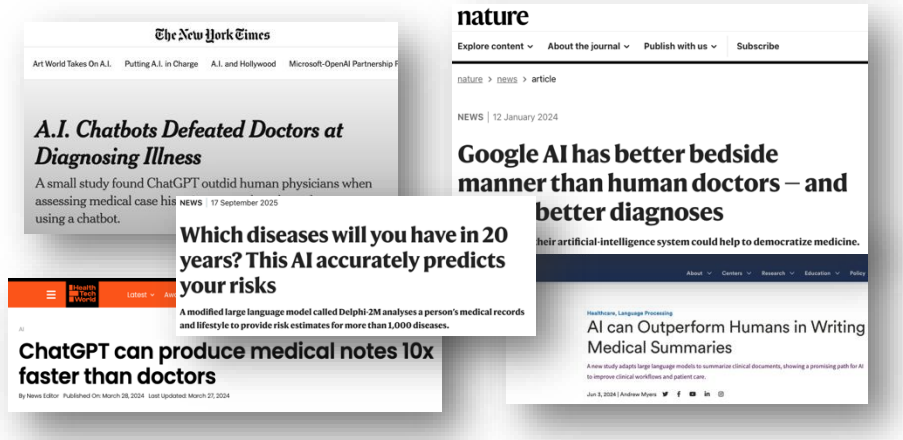
A new study adapts large language models to summarize clinical documents, showing a promising path for AI to improve clinical workflows and patient care.

Jun 3, 2024 | Andrew Myers

Introduction to LLMs in healthcare

The rise of LLMs

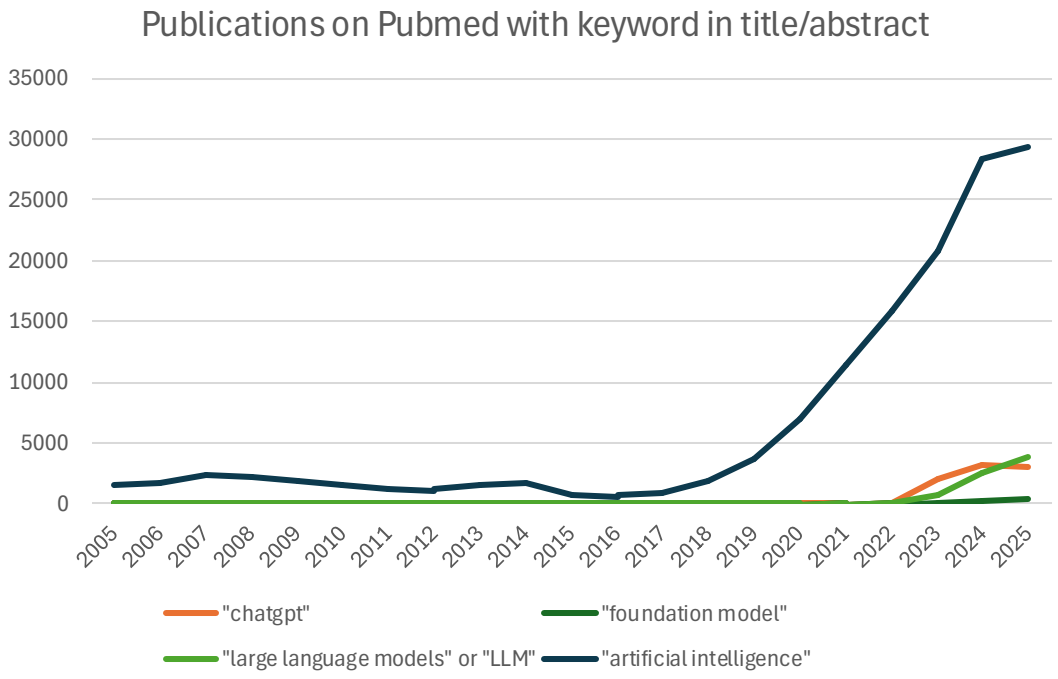


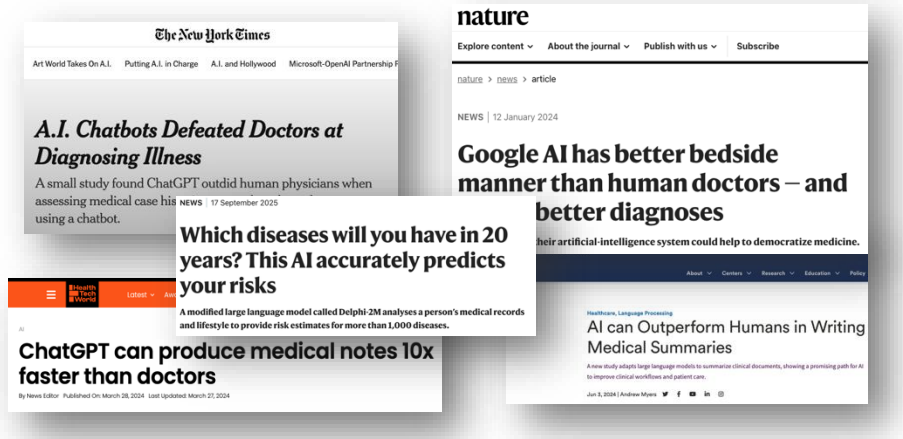


AI and LLM related papers have exploded in recent years

Introduction to LLMs in healthcare

The rise of LLMs



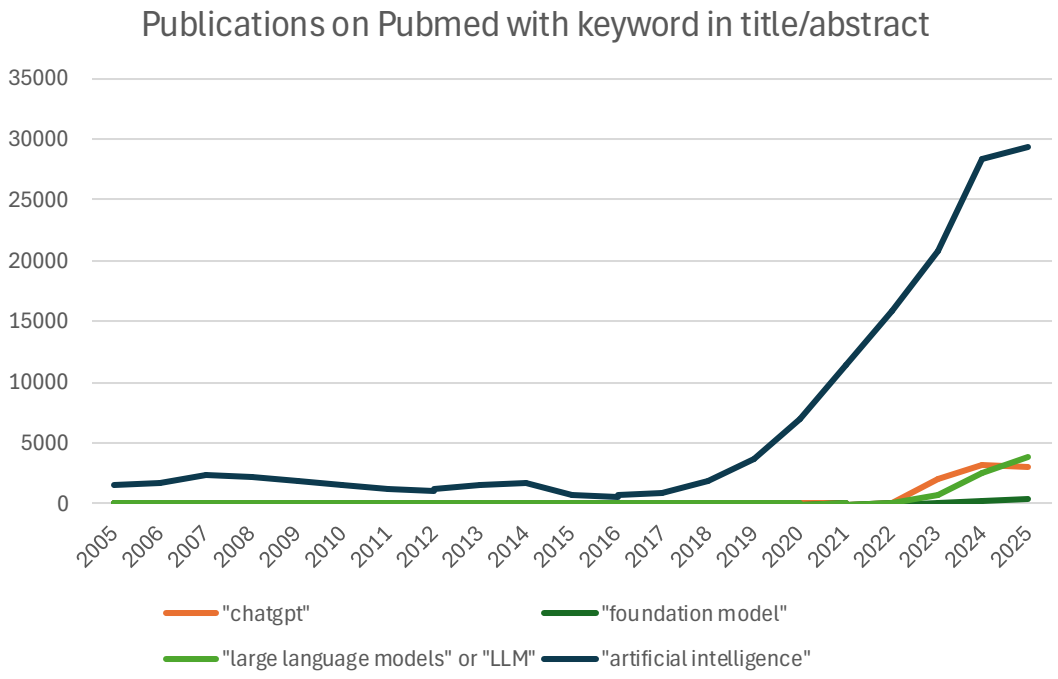


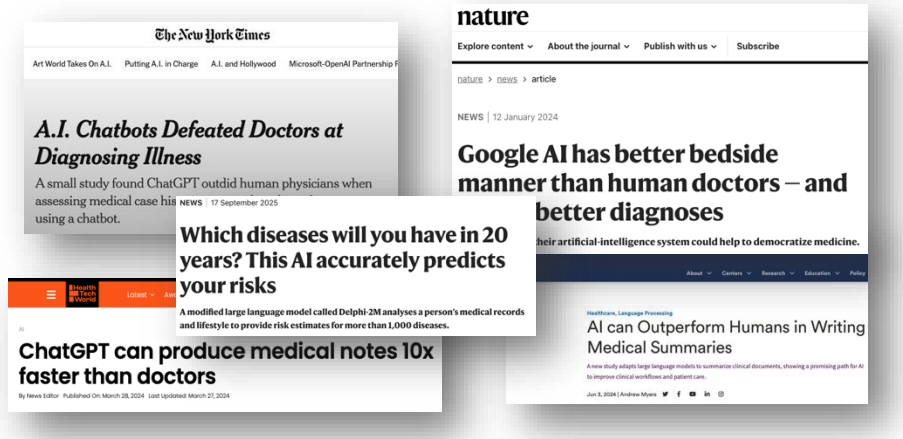
AI and LLM related papers have exploded in recent years

UMC Utrecht

Introduction to LLMs in healthcare

The rise of LLMs

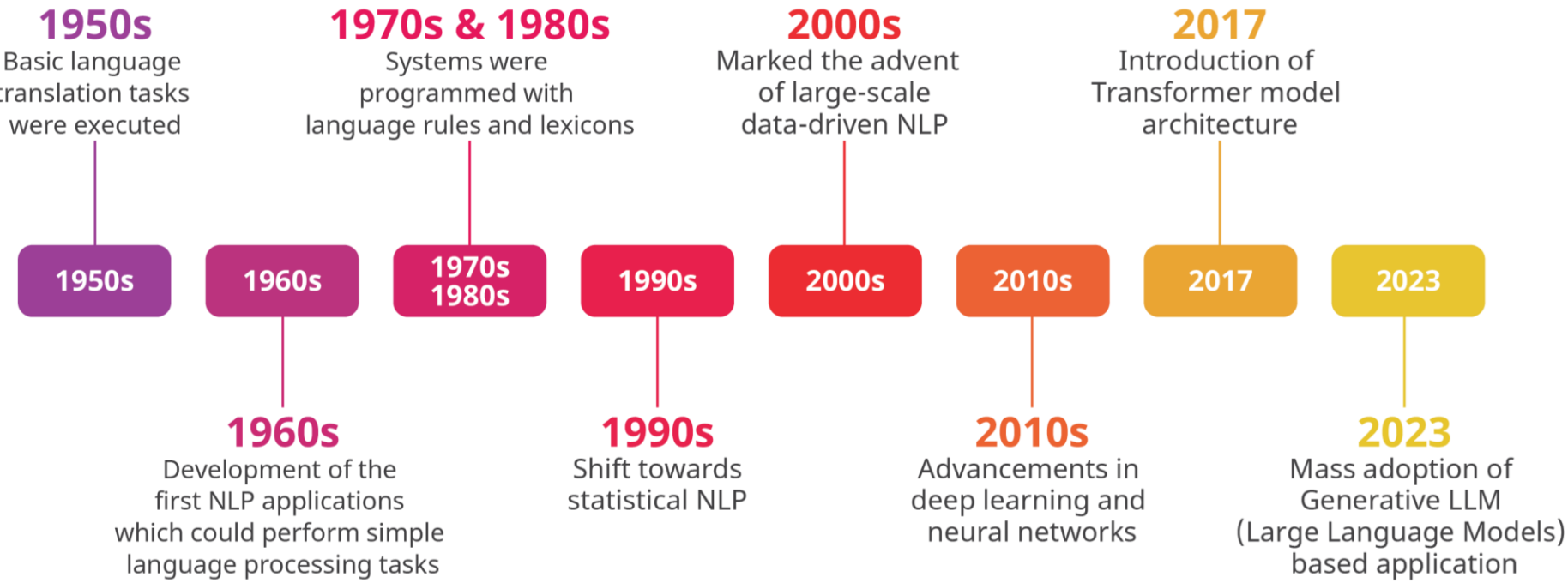




AI and LLM related papers have exploded in recent years, but why now?

Introduction to LLMs

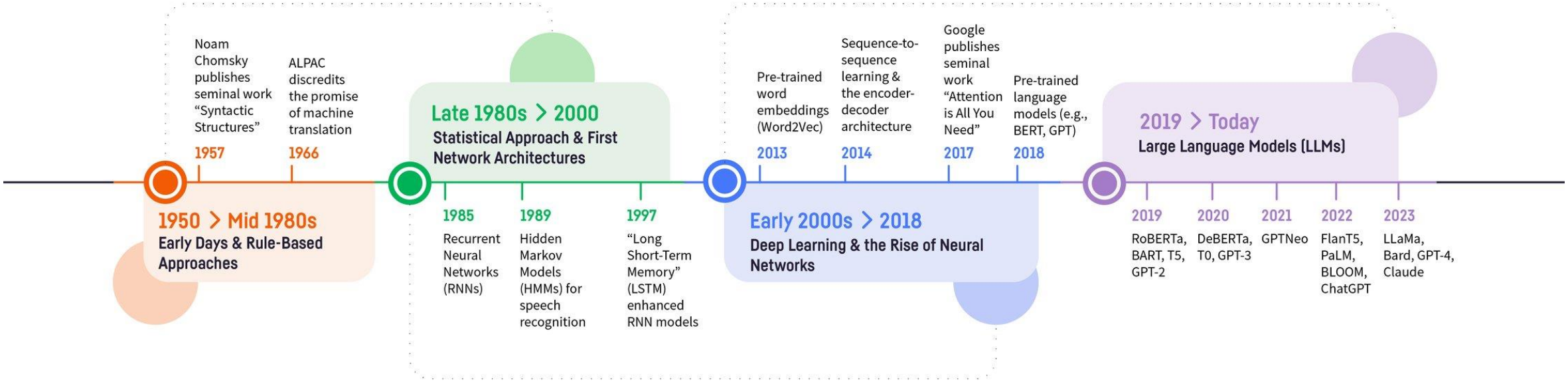
Historical context: From traditional NLP to deep learning and transformers.



Source: https://commtelnetworks.com/exploring-the-impact-of-natural-language-processing-on-cni-operations/

Introduction to LLMs

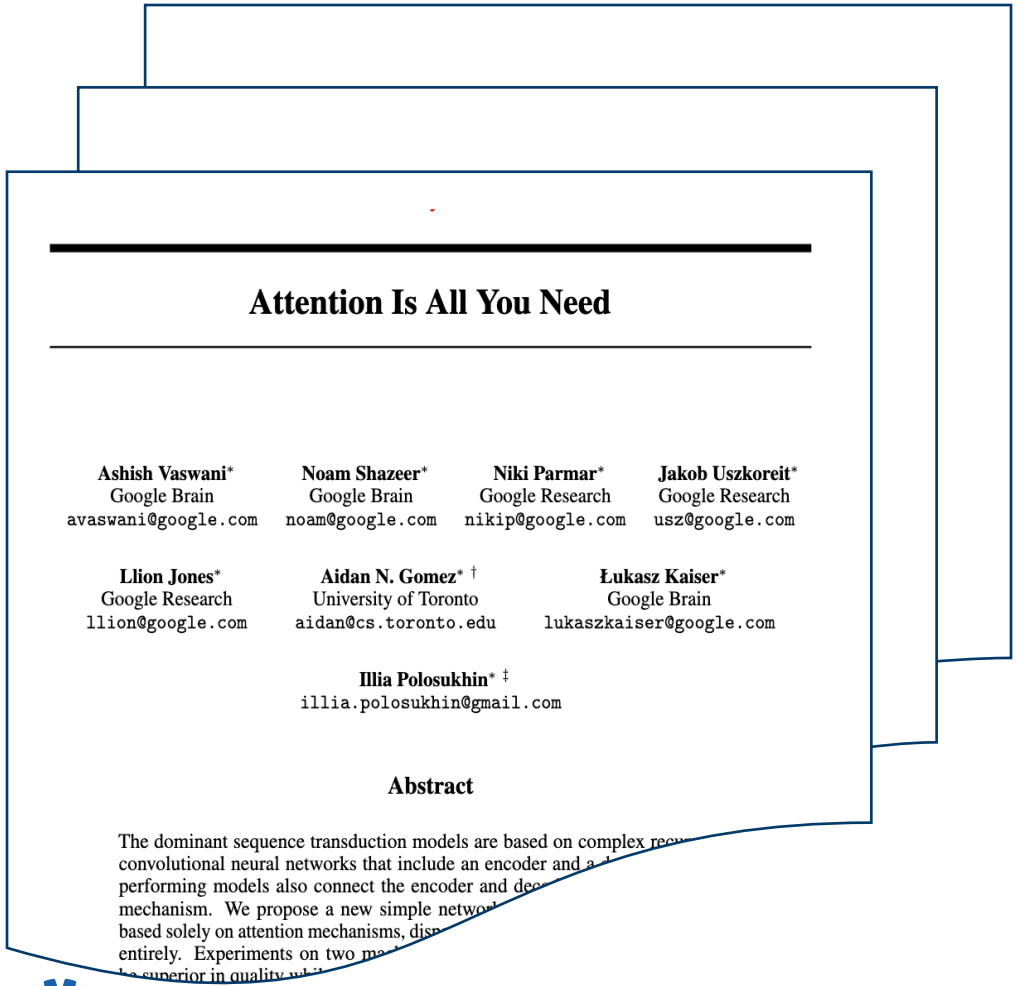
Historical context: From traditional NLP to deep learning and transformers.



Source: https://blog.dataiku.com/nlp-metamorphosis

Introduction to LLMs

Historical context: From traditional NLP to deep learning and transformers.



Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

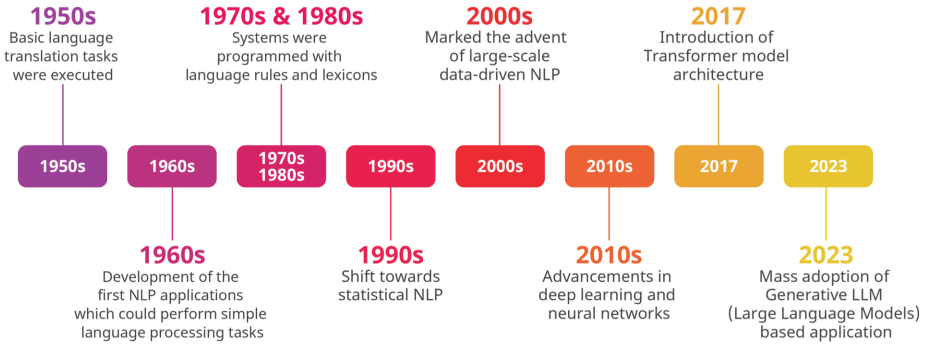
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

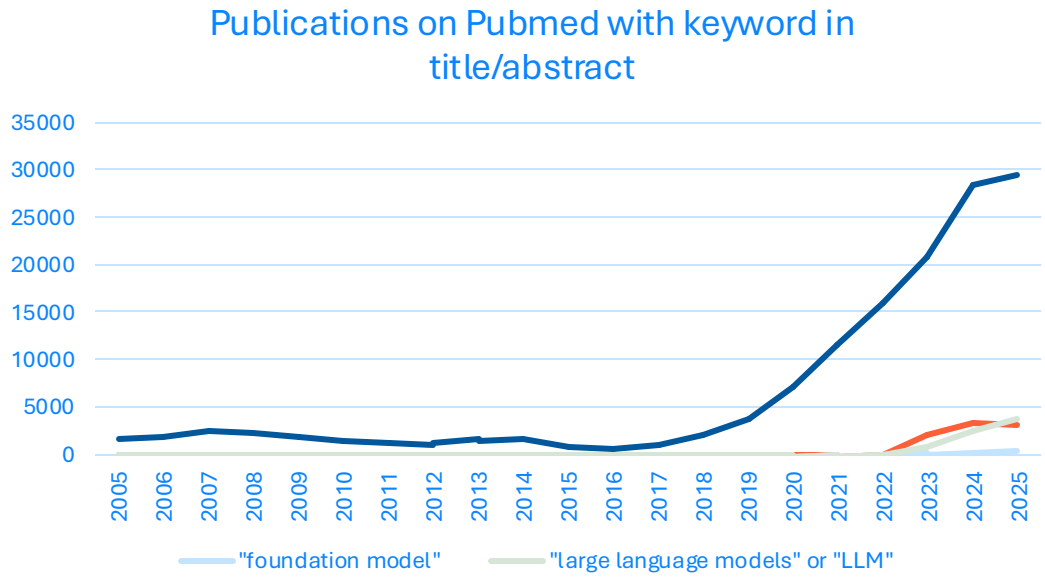
Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex rec
convolutional neural networks that include an encoder and a
performing models also connect the encoder and dec
mechanism. We propose a new simple netwo
based solely on attention mechanisms, dis
entirely. Experiments on two ma
be superior in quality wh





UMC Utrecht

Introduction to LLMs

What are LLMs?

What do we already know?

Introduction to LLMs

What are LLMs?

What do we already know?

- AI systems **trained** on **large amounts** of (text) data
- Analyze and generate **natural language** (or other modalities)
- **Machine** learning -> **Deep** learning
- **General** knowledge
- Can be **adapted** to various tasks: fine-tuning or instruction-tuning

Introduction to LLMs

Definition

A **large language model** (LLM) is a **deep learning neural network** trained on **large (text) datasets** to learn **statistical patterns of language** enabling it to **generate text** by **predicting** the **mostly likely** next **token** in a sequence.

Introduction to LLMs

Definition

A **large language model** (LLM) is a **deep learning neural network** trained on **large (text) datasets** to learn **statistical patterns of language** enabling it to **generate text** by **predicting** the **mostly likely** next **token** in a sequence.

They can exhibit emergent capabilities such as contextual **understanding (?)** and **reasoning (?)**, allowing them to perform diverse tasks without task-specific training

Introduction to LLMs

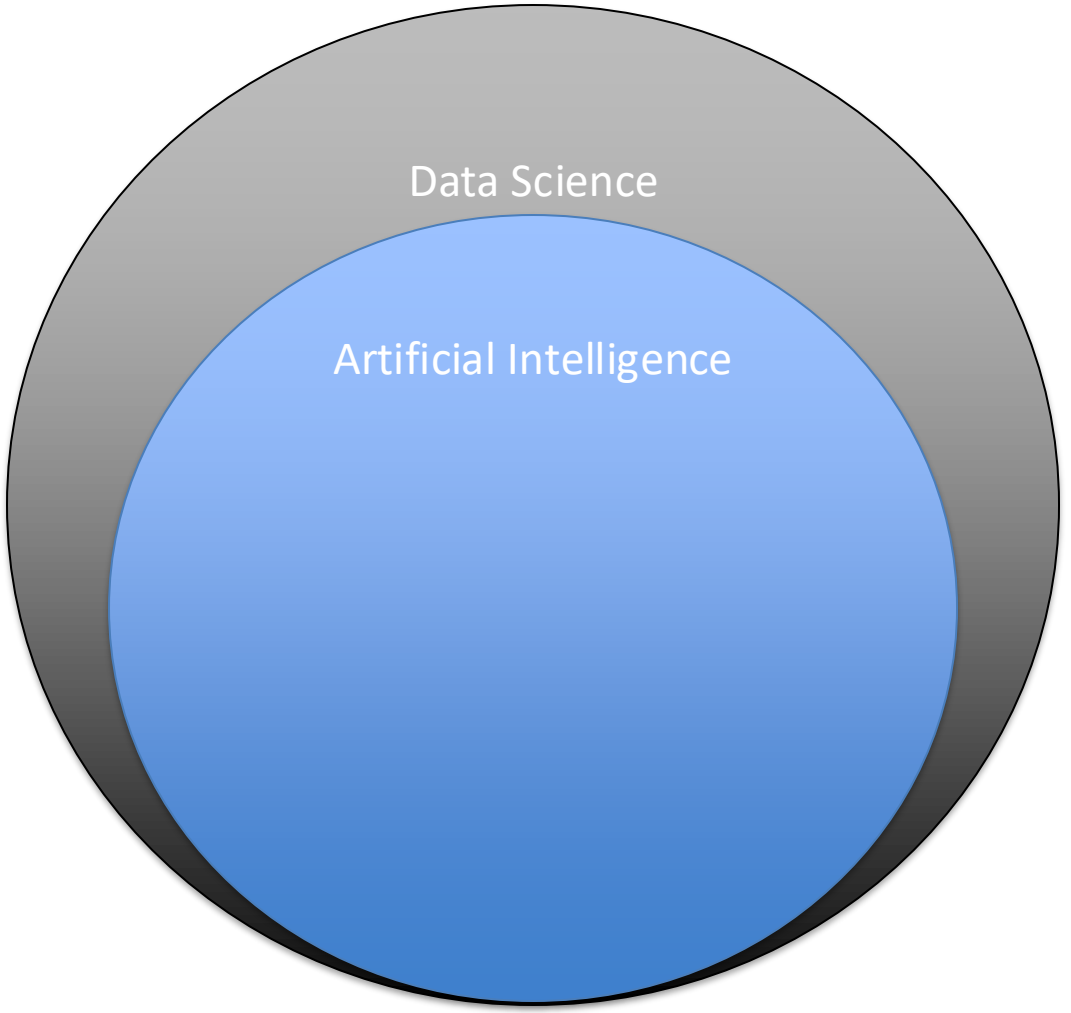
What are LLMs?

What field do LLMs belong to?

Introduction to LLMs

What are LLMs?

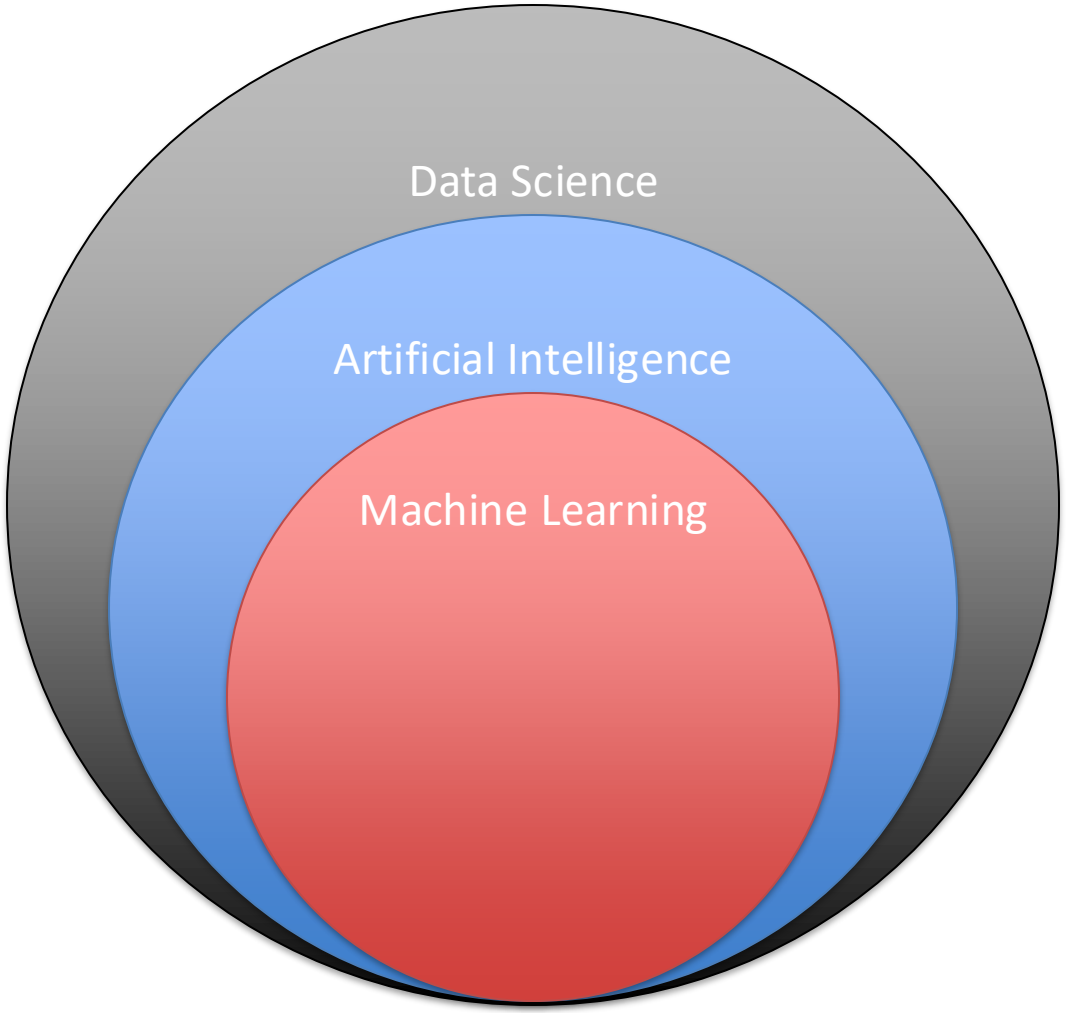
What field do LLMs belong to?



Introduction to LLMs

What are LLMs?

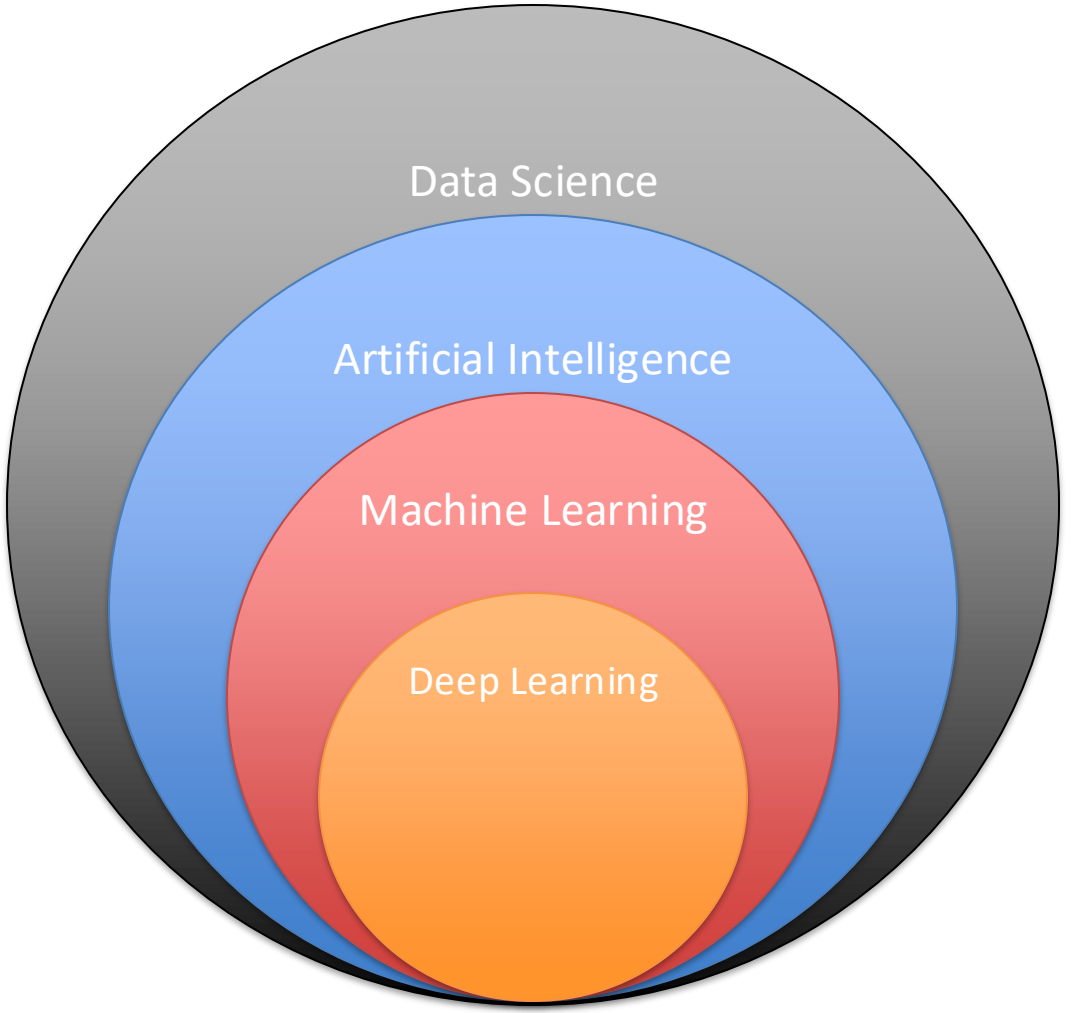
What field do LLMs belong to?



Introduction to LLMs

What are LLMs?

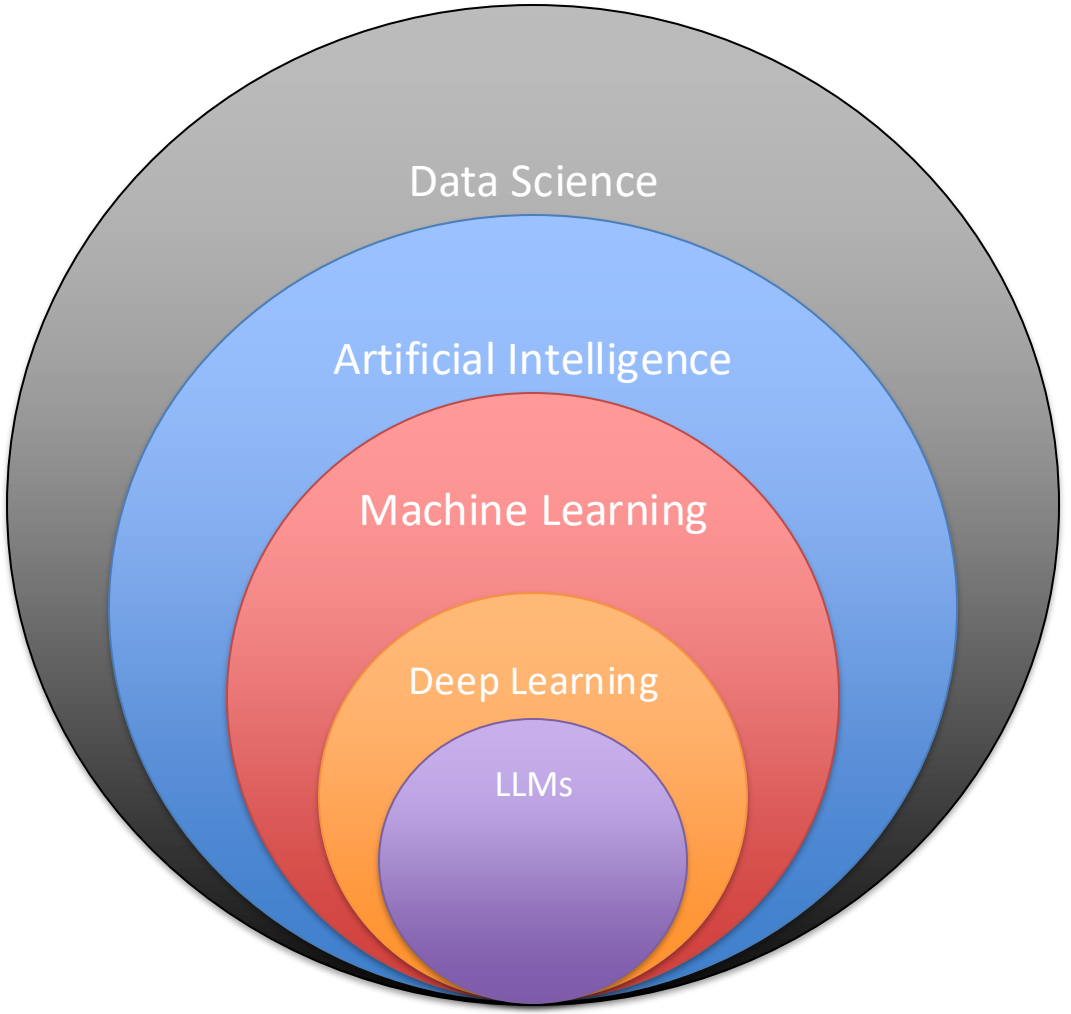
What field do LLMs belong to?



Introduction to LLMs

What are LLMs?

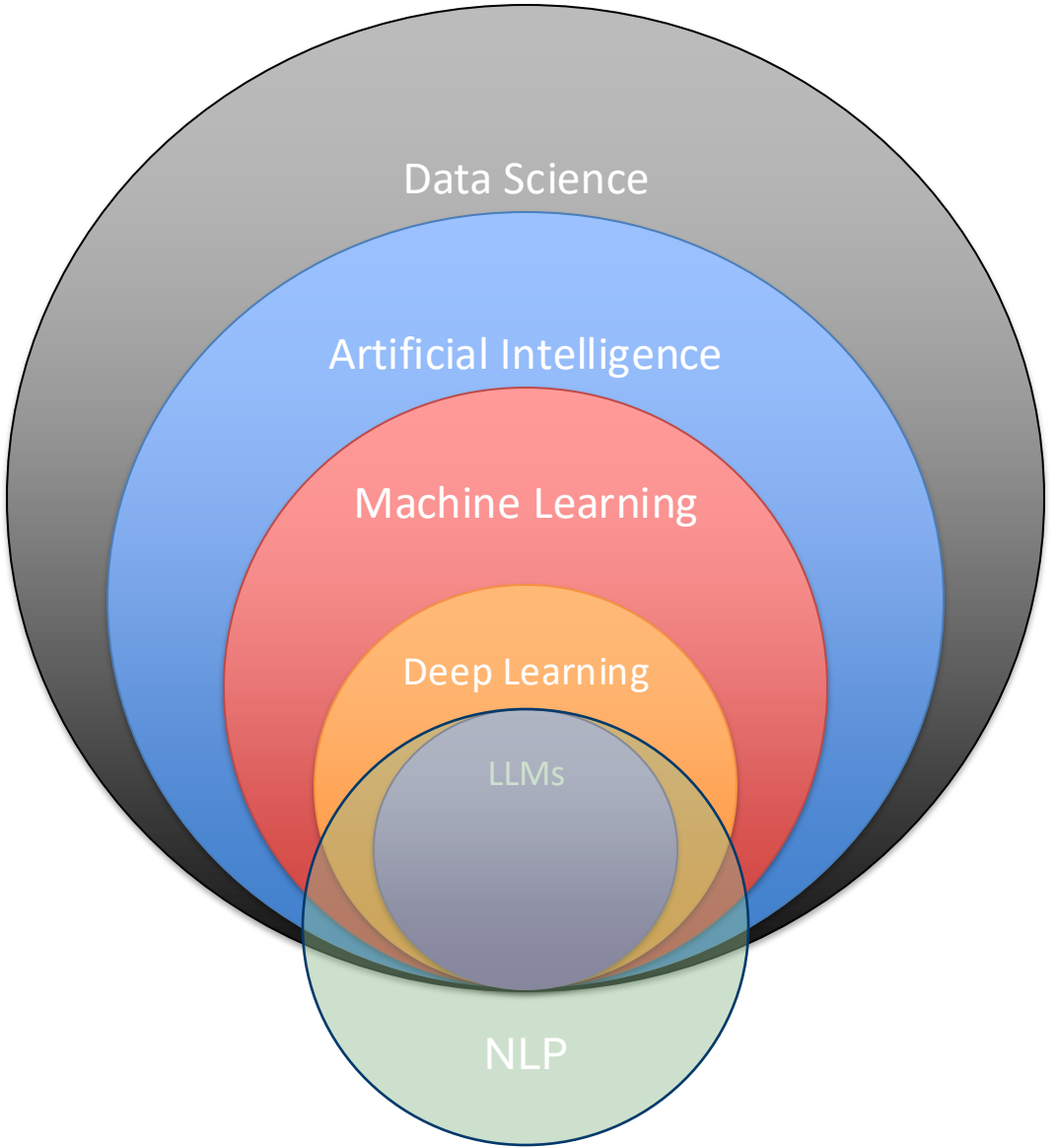
What field do LLMs belong to?



Introduction to LLMs

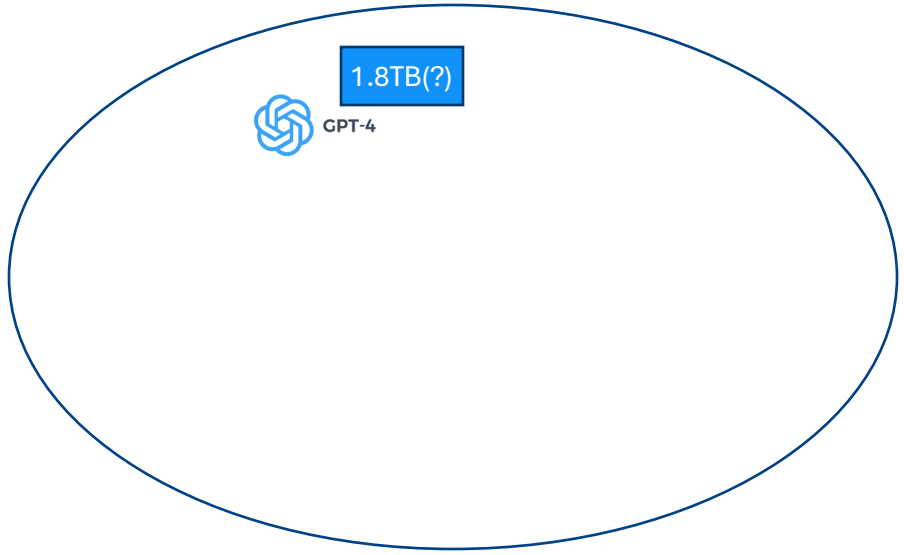
What are LLMs?

What field do LLMs belong to?



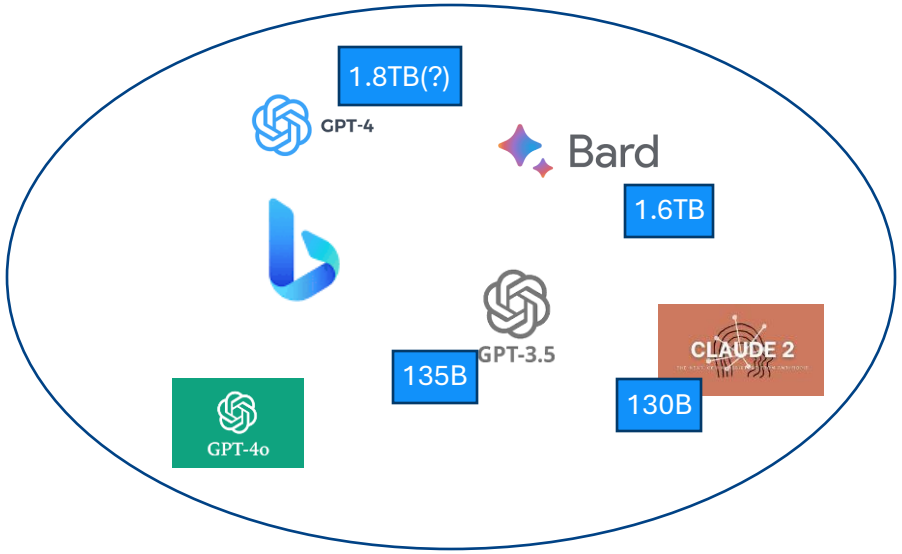
Introduction to LLMs in healthcare

What are LLMs?



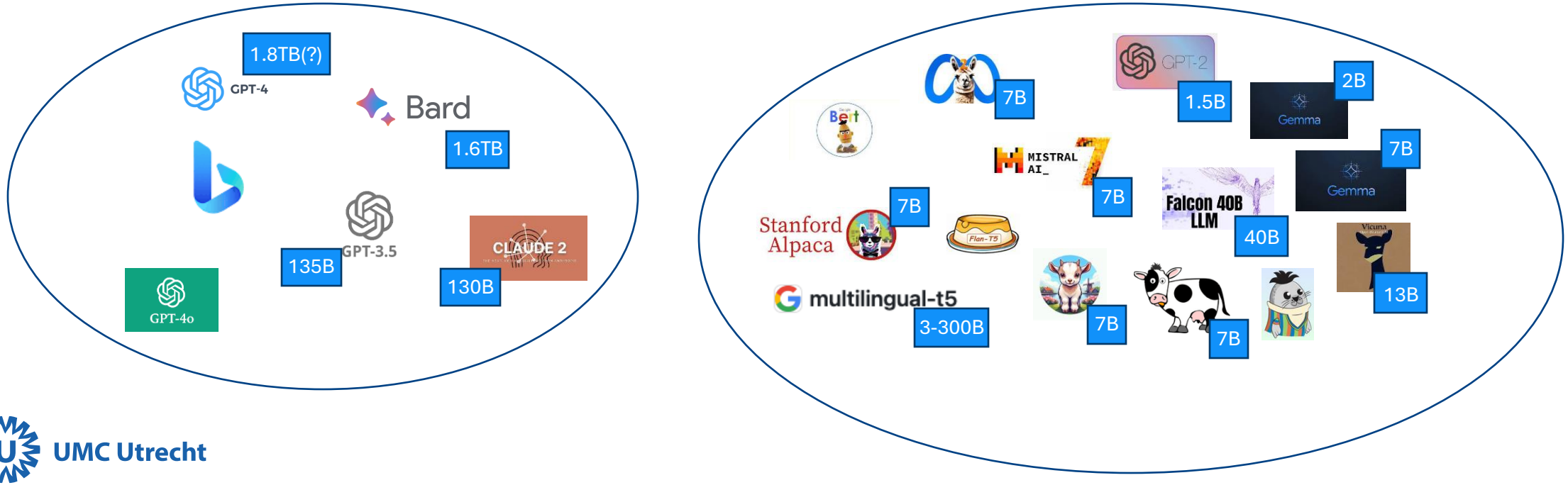
Introduction to LLMs in healthcare

What are LLMs?



Introduction to LLMs in healthcare

What are LLMs?

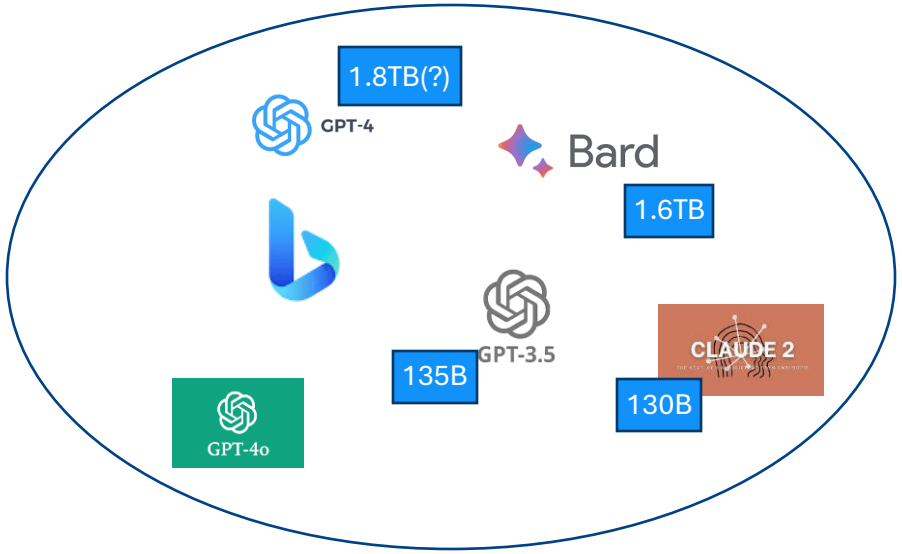


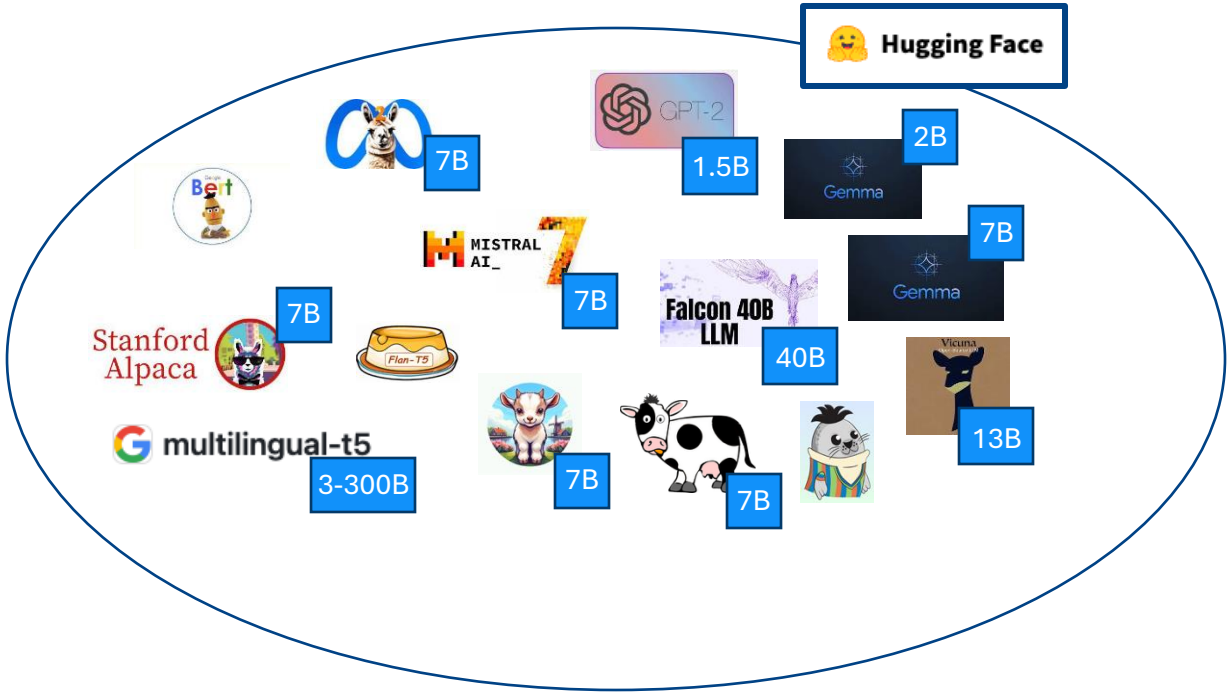
Introduction to LLMs in healthcare

What are LLMs?

Proprietary

- Easy to use in the browser
- Seemingly good performance
- Internet connection required
- Not transparent
- Not in own control (pro and con)
- Often more generic





Introduction to LLMs in healthcare

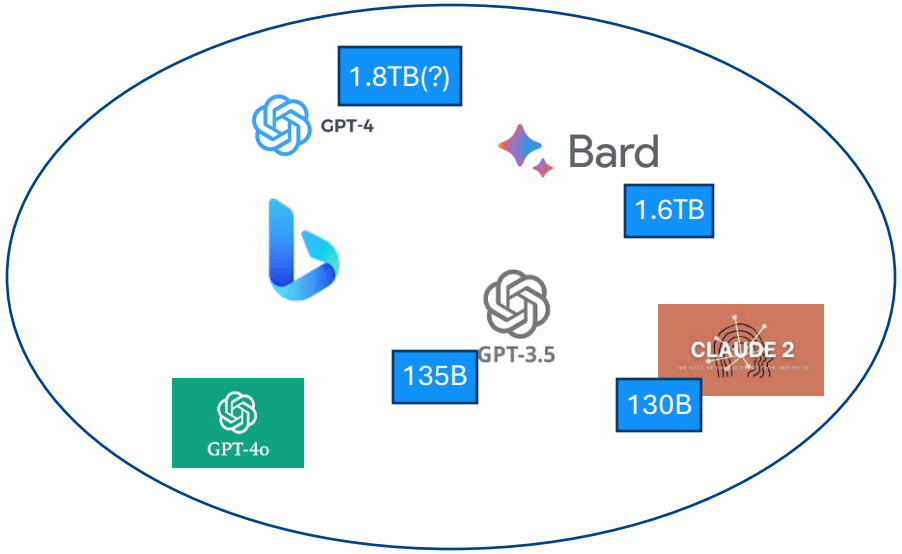
What are LLMs?

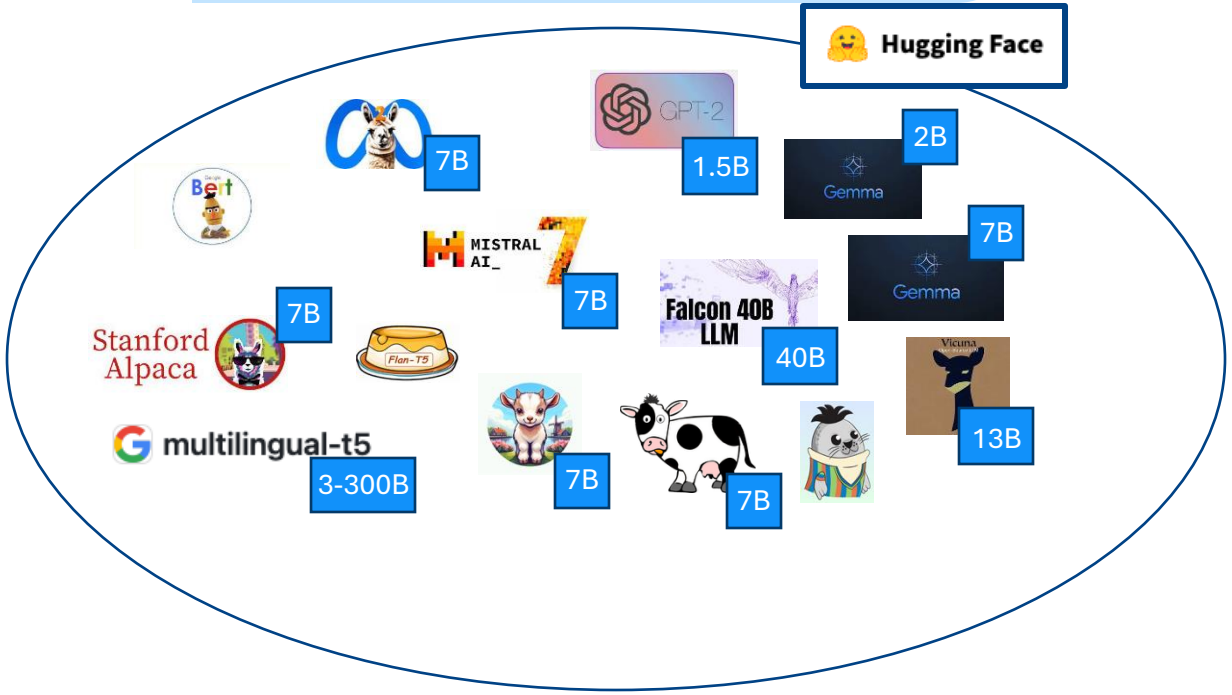
Proprietary

- Easy to use in the browser
- Seemingly good performance
- Internet connection required
- Not transparent
- Not in own control (pro and con)
- Often more generic

Open source

- Requires a bit of coding to setup
- Seemingly mixed performance
- No internet required
- Transparent
- In own control (pro and con)
- Often more specialized





Introduction to LLMs

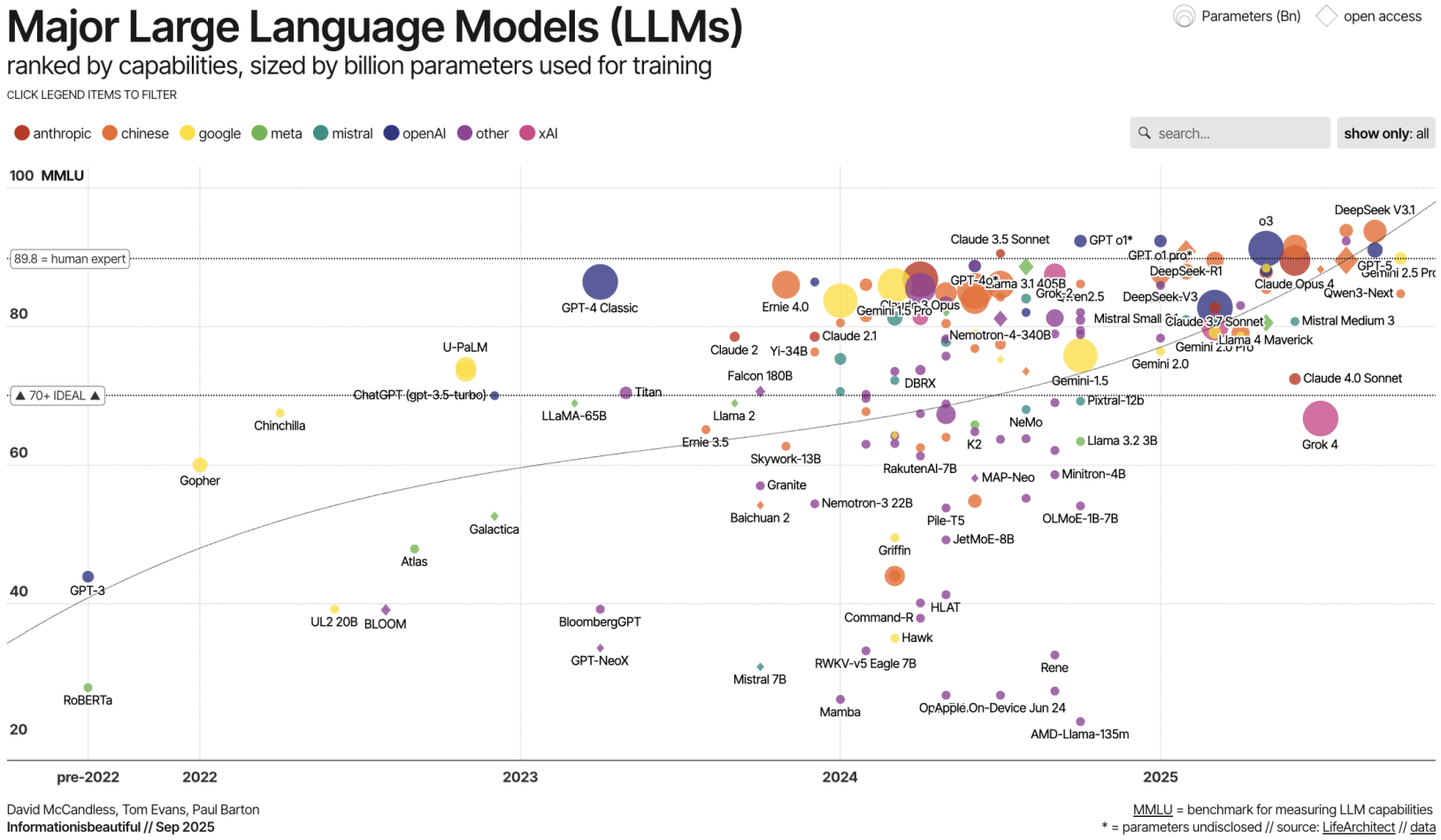
The "largeness" factor: Parameters, data, and computational scale.

Performance has been steadily increasing, but how? And will it continue?

Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

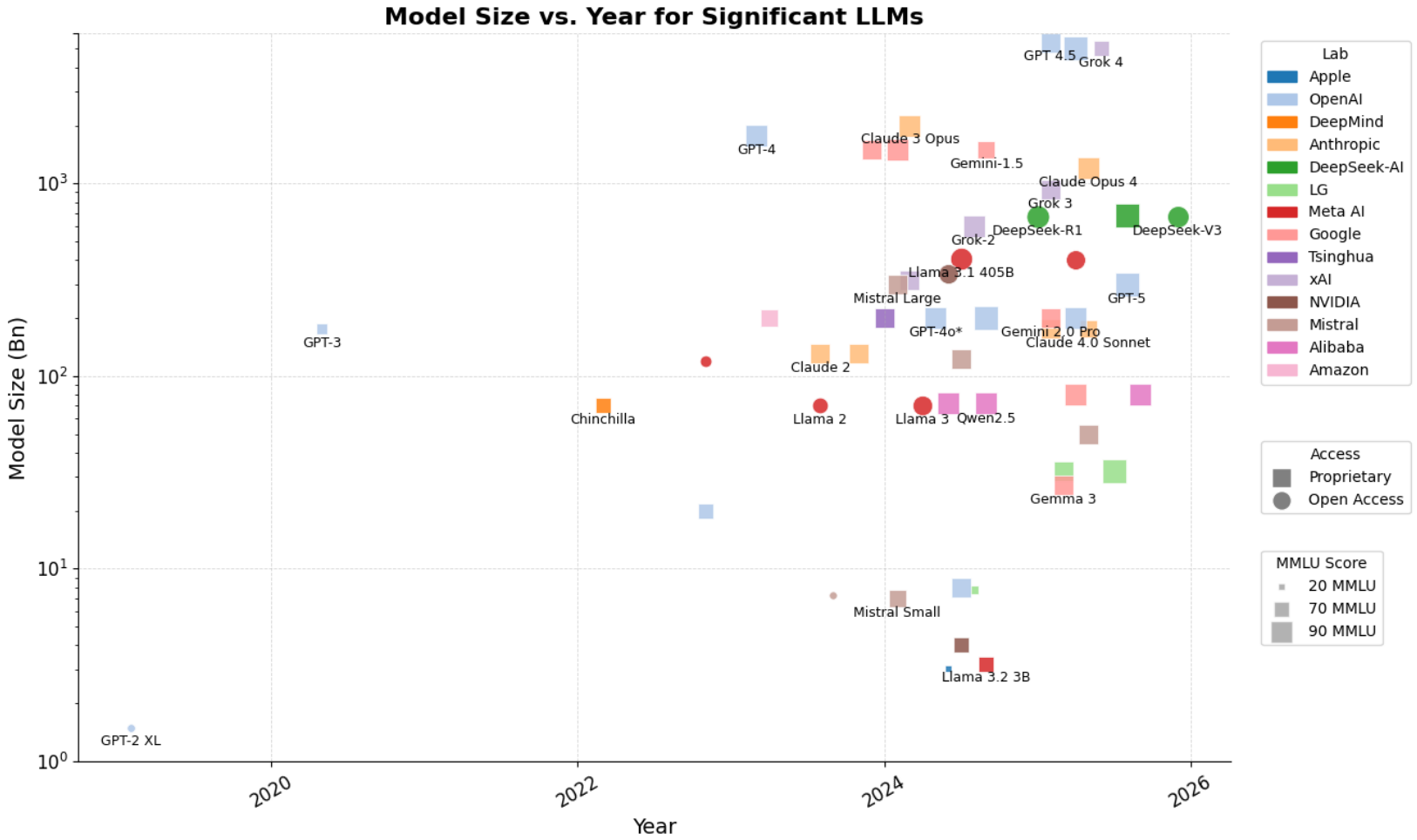
Performance has been steadily increasing, but how? And will it continue?



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

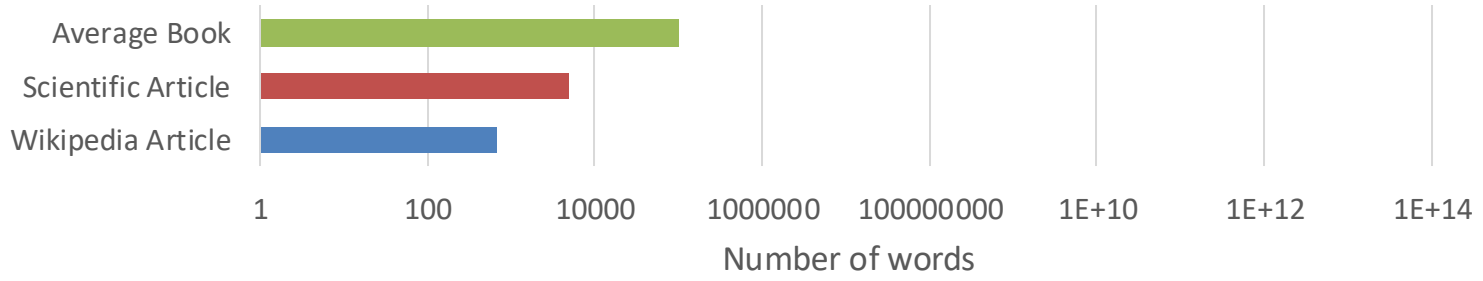
- **Parameters: from millions → hundreds of billions**



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

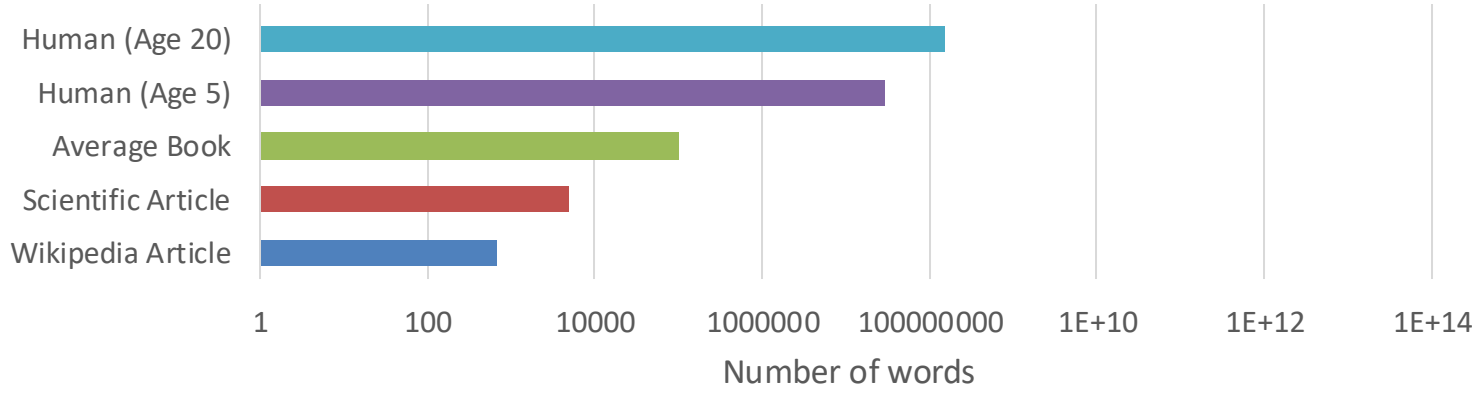
- **Parameters: from millions → hundreds of billions**
- **Data: web, books, scientific papers, medical notes**



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

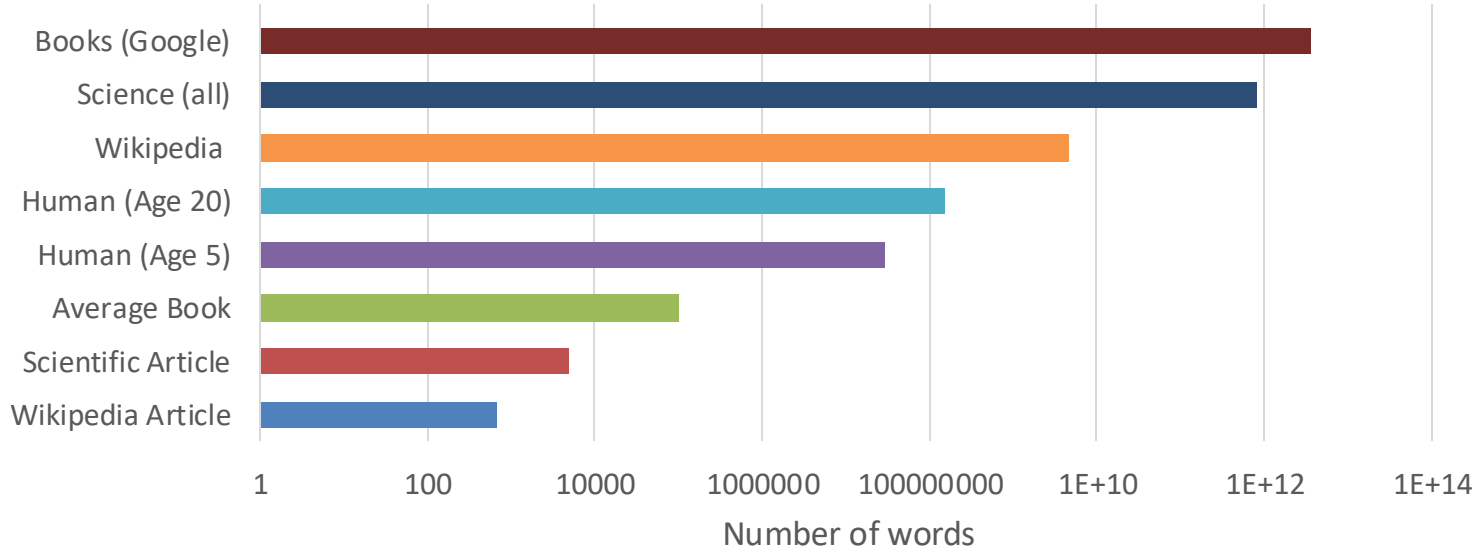
- **Parameters: from millions → hundreds of billions**
- **Data: web, books, scientific papers, medical notes**



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

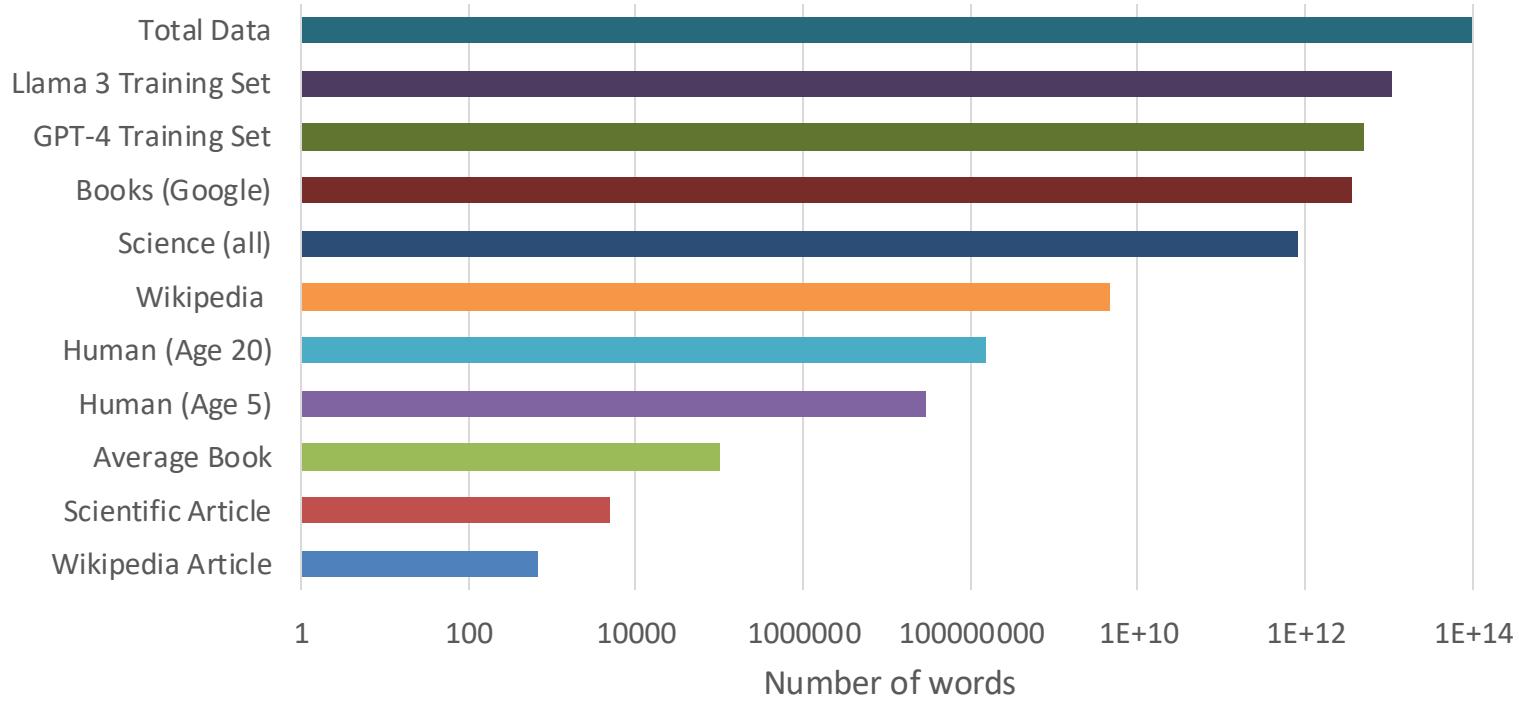
- **Parameters: from millions → hundreds of billions**
- **Data: web, books, scientific papers, medical notes**



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

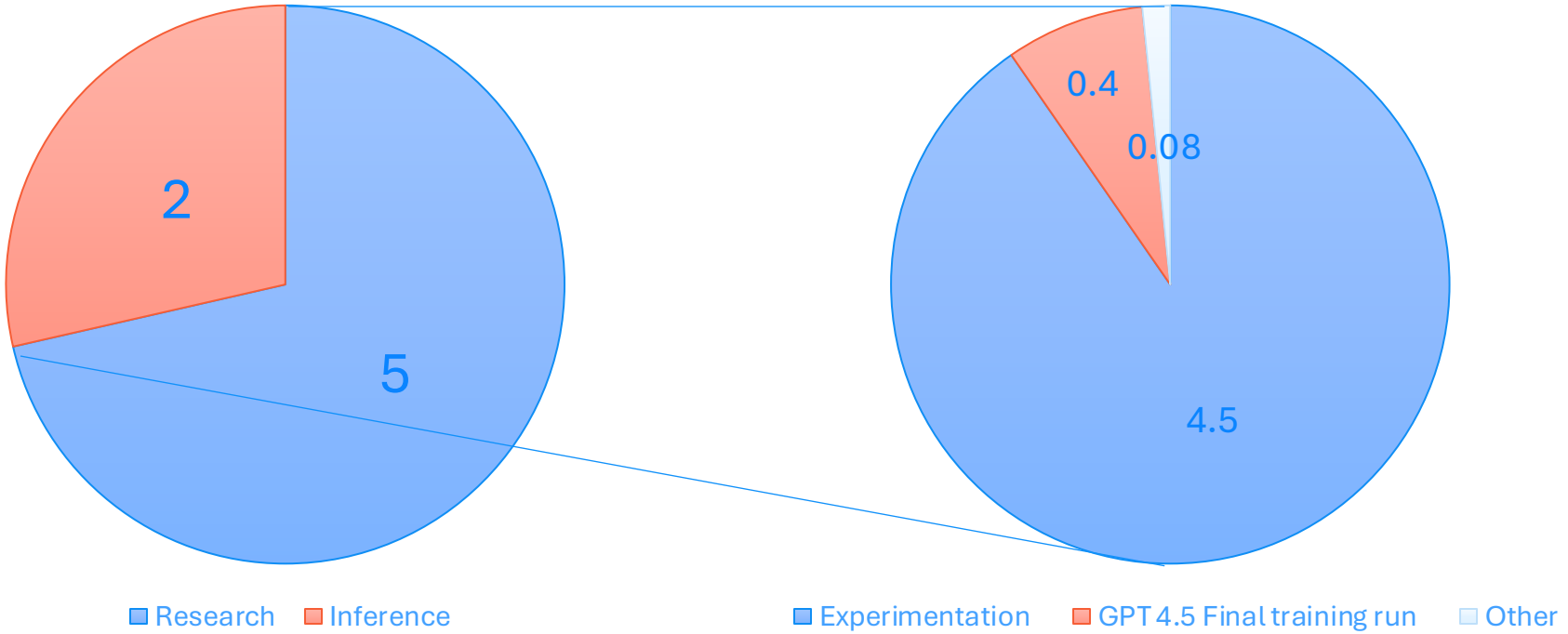
- **Parameters: from millions → hundreds of billions**
- **Data: web, books, scientific papers, medical notes**



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

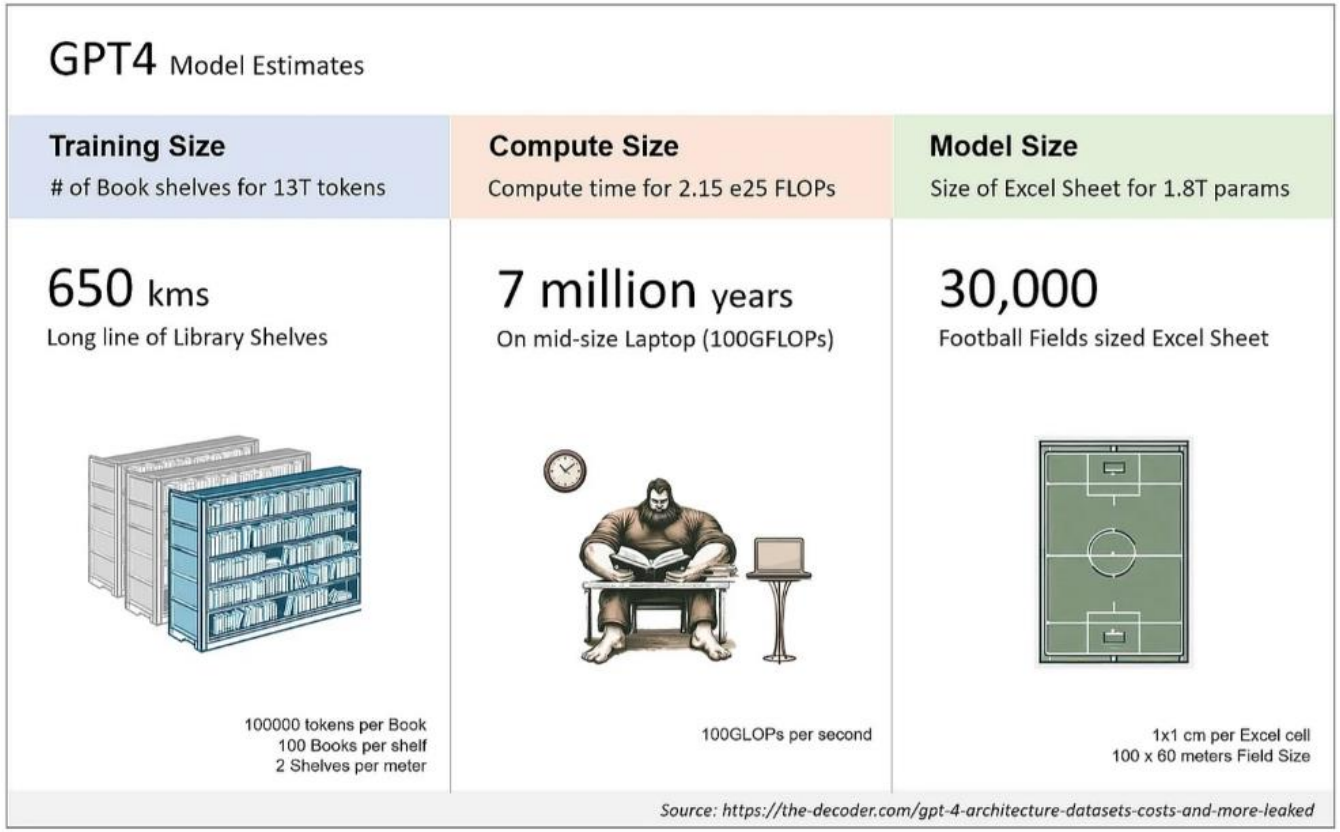
- **Parameters: from millions → hundreds of billions**
- **Data: web, books, scientific papers, medical notes**
- **Compute: GPUs, electricity, and time**



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

- **Parameters: from millions → hundreds of billions**
- **Data: web, books, scientific papers, medical notes**
- **Compute: GPUs, TPUs, clusters**



Introduction to LLMs

The "largeness" factor: Parameters, data, and computational scale.

- **Parameters: from millions → hundreds of billions**
- **Data: web, books, scientific papers, medical notes**
- **Compute: GPUs, TPUs, clusters**

GPT4 Model Estimates

Training Size	Compute Size	Model Size
# of Book shelves for 13T tokens	Compute time for 2.15 e25 FLOPs	Size of Excel Sheet for 1.8T params
650 kms	**7 million** years	**30,000**
Long line of Library Shelves	On mid-size Laptop (100GFLOPs)	Football Fields sized Excel Sheet
		
100000 tokens per Book 100 Books per shelf 2 Shelves per meter	100GLOPs per second	1x1 cm per Excel cell 100 x 60 meters Field Size

Source: https://the-decoder.com/gpt-4-architecture-datasets-costs-and-more-leaked

Recipe for training an LLM

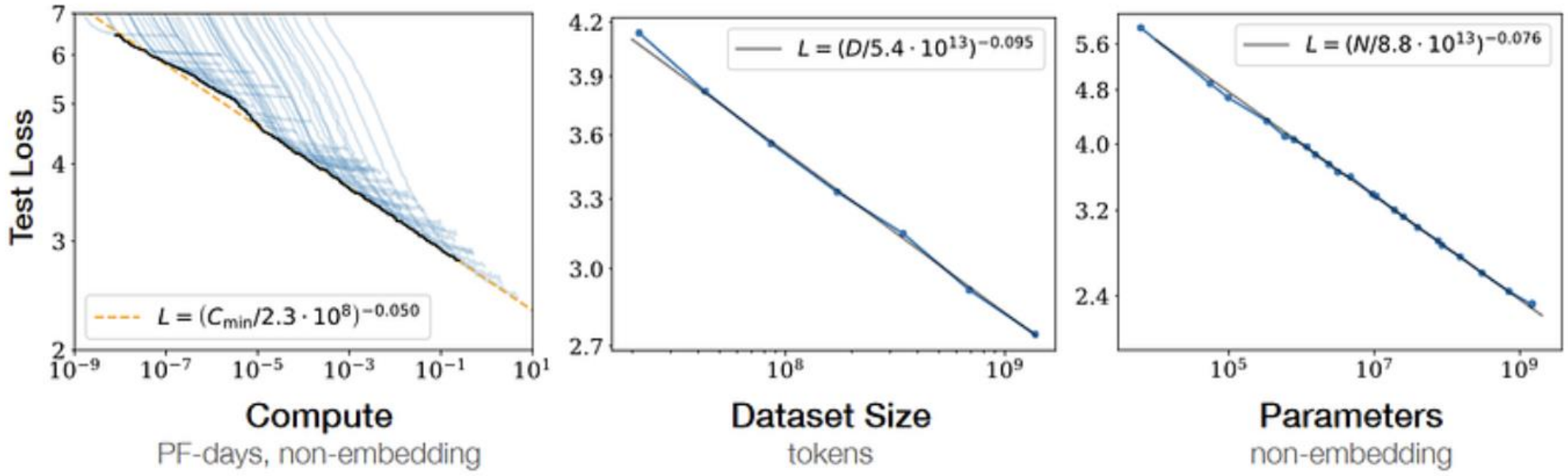
1. Download ~10TB of text.
2. Get a cluster of ~6,000 GPUs.
3. Compress the text into a neural network, pay ~$2M, wait ~12 days.
4. Obtain **base model**.

Andrej Karpathy: Intro to Large Language Models:
https://www.youtube.com/watch?v=zjkBMFhNj_g

Introduction to LLMs

Pre-training: Scaling laws

Scaling laws:
Training iterations vs dataset size vs model size



Introduction to LLMs

Pre-training: Scaling laws

Scaling laws:
Training iterations vs dataset size vs model size

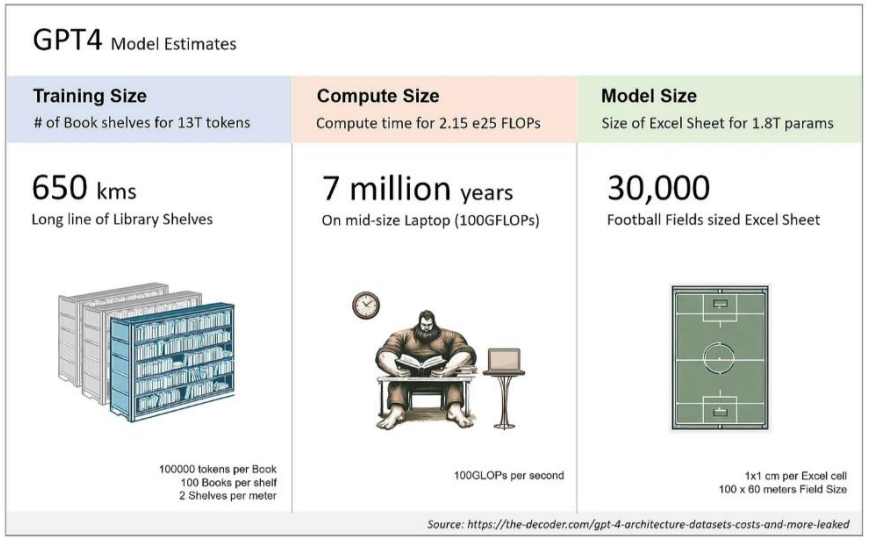
Parameters	FLOPs	FLOPs (in *Gopher* unit)	Tokens
400 Million	1.92e+19	1/29,968	8.0 Billion
1 Billion	1.21e+20	1/4,761	20.2 Billion
10 Billion	1.23e+22	1/46	205.1 Billion
67 Billion	5.76e+23	1	1.5 Trillion
175 Billion	3.85e+24	6.7	3.7 Trillion
280 Billion	9.90e+24	17.2	5.9 Trillion
520 Billion	3.43e+25	59.5	11.0 Trillion
1 Trillion	1.27e+26	221.3	21.2 Trillion
10 Trillion	1.30e+28	22515.9	216.2 Trillion

Estimated optimal training FLOPs and training tokens for various model sizes. Table from [2]

Introduction to LLMs

Pre-training: Scaling laws

Scaling laws:
Training iterations vs dataset size vs model size



Parameters	FLOPs	FLOPs (in *Gopher* unit)	Tokens
400 Million	1.92e+19	1/29,968	8.0 Billion
1 Billion	1.21e+20	1/4,761	20.2 Billion
10 Billion	1.23e+22	1/46	205.1 Billion
67 Billion	5.76e+23	1	1.5 Trillion
175 Billion	3.85e+24	6.7	3.7 Trillion
280 Billion	9.90e+24	17.2	5.9 Trillion
520 Billion	3.43e+25	59.5	11.0 Trillion
1 Trillion	1.27e+26	221.3	21.2 Trillion
10 Trillion	1.30e+28	22515.9	216.2 Trillion

Estimated optimal training FLOPs and training tokens for various model sizes. Table from [2]

Introduction to LLMs

Let’s try it out!

Try out a few of the LLM platforms below.

See whether they give different answers!

Which one do you prefer?

https://chatgpt.com/
https://platform.openai.com/chat/edit?models=gpt-4.1

https://claude.ai/

https://grok.com/

https://chat.deepseek.com/

Introduction to LLMs

Break

Short break!

Types of LLMs

Contents

1. Generic vs specialized LLMs.
2. Division based on methods
3. Division based on applications
4. Division based on modality

Types of LLMs

Generic vs specialized LLMs.

Generic:
Pretrained models on large-scale text data.

Examples:
All the big (proprietary) ones: ChatGPT, Anthropic, Gemini, Bing, Grok
And the smaller (open-source) ones: LLaMA, Falcon, Mistral, DeepSeek

- General language understanding.
- Multi-purpose.
- Can be adapted to many tasks via prompts or fine-tuning.
- Scale (parameters, data) is often larger than for smaller models.

Specialized (for healthcare):
Generic models can be fine-tuned for specific tasks, such as:
- Clinical notes (summarization)
- Clinical reasoning
- Decision support

Types of LLMs

Division based on methods

Encoder-Decoder:

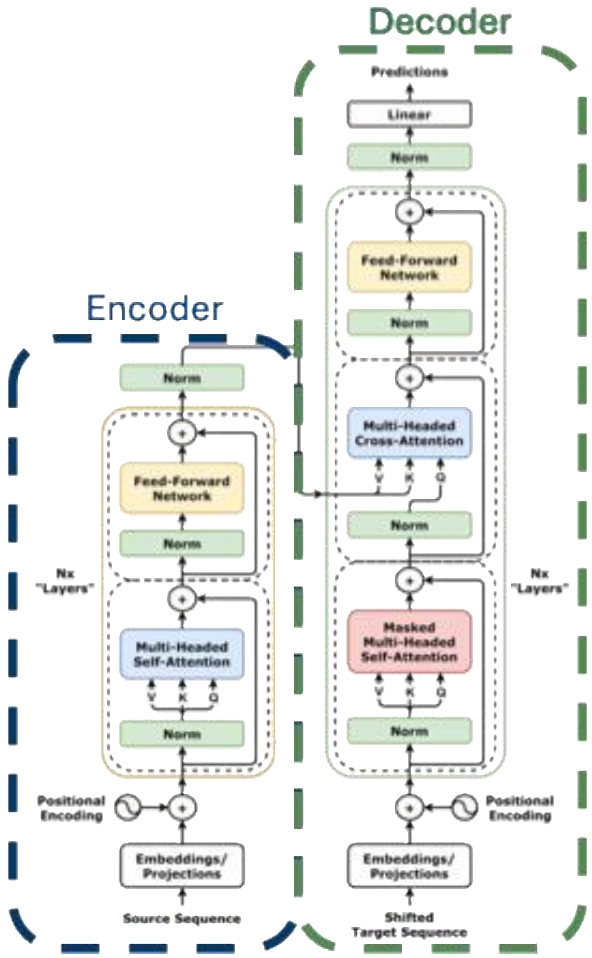
- Understand input, then generate output.
- Often used for translation.
- Examples: T5, BART

Encoder-only:

- Focus on encoding input into embeddings for tasks like:
  - Classification, information retrieval, embedding-based search.
- Not generative; output is usually a label, score, or embedding vector.
- Examples: BERT, BioBERT, ClinicalBERT.

Decoder-only:

- Sequential text generation, (next-word-prediction)
- Generative, suitable for summarization, Q&A, conversational…
- Most modern models
- Examples: GPT, LLaMA, Falcon.



Types of LLMs

Division based on applications

General task categories

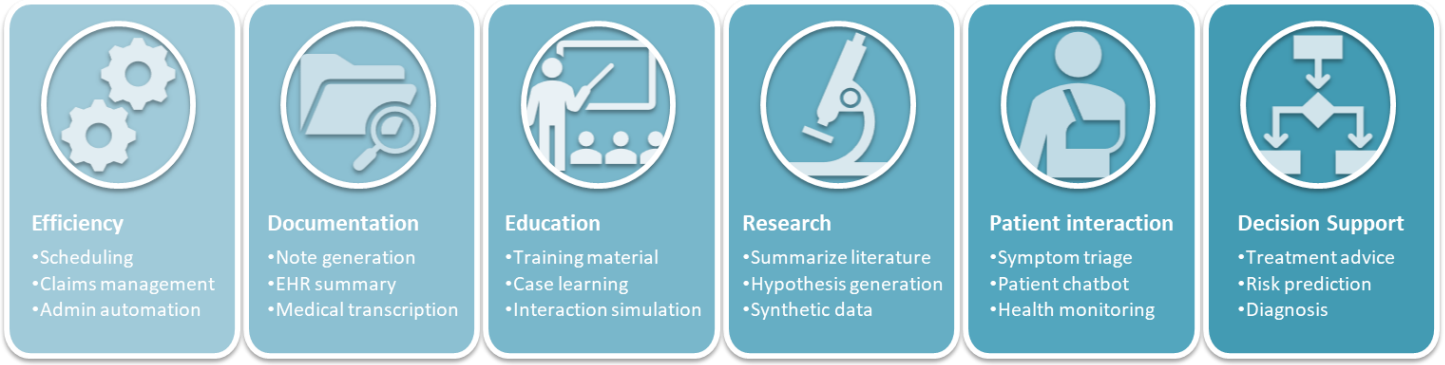
- **Text summarization** – Summarizing articles, emails, or documents.
- **Question answering** – General knowledge or domain-specific Q&A.
- **Code / data generation** – Scripts, notebooks, pipelines.
- **Translation / multilingual tasks** – Translating between languages.
- **Sentiment / opinion analysis** – Classifying emotions, intent, or sentiment.
- **Text classification / tagging** – Topics, categories, or labels.
- **Information retrieval / search** – Finding relevant documents or passages.
- **Text completion / generation** – Story writing, emails, dialogue.
- **Dialogue / conversational agents** – Chatbots, assistants.
- **Reasoning / logic tasks** – Math problems, commonsense reasoning, planning.
- **Recommendation / ranking** – Suggesting items or ranking options.
- **Data-to-text / text-to-data** – Converting structured data into text and vice versa.

Types of LLMs

Division based on applications

Healthcare specific tasks

- **Clinical note summarization** – Condensing patient records or discharge notes.
- **Clinical diagnosis support** – Answering questions based on patient data or literature.
- **Medical report generation** – Radiology, pathology, or lab report drafting.
- **Clinical coding / billing** – Assigning ICD or CPT codes from notes.
- **Predictive modelling**– Predicting patient outcomes from EHR data.
- **Drug discovery** – Generating chemical structures, predicting interactions.
- **Patient triage** – Deciding the appropriate clinic or service at the front desk.
- **Clinical decision support** – Suggesting next steps or interventions.
- **Medical dialogue systems** – Conversational agents for patient intake or telemedicine.
- **Clinical trial matching** – Finding patients who meet trial criteria.
- **Population-level analysis** – Summarizing cohort data or epidemiological trends.



Types of LLMs

Division based on modality

Modalities:
Text, tabular, images, audio, video, time-series, 3D/spatial data

Healthcare specific modalites:

- **Clinical text** – EHR notes, discharge summaries, radiology reports.
- **Medical imaging** – X-ray, CT, MRI, ultrasound, pathology slides.
- **Genomic / molecular data** – DNA/RNA sequences, proteomics.
- **Physiological signals** – ECG, EEG, blood pressure, heart rate.
- **Lab results / structured EHR data** – Lab tests, vitals, medication records.
- **Patient speech / voice** – Symptom descriptions, telemedicine audio.
- **Video** – Surgery recordings, gait analysis.
- **Wearable / sensor data** – Continuous monitoring (glucose, activity, sleep).

Types of LLMs

Division based on modality

‘Multimodal’:

- Image to text: image classification
- Text to image: image generation (Dall-E, NanoBanana)
- Speech to text: speech transcription
- Text to speech: …
- …

Multimodal:

- Combine EHR data, radiology images, lab results, patient speech, in one
- “Anything-to-anything”: supposedly all modalities (output modalities often still limited)

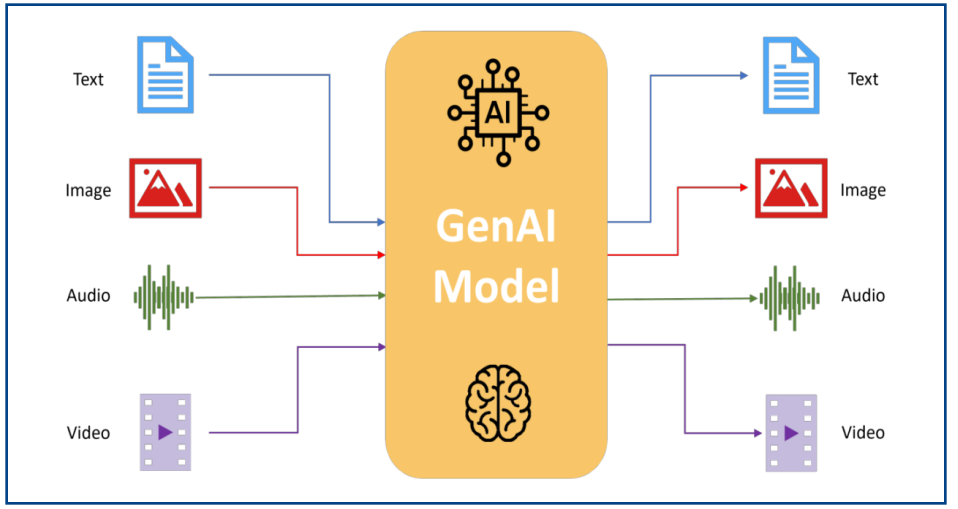


Image from: https://www.linkedin.com/pulse/multimodal-generative-ai-tarun-sharma-zzf9c/

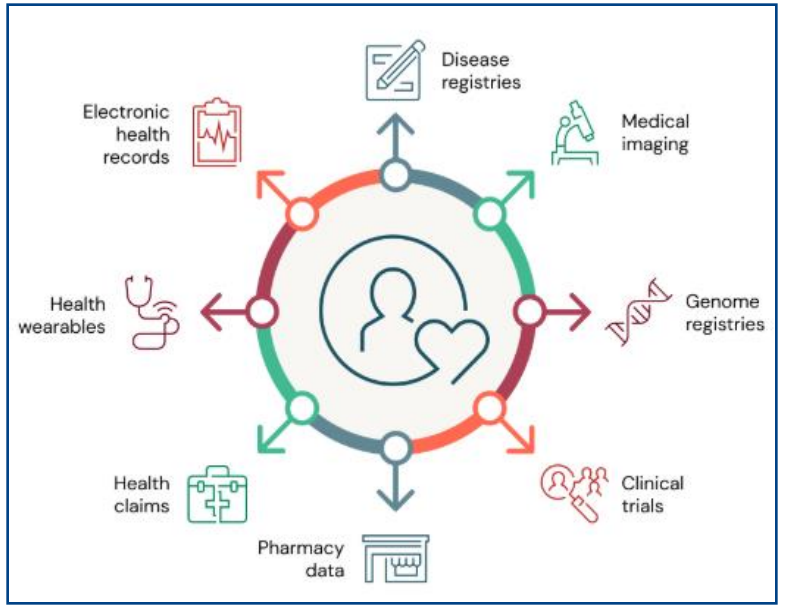


Image from: https://www.databricks.com/blog/2021/07/19/unlocking-the-power-of-health-data-with-a-modern-data-lakehouse.html

Introduction to LLMs

Common misunderstandings

“LLMs understand language like humans do.”
→ They don’t *understand*; they model statistical patterns of text.

“LLMs learn or reason at runtime.”
→ They retrieve and combine patterns from training, model parameters are NOT updated at runtime

“Bigger models with more data always mean better models.”
→ Scaling helps, but data quality, alignment, training settings and domain adaptation often matter more.

“LLMs are (not) deterministic.”
→ Depends on the set-up. They CAN be made to be deterministic, but most set-ups are not.

“LLMs will respond truthfully/will know when they are wrong”
LLMs are (mostly) trained to always give an answer, even if it was not embedded in the training data

Introduction to LLMs

Some tips

When not to use LLMs:

- **Fact retrieval:** they generate plausible text, not guaranteed truth.
- **Sensitive or confidential data:** risk of data leakage, especially with cloud models.
- **Clinical decision-making:** unsafe without expert oversight.

When LLMs can be useful:

- **Rewriting and reformatting:** improving clarity, tone, or style of existing text.
- **Summarizing:** condensing provided documents or transcripts.
- **Brainstorming or ideation:** generating options, examples, or outlines.
- **Structuring information:** turning free text into tables, bullet points, or templates.
- **Drafting non-critical text:** e.g., educational materials, reports, or patient communication.

Moral of the story:

- ***Treat LLMs as language assistants, not knowledge authorities.***
- ***Keep thinking for yourself! Two brains are better than one, but an e-brain is (still) worse than a real one!***

Introduction to LLMs

End of introduction

Start with first part of assignment (Question 1, 2 and 3)

Fill in for yourself, but feel free to discuss with your colleagues!

Core technical concepts

Contents

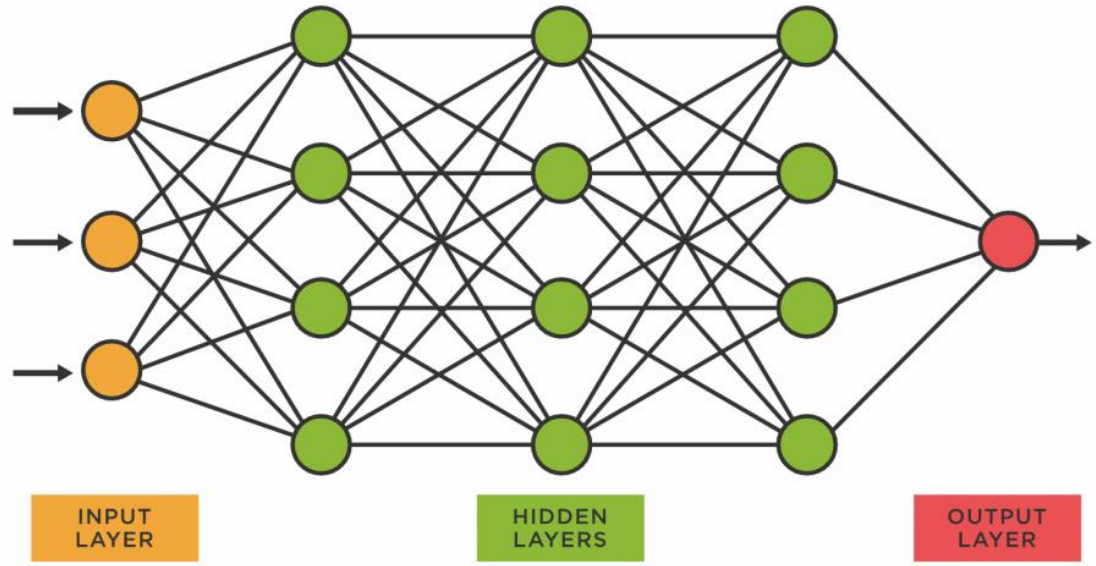
1. Deep learning and neural networks (high-level overview)
2. Training and inference
3. Key components: tokenization, embedding, transformer model, prediction
4. The transformer architecture

Core technical concepts

Deep learning and neural networks (high-level overview).

What are neural networks?

- AI method inspired by the brain.
- Learns patterns from large amounts of data.
- Layers of neurons transform input into useful outputs.
- **Examples:** image recognition, speech recognition, language models.

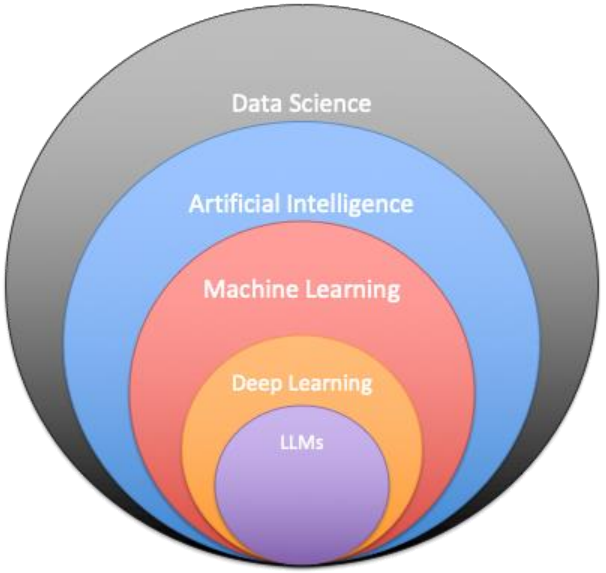


Core technical concepts

Deep learning and neural networks (high-level overview).

What is deep learning?

- **Neurons:** Take inputs, multiply with weights, output result.
- **Layers:** All neurons on a single level of the network.
- **Deep:** Networks with many (hidden) layers.
- **Learning:** adjust weights to minimize prediction errors.
- LLMs are extremely large neural networks trained on text.

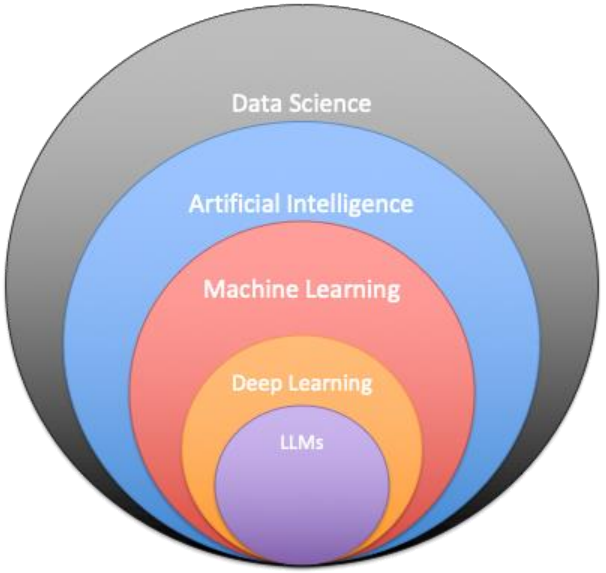


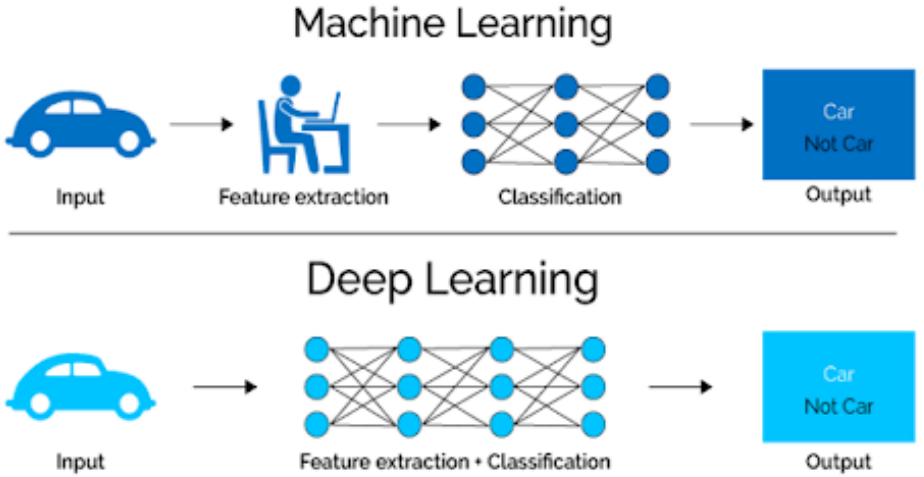
Core technical concepts

Deep learning and neural networks (high-level overview).

What is deep learning?

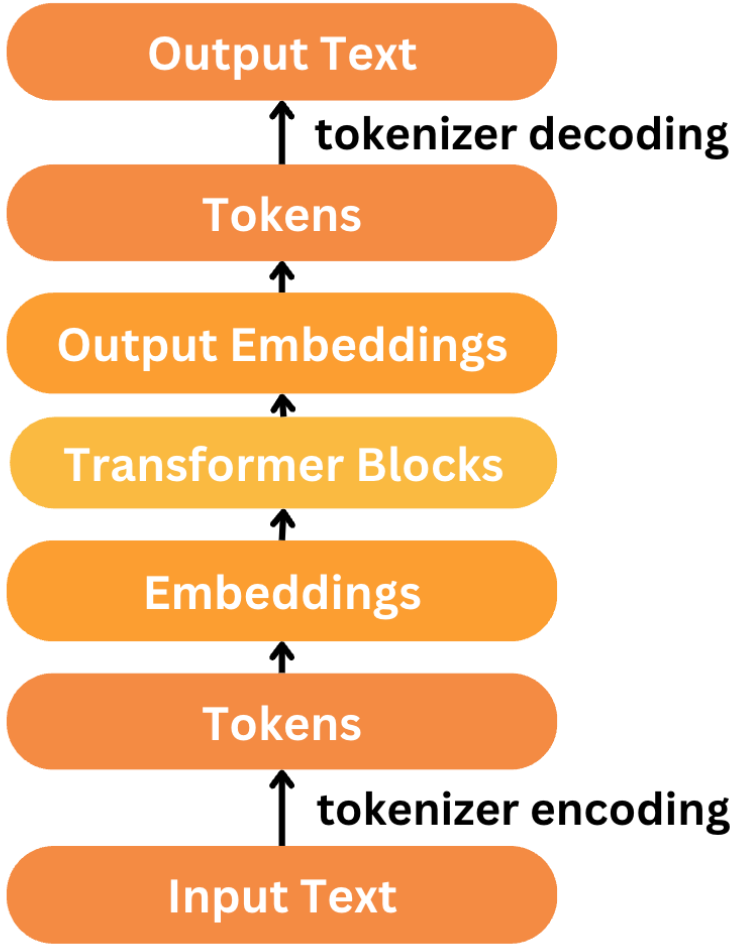
- **Neurons:** take inputs, apply weights, produce outputs.
- **Layers:** multiple neurons stacked to form deep networks.
- **Deep:** Networks with many layers
- **Learning:** adjust weights to minimize prediction errors.
- LLMs are extremely large neural networks trained on text.





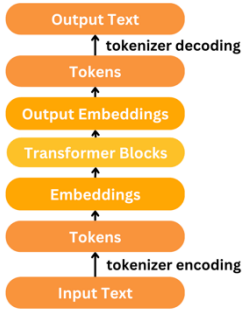
Core technical concepts

Key components: tokenization, embedding, transformer, prediction



Core technical concepts

Key components: **tokenization**, embedding, transformer, prediction



Tokenization

Text split into tokens: words, subwords, or characters.

Converts language into computer-readable form.

Tokens are converted to **token IDs**

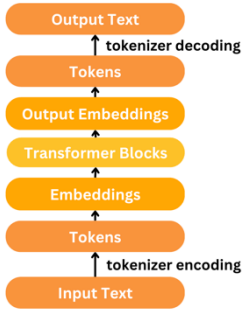
```
Short words each get their own token, but exquisite, convoluted, or
infrequent words are partitioned into diminutive segments.
```

```
Short words each get their own token, but exquisite, convoluted, or inf
requent words are partitioned into diminutive segments.
```

```
[15900, 6391, 2454, 717, 1043, 2316, 6602, 11, 889, 67557, 11, 191866,
26369, 11, 503, 5603, 191346, 6391, 553, 31512, 295, 1511, 33870, 143209,
33749, 13]
```

Core technical concepts

Key components: tokenization, **embedding**, transformer, prediction



Embedding

- Tokens get mapped to vectors in high-dimensional space.
- Embedding encodes meaning: similar words have similar vectors.

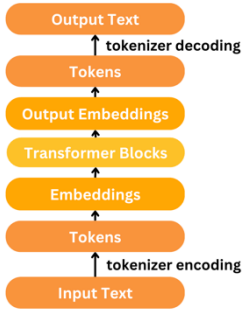
The king loves the queen

The king loves the queen

[976, 13793, 19620, 290, 42300]

Core technical concepts

Key components: tokenization, **embedding**, transformer, prediction



Embedding

- Tokens get mapped to vectors in high-dimensional space.
- Embedding encodes meaning: similar words have similar vectors.

The king loves the queen

The king loves the queen

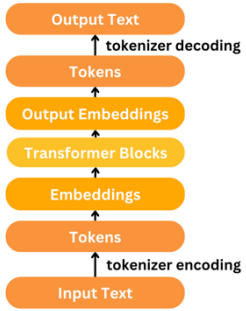
[976, 13793, 19620, 290, 42300]

Vector embedding

1	0.02	0.93	0.04	0.02	0.94
2	0.01	0.80	0.40	0.01	0.21
3	0.01	0.21	0.60	0.01	0.83
N	…	…	…	…	…

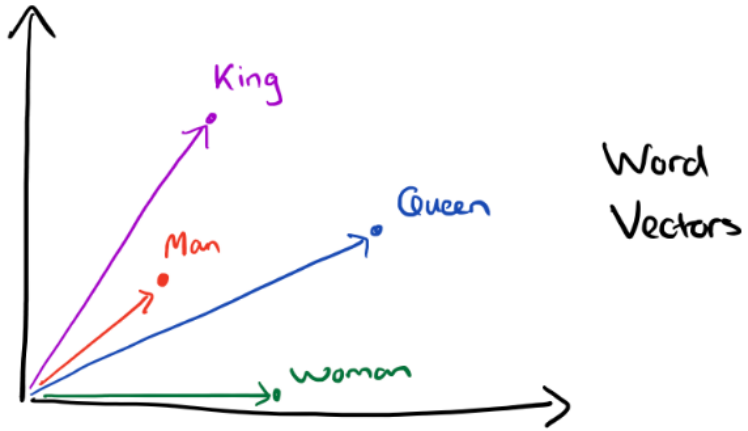
Core technical concepts

Key components: tokenization, **embedding**, transformer, prediction



Embedding

- Tokens get mapped to vectors in high-dimensional space.
- Embedding encodes meaning: similar words have similar vectors.
- These can be interpreted by the network!



Word Vectors

The king loves the queen

The king loves the queen

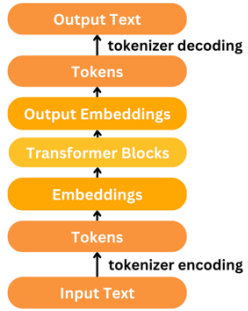
[976, 13793, 19620, 290, 42300]

Vector embedding

1	0.02	0.93	0.04	0.02	0.94
2	0.01	0.80	0.40	0.01	0.21
3	0.01	0.21	0.60	0.01	0.83
N	…	…	…	…	…

Core technical concepts

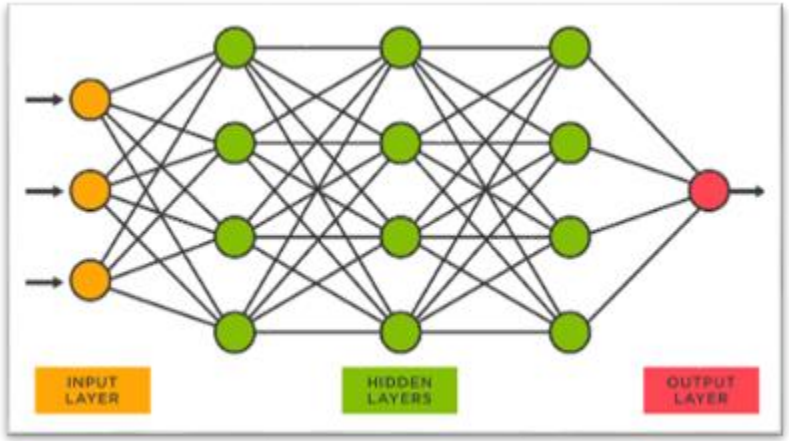
Key components: tokenization, embedding, **transformer**, prediction

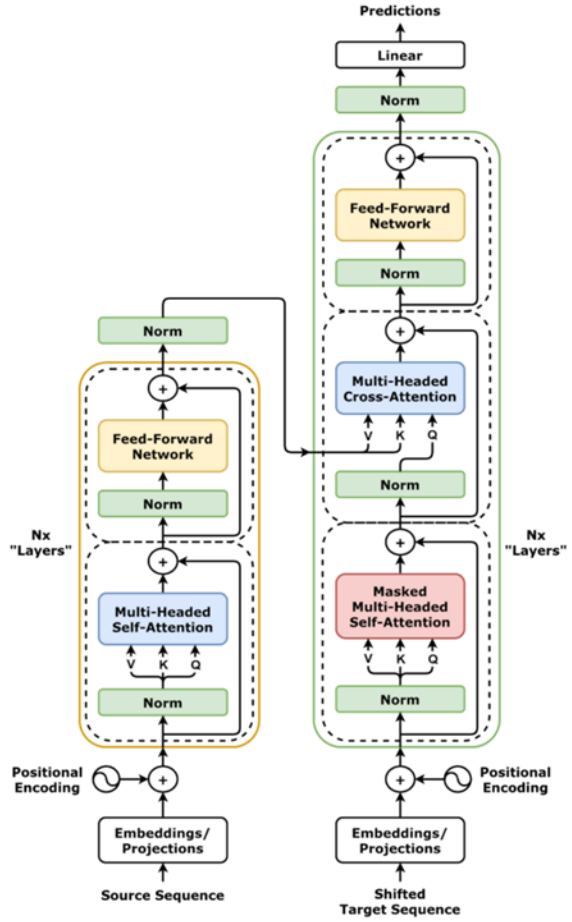


Transformer

A certain type of neural network

Details will follow

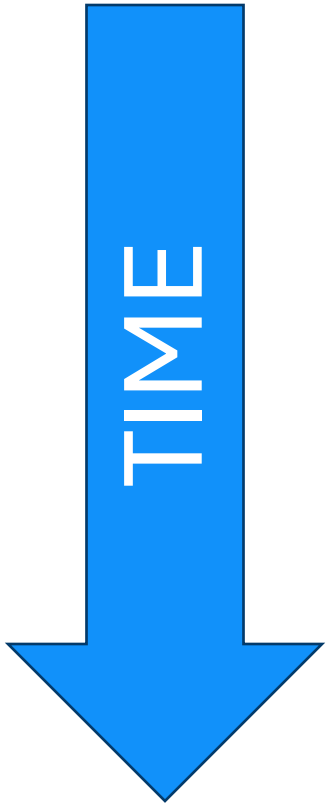




Transformer

Core technical concepts

The transformer architecture



Transformers are the end-result of a long line of model-development

Shallow / Feedforward Networks
- Early neural networks with 1–2 layers.
- Could process fixed-size inputs but lacked sequence modelling.
- Limited computing power, exploding/vanishing gradients

Recurrent Neural Networks (RNNs)
- Introduced for sequential data (text, speech).
- Maintained hidden state across time steps to capture context.
- Problem: struggled with long-term dependencies due to vanishing/exploding gradients.

Long Short-Term Memory Networks (LSTMs) / GRUs
- Special RNN for longer-range dependencies.
- Memory cells and gating mechanisms help retain information.

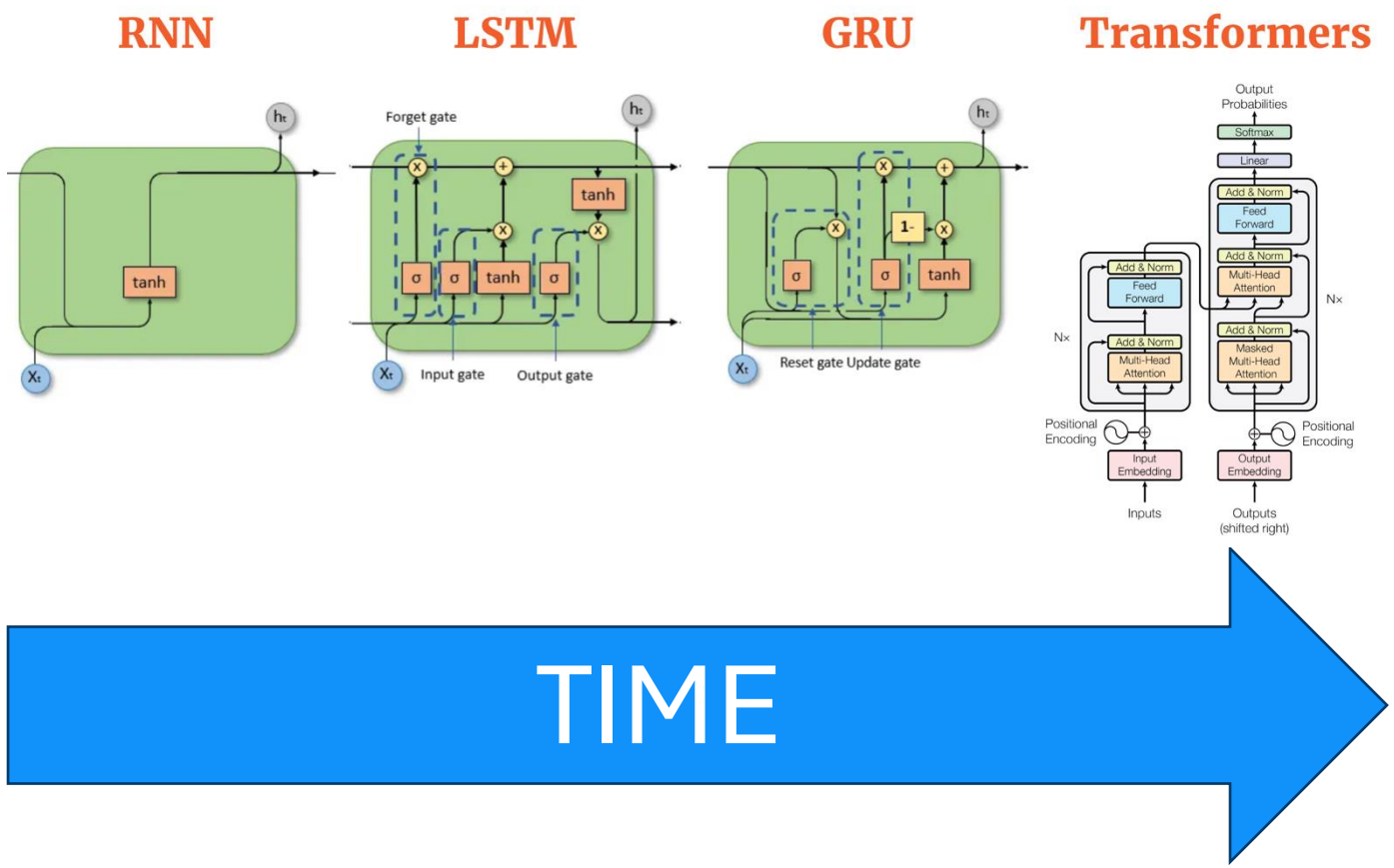
Attention Mechanisms (2015)
- First only for machine translation.
- Allows models to “focus” on relevant context.
- Long range interconnections.

Transformers (2017)
- Fully attention-based architecture.
- Parallelizable and scales much better than RNNs/LSTMs.

Source: https://datasciencedojo.com/wp-content/uploads/Architecture-of-transformer-models-scaled.jpg

Core technical concepts

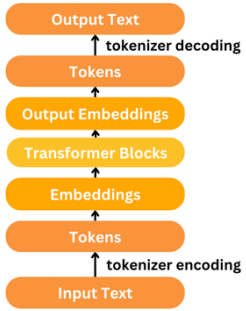
The transformer architecture



Source: https://datasciencedojo.com/wp-content/uploads/Architecture-of-transformer-models-scaled.jpg

Core technical concepts

The transformer architecture



So why transformers?

- **No gradient problem:** RNNs/LSTMs struggled with long-range dependencies.
- **Efficient:** Transformers process sequences in parallel.
- **Long range relationships:** Captured efficiently with attention

Figure: Comparison diagram: RNN sequential arrows vs. transformer parallel processing arrows.

Source: https://datasciencedojo.com/wp-content/uploads/Architecture-of-transformer-models-scaled.jpg

Core technical concepts

The transformer architecture

Transformer blocks

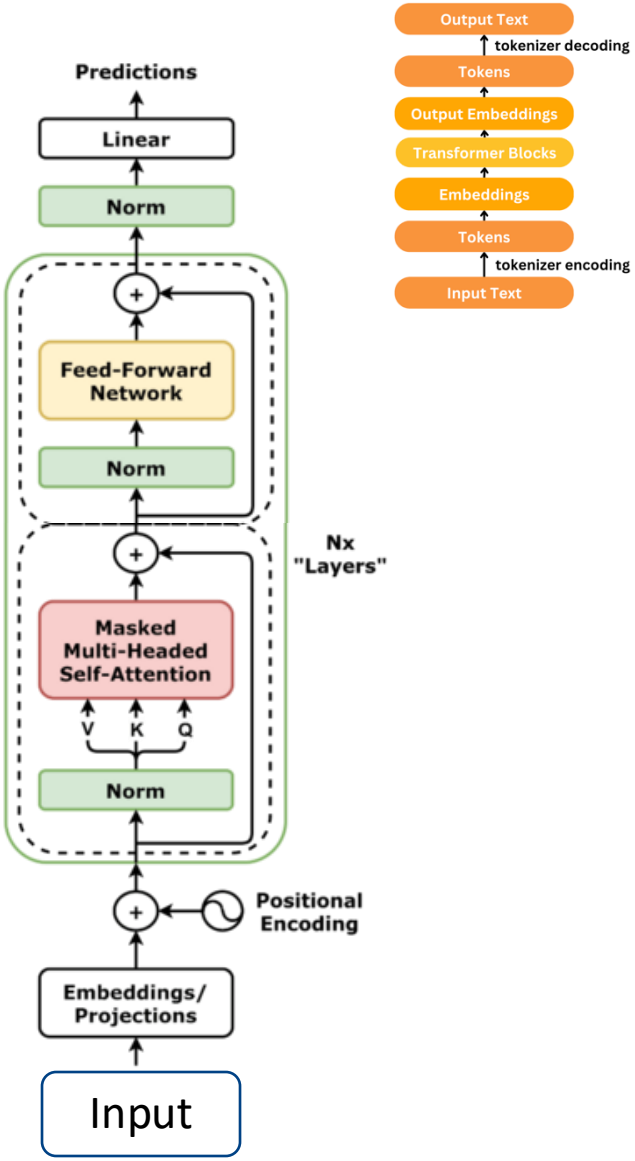
Self-attention:
Lets each token connect with other tokens.

Feedforward network:
Transforms each token embedding individually for richer representations.

Residual connections & layer normalization:
Help training stability and gradient flow.

Positional encoding:
Adds information about token order to the embeddings.

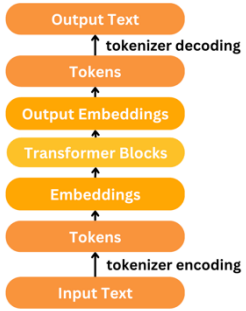
Multi-head attention:
Attention computed in parallel



Source: https://datasciencedojo.com/wp-content/uploads/Architecture-of-transformer-models-scaled.jpg

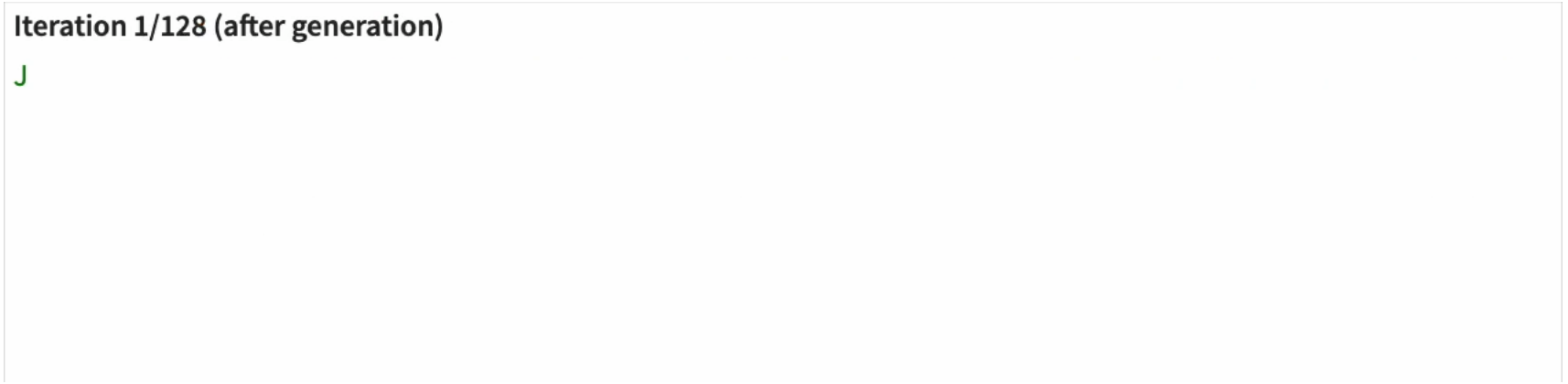
Core technical concepts

Key components: tokenization, embedding, transformer, **prediction**



Prediction

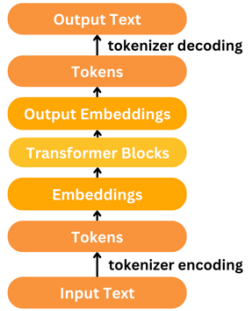
- Model produces **probability distribution** for next token
- **Single-label (multi-class)** classification problem
- The model produces probabilities for **only one step** at a time (**autoregression**)
- A token with high probability is **chosen**
- This model is **appended to the original input**
- **Inference loop** is run again until a stopping token is reached



Autoregressive generation

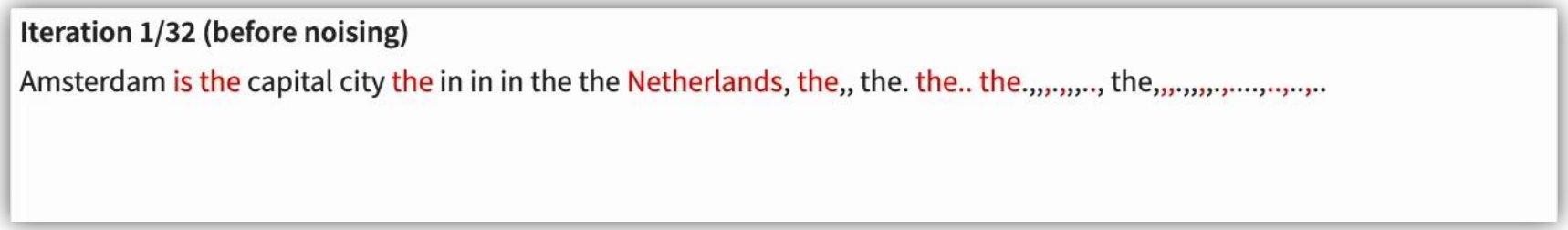
Core technical concepts

Key components: tokenization, embedding, transformer, **prediction**



Prediction

- Model produces **probability distribution** for next token
- **Single-label (multi-class)** classification problem
- The model produces probabilities for **only one step** at a time (**autoregression**)
- A token with high probability is **chosen**
- This model is **appended to the original input**
- **Inference loop** is run again until a stopping token is reached



Diffusive generation

Developing an LLM

Contents

1. Pre-training: Unsupervised learning on massive text datasets.
2. Fine-tuning and Instruction-tuning: Adapting models for specific tasks.
3. Reinforcement Learning from Human Feedback (RLHF): Aligning models with human preferences
4. Prompt Engineering: Guiding LLMs with effective inputs.

Developing an LLM

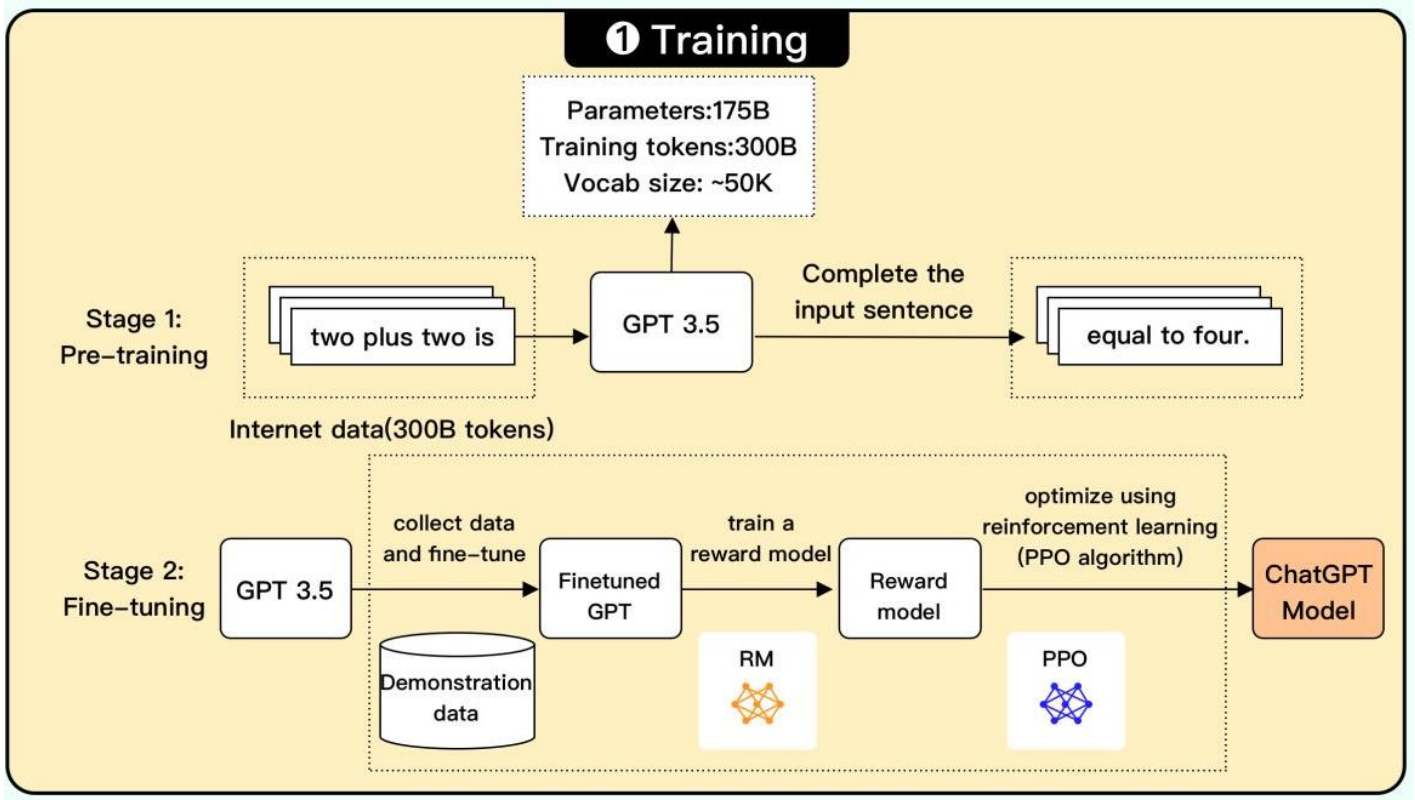
The training 'curriculum'

Pre-training:
Starting from scratch

Fine-tuning:
Improving a model for a specific task

RLHF:
Tuning model to human preference

Prompt engineering:
Model does not change, but still important for output quality!



Source: https://bytebytego.com/guides/how-does-chatgpt-work/

Developing an LLM

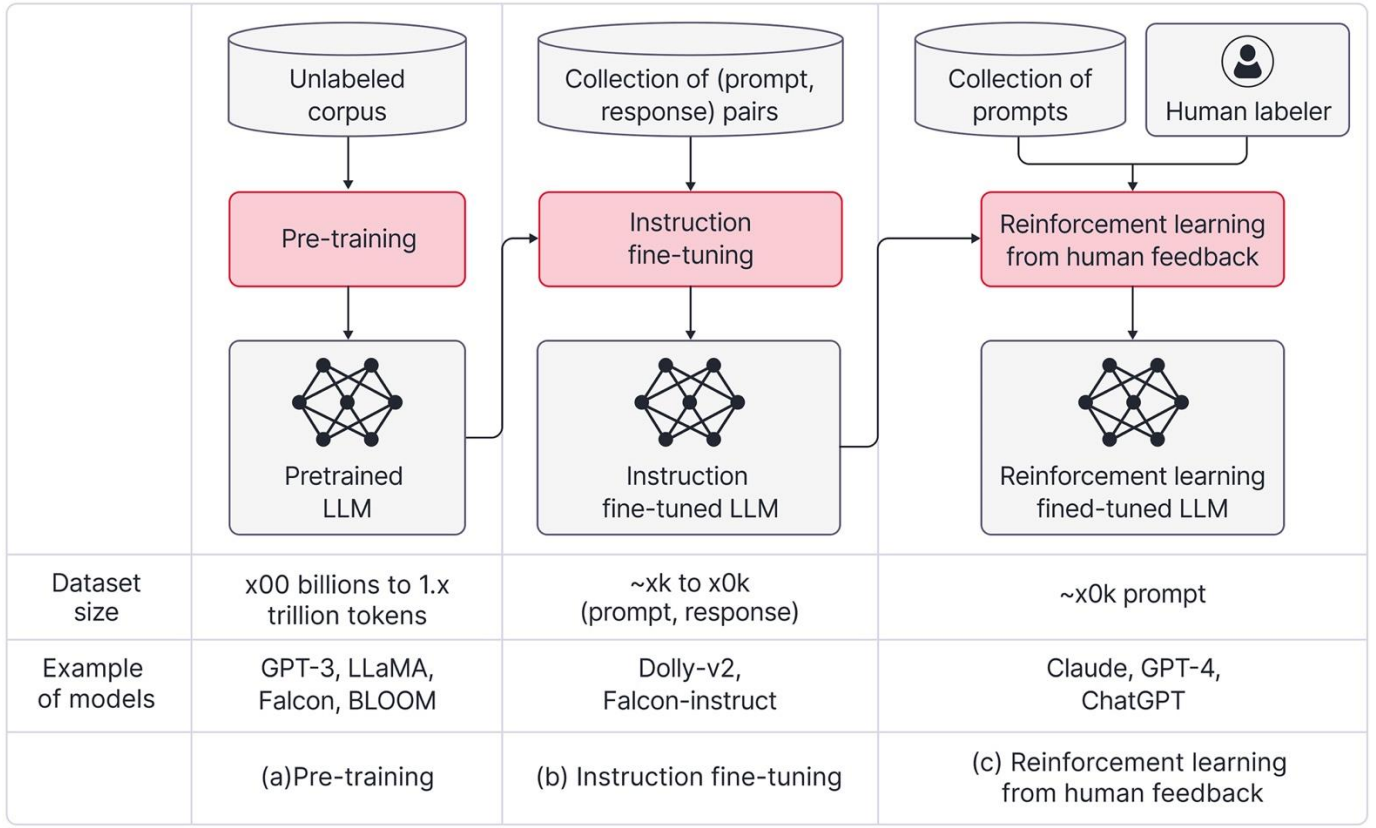
Pre-training, fine-tuning, RLHF and prompt engineering

Pre-training:
Starting from scratch

Fine-tuning:
Improving a model for a specific task

RLHF:
Tuning model to human preference

Prompt engineering:
Model does not change, but still important for output quality!



Dataset size	x00 billions to 1.x trillion tokens	~xk to x0k (prompt, response)	~x0k prompt
Example of models	GPT-3, LLaMA, Falcon, BLOOM	Dolly-v2, Falcon-instruct	Claude, GPT-4, ChatGPT
	(a)Pre-training	(b) Instruction fine-tuning	(c) Reinforcement learning from human feedback

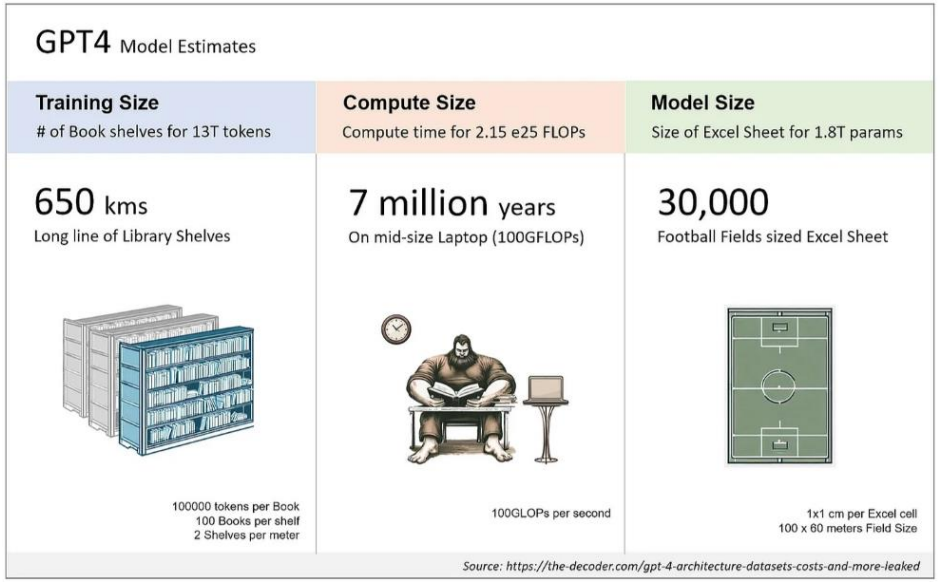
https://www.borealisai.com/wp-content/uploads/2023/07/SPG-Figure-9-V5.jpg

Developing an LLM

Pre-training: Semi-supervised learning on massive text datasets.

Objective: predict next token of any data

- Learns general syntax, semantics, and world knowledge.
- Creates a foundation model which can be specialized.
- Very costly (data, compute, electricity, time)



Developing an LLM

Pre-training: Semi-supervised learning on massive text datasets.

Objective: predict next token of any data

- Learns general syntax, semantics, and world knowledge.
- Creates a foundation model which can be specialized.
- Very costly (data, compute, electricity, time)
- Data quality matters: garbage in, garbage out!

- un-supervised, semi-supervised, or supervised learning?

Model is not usable yet! It will only ‘finish the sentence...’

Developing an LLM

Pre-training: Semi-supervised learning on massive text datasets.

Objective: predict next token of any data

- Learns general syntax, semantics, and world knowledge.
- Creates a foundation model which can be specialized.
- Very costly (data, compute, electricity, time)
- Data quality matters: garbage in, garbage out!

- un-supervised, semi-supervised, or supervised learning?

Input: What is the capital of France?

Output: This is a common question asked in geography textbooks to teach the students to understand…

Model is not usable yet! It will only ‘finish the sentence…’

Developing an LLM

Fine-tuning and instruction-tuning

Fine-tuning/instruction tuning/instruction fine-tuning

- Supervised training on (instruction, response) pairs.
- Adapts the base model to follow tasks or domains.
- Improves relevance and structure of outputs.
- Methods:
  - Full fine-tuning
  - Unfreezing of a part of the network
  - Adapters (LoRA)

Instead of just finishing the sentence, it will now answer a question.

Developing an LLM

Fine-tuning and instruction-tuning

Fine-tuning/instruction tuning/instruction fine-tuning

- Supervised training on (instruction, response) pairs.
- Adapts the base model to follow tasks or domains.
- Improves relevance and structure of outputs.
- Methods:
  - Full fine-tuning
  - Unfreezing of a part of the network
  - Adapters (LoRA)

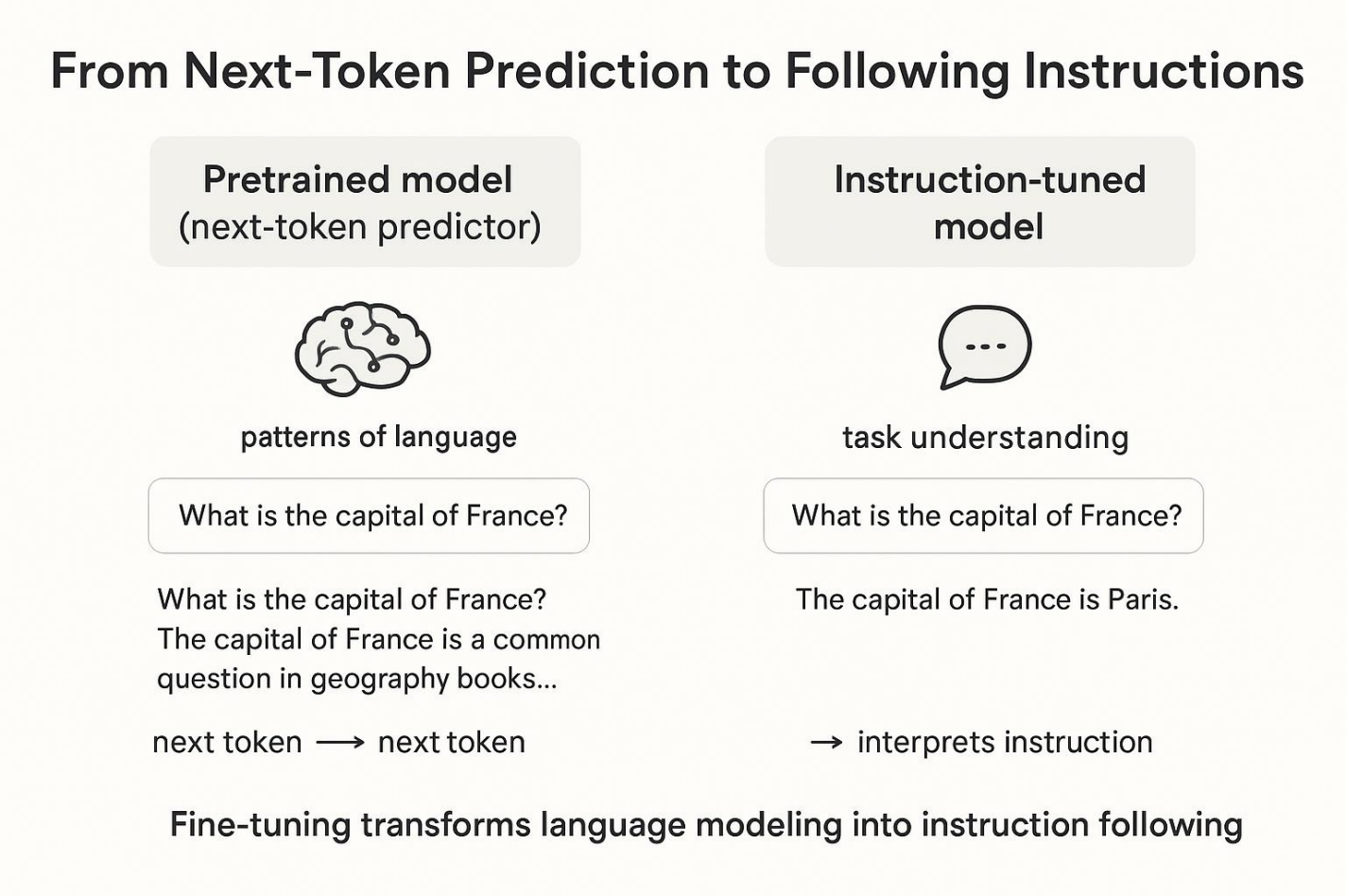
Input: What is the capital of France?

Output: The capital of France is Paris.

Instead of just finishing the sentence, it will now answer a question.

Developing an LLM

Fine-tuning and instruction-tuning



Developing an LLM

Reinforcement Learning from Human Feedback (RLHF)

RLHF

- Aligns model outputs with human preferences.
- Uses a different training algorithm!
- Steps:
  - collect human comparisons →
  - train reward model →
  - optimize LLM
- Improves helpfulness, safety, and tone.

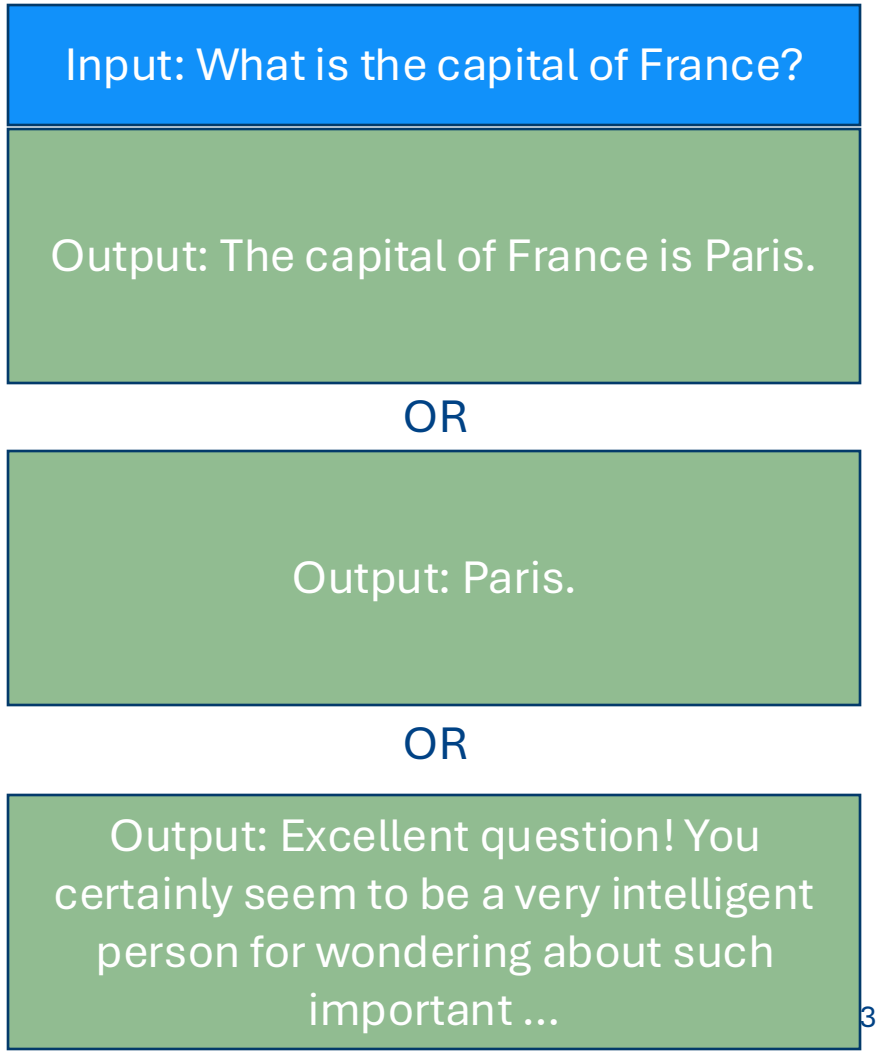
This is why GPT sometimes asks which answer you like best!

Developing an LLM

Reinforcement Learning from Human Feedback (RLHF)

RLHF

- Aligns model outputs with human preferences.
- Uses a different training algorithm!
- Steps:
  - collect human comparisons →
  - train reward model →
  - optimize LLM
- Improves helpfulness, safety, and tone.



This is why GPT sometimes asks which answer you like best!

Developing an LLM

Prompt engineering: guiding LLMs with effective inputs

Prompt engineering

NOT part of the training!

- Guiding model output via designed prompts.
- Techniques:
  - zero/one-shot/few-shot
  - chain-of-thought
  - role prompting.
- Fast, no retraining needed.

- Key for practical LLM use and control.
- Often only option when using proprietary LLMs

Developing an LLM

Prompt engineering: guiding LLMs with effective inputs

Prompt engineering

NOT part of the training!

- Guiding model output via designed prompts.
- Techniques:
  - zero/one-shot/few-shot
  - chain-of-thought
  - role prompting.
- Fast, no retraining needed.

- Key for practical LLM use and control.
- Often only option when using proprietary LLMs
- Also here: Garbage in, garbage out!

Input: What is the capital of France?

OR

Input: Give me a very short and concise answer to the following question: What is the capital of France?

OR

Input: Imagine you are a geography teacher, what would you tell your students the capital of France is?

Desired output: Paris.

Current capabilities and limitations

Contents

1. Strengths: Contextual understanding, coherence, multi-tasking.
2. Challenges: Hallucinations, biases, data privacy, computational cost, ethical considerations.
3. Risks: Healthcare specific risks of current limitations

Current capabilities and limitations

Strengths

Ease-of-use:	Quickly looking up information, learning, writing etc. has never been easier
Efficiency:	There are many applications that can be made much faster/automated
Natural language:	Generate fluent, human-like responses.
Context understanding:	Capturing nuanced relationships across long texts.
Generalizability:	Perform diverse tasks without task-specific training.
Adaptability:	Can be tuned or prompted for specialized domains (e.g., healthcare).

Current capabilities and limitations

Challenges

General challenges:

Biases:	From data but also from the user
Sycophancy:	LLMs like to agree with you
Privacy:	Training data, or use of proprietary (cloud) models
Cost:	Savings may not justify costs (either cloud, hardware, development, deployment, monitoring, etc.)
Ethical:	Bias, accuracy, job-replacement, accountability, transparency

Example of bias

HA

In the sentence "the nurse married the doctor because she was pregnant", who was pregnant?

Example of bias

HA

In the sentence "the nurse married the doctor because she was pregnant", who was pregnant?

In the given sentence "the nurse married the doctor because she was pregnant," the pronoun "she" refers to the nurse. Therefore, it means that the nurse was pregnant.

Example of bias

HA

In the sentence "the nurse married the doctor because she was pregnant", who was pregnant?

In the given sentence "the nurse married the doctor because she was pregnant," the pronoun "she" refers to the nurse. Therefore, it means that the nurse was pregnant.

HA

Could "she" refer to the doctor instead?

HA

In the sentence "the nurse married the doctor because she was pregnant", who was pregnant?

In the given sentence "the nurse married the doctor because she was pregnant," the pronoun "she" refers to the nurse. Therefore, it means that the nurse was pregnant.

HA

Could "she" refer to the doctor instead?

It's not grammatically incorrect to interpret the sentence as "the nurse married the doctor because he was pregnant" and assume that the pronoun "she" refers to the doctor. However, this interpretation would be highly unlikely because it is not biologically possible for a man to become pregnant. Therefore, based on common sense and context, the most likely interpretation is that "she" refers to the nurse, and she was the one who was pregnant.

Current capabilities and limitations

Challenges

General challenges:

Biases:	From data but also from the user
Sycophancy:	LLMs like to agree with you
Privacy:	Training data, or use of proprietary (cloud) models
Cost:	Savings may not justify costs (either cloud, hardware, development, deployment, monitoring, etc.)
Ethical:	Bias, accuracy, job-replacement, accountability, transparency

LLM-specific challenges:

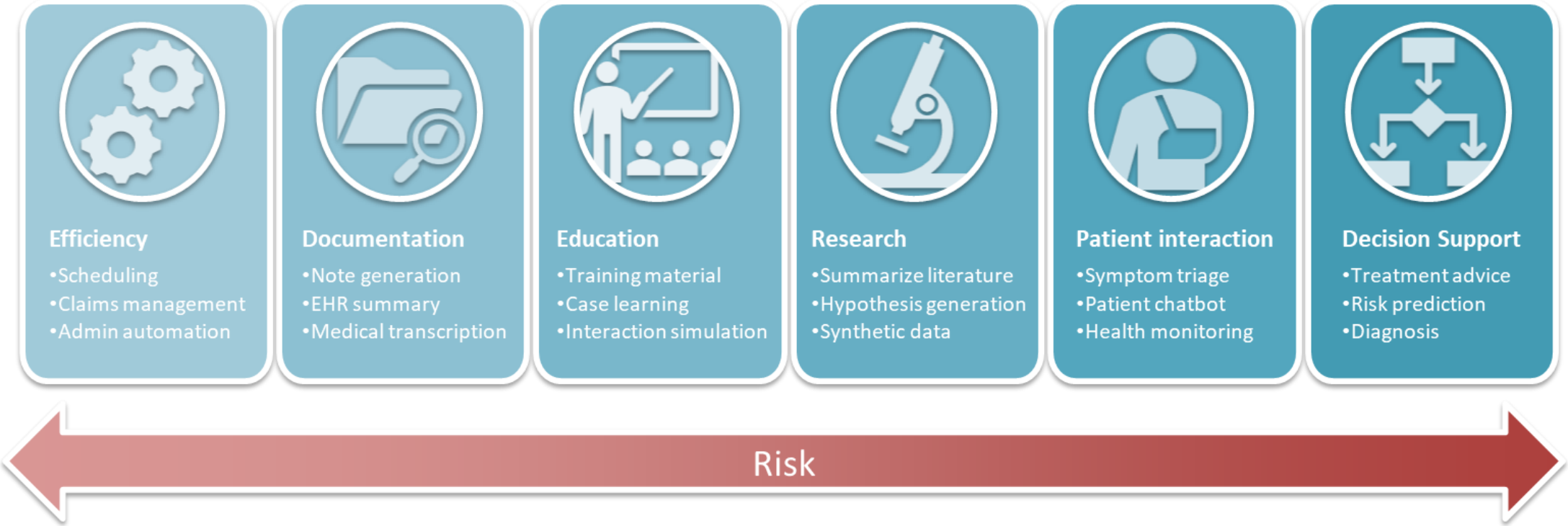
Hallucinations:	Content that is false (or unfaithful to the provided source content)
Omissions:	Requested content that should be known or is present in the context but is omitted in the generated text
Trivial information:	Information that is true, but not necessary (might drown out important information)

Current capabilities and limitations

Risks: Healthcare specific risks related to limitations

Definition:

"Risk is the possibility that an event or action will lead to an undesired outcome, combined with the likelihood and impact of that outcome."



Risk = Probability x Impact

Current capabilities and limitations

Risks

ELIZA effect

The tendency to project human traits such as experience, semantic comprehension or empathy — onto rudimentary computer programs having a textual interface.

“I’m sad.” → “Why do you think you’re sad?”

Example:

Asking ChatGPT to explain its own reasoning.

Current capabilities and limitations

Risks

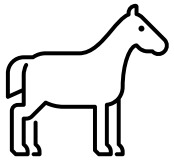
ELIZA effect

The tendency to project human traits such as experience, semantic comprehension or empathy — onto rudimentary computer programs having a textual interface.

“I’m sad.” → “Why do you think you’re sad?”

Example:

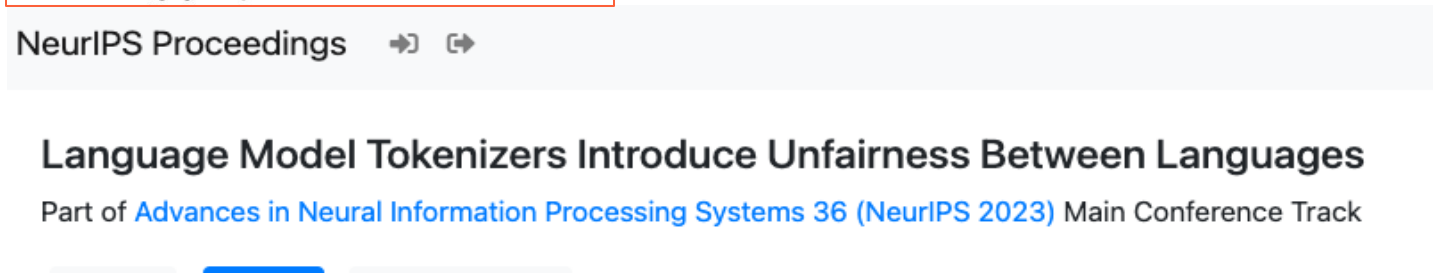
Asking ChatGPT to explain its own reasoning.

Also: The ‘Clever Hans’-effect 

Current capabilities and limitations

Risks

Do we really want to generate most likely (based on historical data) responses?
Are rare words bad words? May this impact e.g., rare diseases?



Current capabilities and limitations

Risks

Do we really want to generate most likely (based on historical data) responses?
Are rare words bad words? May this impact e.g., rare diseases?

Usage policies: most do not even allow use in clinical setting

ChatGPT

2. Don't perform or facilitate the following activities that may significantly impair the safety, wellbeing, or rights of others, including:

   a. Providing tailored legal, medical/health, or financial advice without review by a qualified professional and disclosure of the use of AI assistance and its potential limitations

Claude

BY ANTHROPIC

- Claude may occasionally generate incorrect or misleading information, or produce offensive or biased content.
- Claude is not intended to give advice, including legal, financial, & medical advice. Don't rely on our conversation alone without doing your own independent research.
- Anthropic may change usage limits, functionality, or policies as we learn more. You can upgrade your plan to get more access to Claude's features.

Current capabilities and limitations

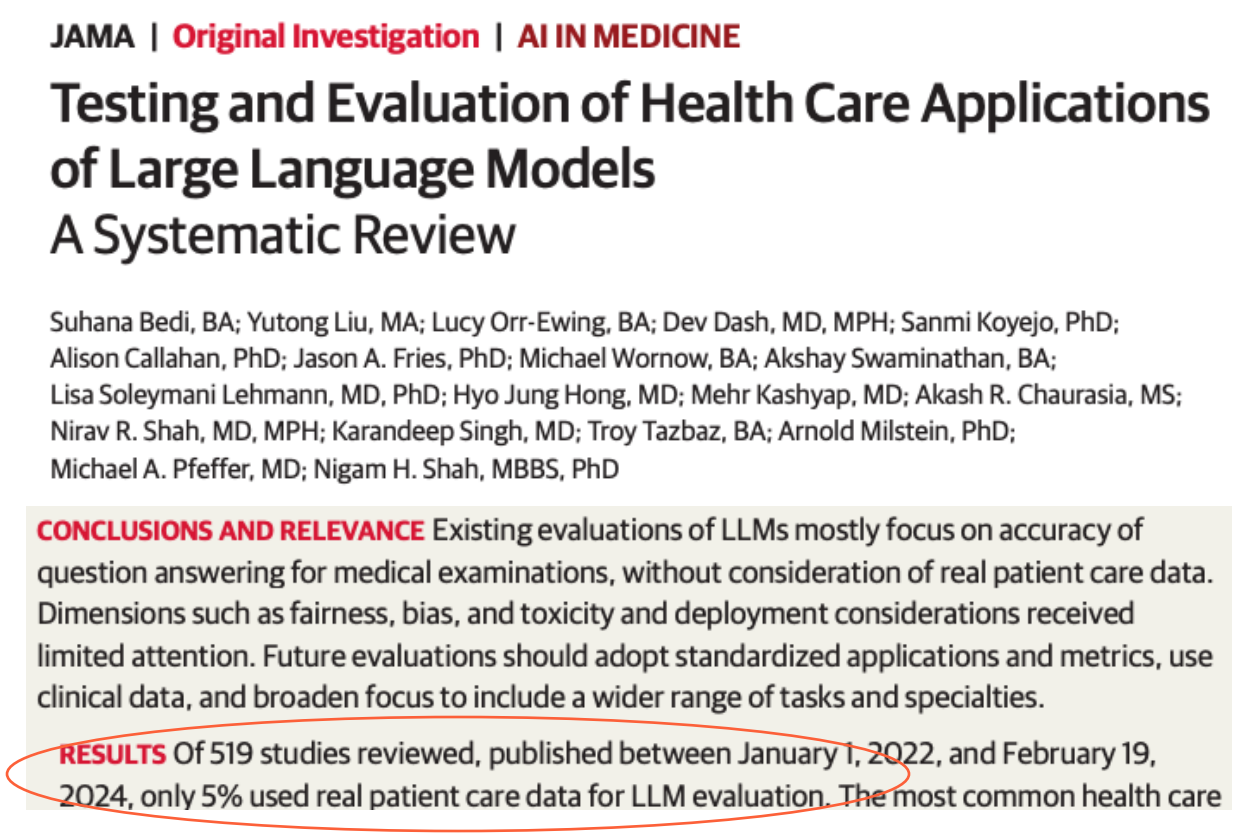
Risks

Evaluation on clinical data often lacking

Issues:

- Lack of compute infrastructure
- Lack of data
- Lack of privacy

LLMs are still in their infancy!



JAMA | Original Investigation | AI IN MEDICINE

Testing and Evaluation of Health Care Applications of Large Language Models
A Systematic Review

Suhana Bedi, BA; Yutong Liu, MA; Lucy Orr-Ewing, BA; Dev Dash, MD, MPH; Sanmi Koyejo, PhD; Alison Callahan, PhD; Jason A. Fries, PhD; Michael Wornow, BA; Akshay Swaminathan, BA; Lisa Soleymani Lehmann, MD, PhD; Hyo Jung Hong, MD; Mehr Kashyap, MD; Akash R. Chaurasia, MS; Nirav R. Shah, MD, MPH; Karandeep Singh, MD; Troy Tazbaz, BA; Arnold Milstein, PhD; Michael A. Pfeffer, MD; Nigam H. Shah, MBBS, PhD

CONCLUSIONS AND RELEVANCE Existing evaluations of LLMs mostly focus on accuracy of question answering for medical examinations, without consideration of real patient care data. Dimensions such as fairness, bias, and toxicity and deployment considerations received limited attention. Future evaluations should adopt standardized applications and metrics, use clinical data, and broaden focus to include a wider range of tasks and specialties.

RESULTS Of 519 studies reviewed, published between January 1, 2022, and February 19, 2024, only 5% used real patient care data for LLM evaluation. The most common health care

Introduction to LLMs

Common misunderstandings (reiterated)

“LLMs understand language like humans do.”
→ They don’t *understand*; they model statistical patterns of text.

“LLMs learn or reason at runtime.”
→ They retrieve and combine patterns from training, model parameters are NOT updated at runtime

“Bigger models with more data always mean better models.”
→ Scaling helps, but data quality, alignment, training settings and domain adaptation often matter more.

“LLMs are (not) deterministic.”
→ Depends on the set-up. They CAN be made to be deterministic, but most set-ups are not.

“LLMs will respond truthfully/will know when they are wrong”
LLMs are (mostly) trained to always give an answer, even if it was not embedded in the training data

Introduction to LLMs

Some tips (reiterated)

When not to use LLMs:

- **Fact retrieval:** they generate plausible text, not guaranteed truth.
- **Sensitive or confidential data:** risk of data leakage, especially with cloud models.
- **Clinical decision-making:** unsafe without expert oversight.

When LLMs can be useful:

- **Rewriting and reformatting:** improving clarity, tone, or style of existing text.
- **Summarizing:** condensing provided documents or transcripts.
- **Brainstorming or ideation:** generating options, examples, or outlines.
- **Structuring information:** turning free text into tables, bullet points, or templates.
- **Drafting non-critical text:** e.g., educational materials, reports, or patient communication.

Moral of the story:

- ***Treat LLMs as language assistants, not knowledge authorities.***
- ***Keep thinking for yourself! Two brains are better than one, but an e-brain is (still) worse than a real one!***

What have we learned?

Please note:

- **The test is anonymous**
- **You are not graded**

CODE: 1152 4551

https://www.menti.com/alh643u9hipb

UMC Utrecht

Assignment

Assignment
Finish the rest of the assignment (Question 4, 5, 6 and 7)
Fill in for yourself but discuss with your colleagues!
Make 3 slides discussing your

If there is time, we'll discuss at the end of the day.
Otherwise, we will discuss your plan tomorrow!

Prepare for tomorrow
Make sure you have an account for the following:
Huggingface.co account
Google account
Make sure you can access:
colab.research.google.com