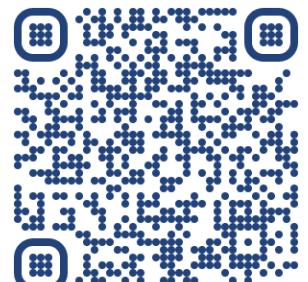


Hoe past het
concept
Generatieve AI
in het
AI Data Science
Ecosysteem?

Generative AI

**“Conversational
Agents
&
Prompt
Engineering”**



What Is AI?

Some Definitions...

Cognitive System is a natural or artificial system consisting of connected components that process informational input and transform it into observable output (i.e., a set of systems that interact with the world, understand, reason and learn).

Artificial Intelligence is a computer engineering discipline — a series of mathematically or logic-based techniques, uncovering, capturing, coding knowledge, and leveraging sophisticated and clever mechanisms to solve problems. i.e., a simulation of cognitive processes by means of computer programs.

Machine Learning at its most basic is the practice of using algorithms to parse data, capture knowledge, learn from it, and then make a determination or prediction.

Deep Learning is a subset of machine learning composed of algorithms that create knowledge from multiple layers of information processing.

Rule-Based Systems

Optimization Systems

Natural Language Processing

Knowledge Representation

Agent-Based Computing and Orchestration

Perception Systems and Ambient Intelligence

Wat is AI?

enige definities ...

Cognitief systeem is een natuurlijk of kunstmatig systeem dat bestaat uit verbonden componenten die informatieve input verwerken en omzetten in waarneembare output (d.w.z. een reeks systemen die interacteren met de wereld, begrijpen, redeneren en leren).

Artificial Intelligence is een computer technische discipline - een reeks op wiskundige of logica gebaseerde technieken, die kennis blootlegt, vastlegt, codeert en geavanceerde en slimme mechanismen gebruikt om problemen op te lossen. d.w.z. een simulatie van cognitieve processen door middel van computerprogramma's.

Machine learning in de basis is het gebruik van algoritmen om gegevens te ontleden, kennis vast te leggen, ervan te leren en vervolgens een bepaling of voorspelling te doen.

Deep learning is een subset van machine learning die bestaat uit algoritmen die kennis creëren uit meerdere lagen van informatieverwerking.

Rule-based systemen

Optimalisatie systemen

Natural Language Processing

Knowledge representatie

Agent-Based Computing and orchestratie

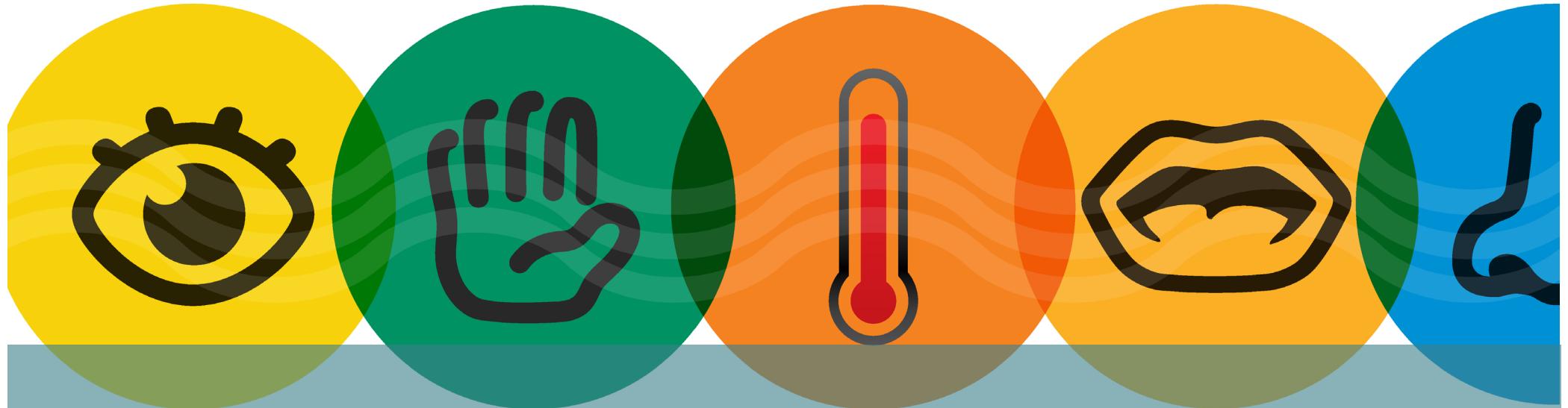
Perception systemen en Ambient Intelligence

Artificial intelligence {AI}

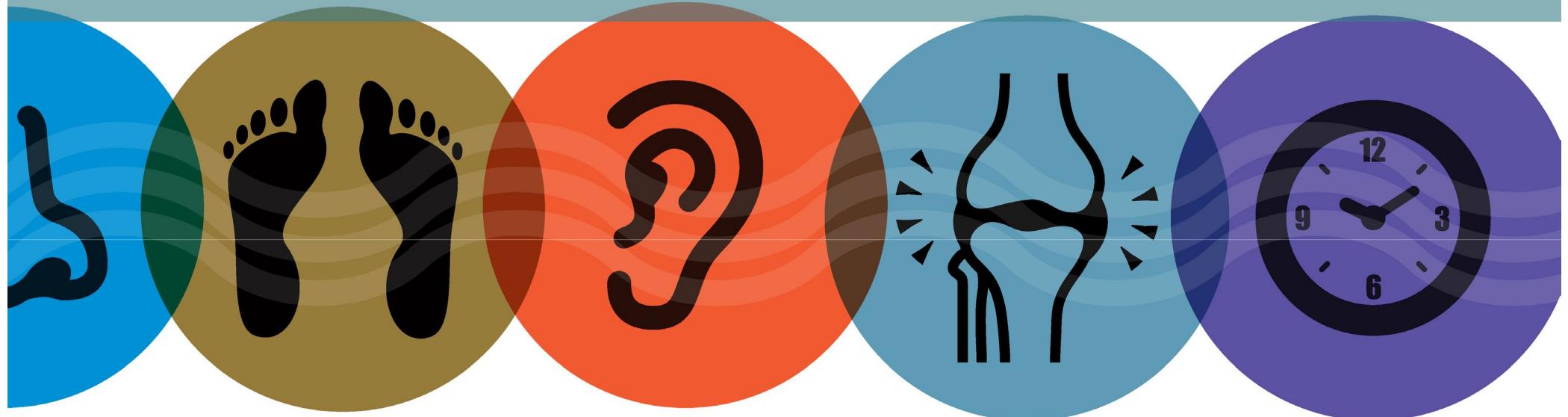
“Kunstmatige Intelligentie”
Wat is dat eigenlijk?

"Kunstmatige intelligentie" {AI}

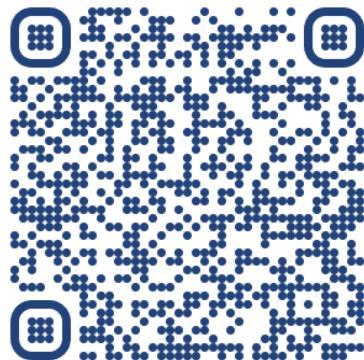
is een multidisciplinair vakgebied gericht op onderzoek naar en het ontwikkelen van technologie dat menselijke [multimodale] perceptuele, cognitieve en/of motorische vermogens kan nabootsen en/of automatiseren.



ZINTUIGELIJKE MODALITEITEN



<https://doi.org/10.1609/aimag.v37i1.2643>



Cognitie	Menselijk Vermogen	Toepassing
Perceptie (multimodaal) (invoer)	Verwerken, Begrijpen en Onthouden van informatie via het Brein.	Spraak, Natuurlijke Taalverwerking , Logisch Redeneren, Problemen Oplossen, Theory of Mind [ToM] .
Motorische vermogens (uitvoer)	Verzamelen van multimodale Ongestructureerde Informatie via Sensoren : Ogen, Oren, Neus, Huid, Tong	Beeldherkenning, Sprakherkenning, Object Lokalisatie.
	Uitvoeren van fysieke acties via Effectoren : Armen, Handen, Benen, Voeten, Mond, Lippen	Zelfrijdende auto's, Drones, Humanoïde Robots, Tekst-naar-spraak synthese

<https://robfvdw.medium.com/the-world-wide-web-ai-safari-b2e4f7f90647>

AI-model versus Agent

Agent == taak specifiek + doelgericht

Een versimpelde versie van de echte wereld **{invoer-uitvoer model}** dat de directe omgeving kan waarnemen via sensoren (**input**) en deze omgeving kan beïnvloeden via effectoren (**output**), gecombineerd met **zelflerend vermogen via feedback**.

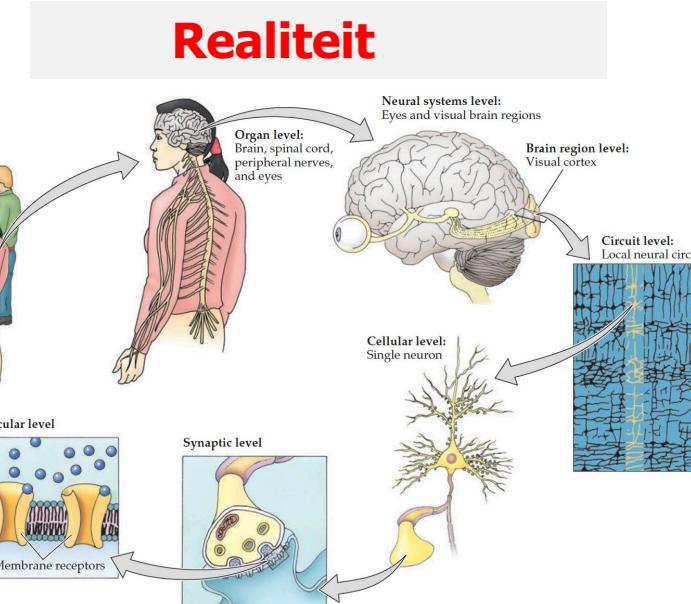
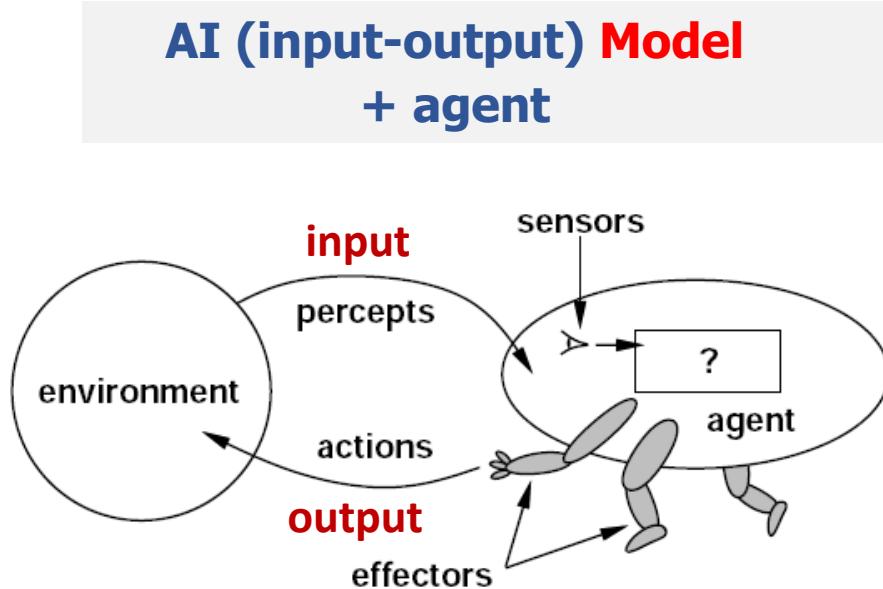
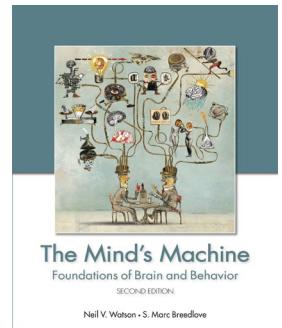
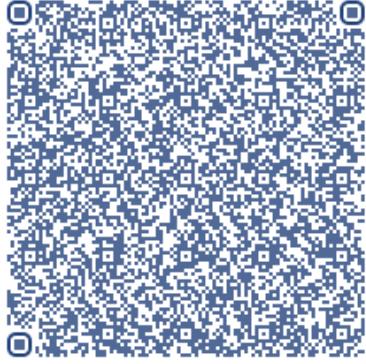


FIGURE 1.10 Levels of Analysis in Biological Psychology

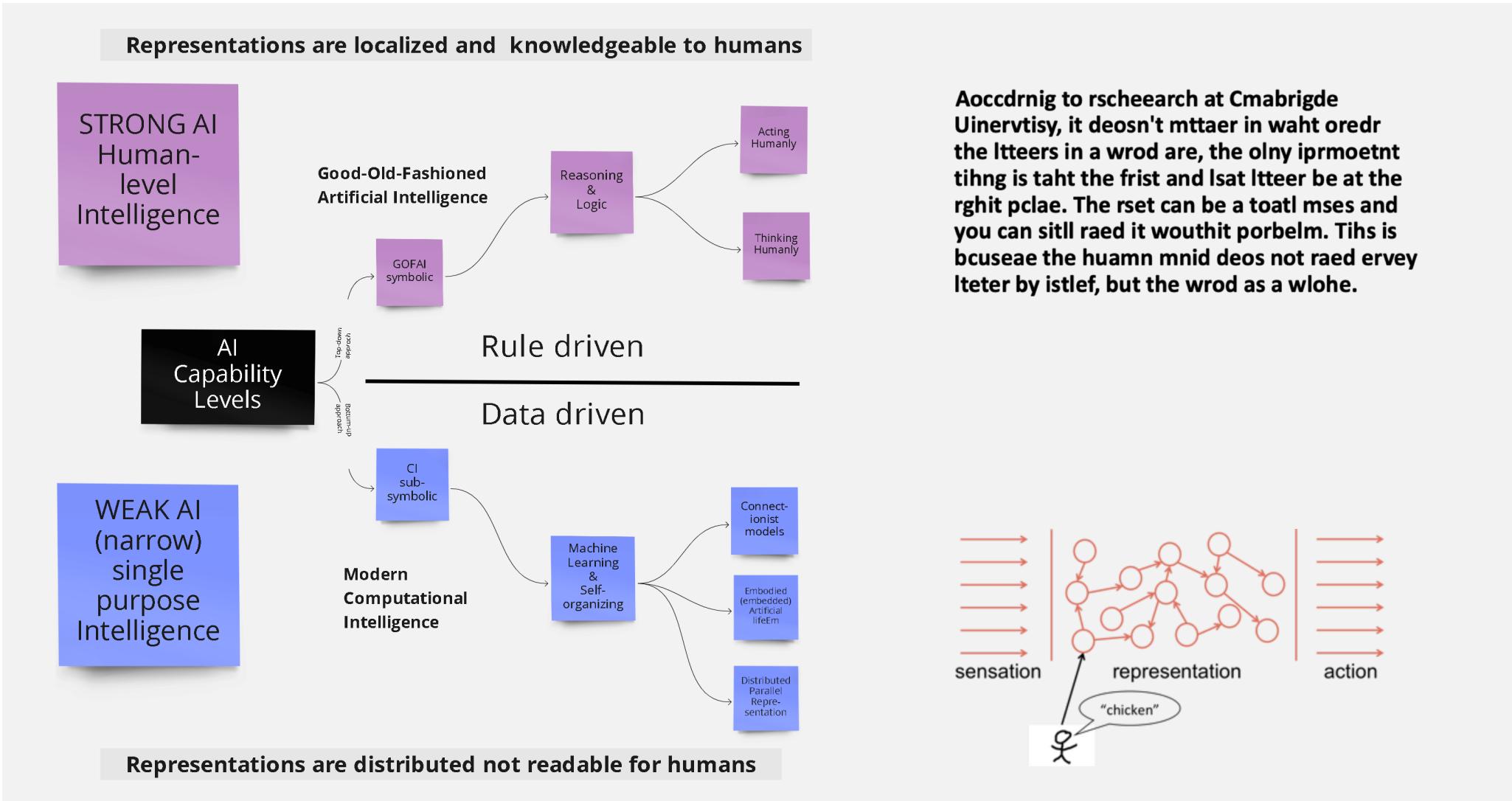
Animal research is an essential part of life sciences research, including biological psychology

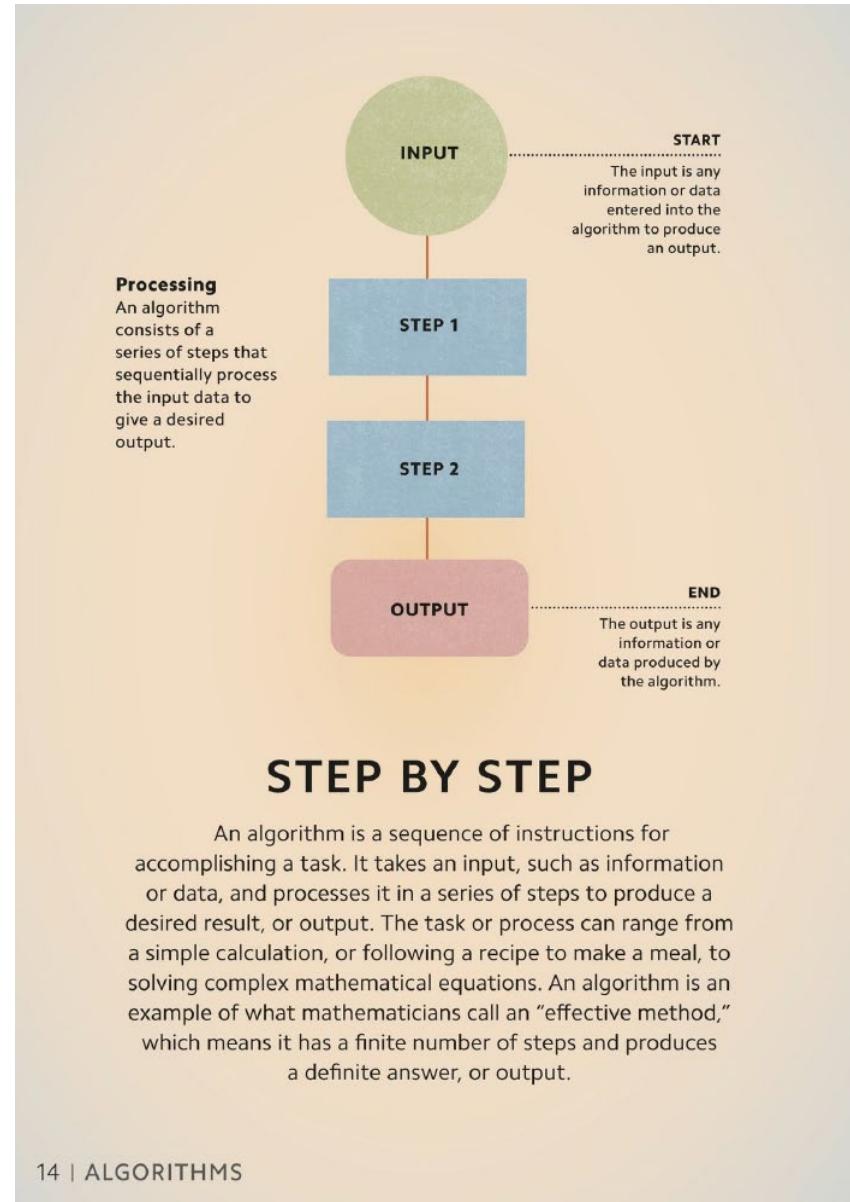
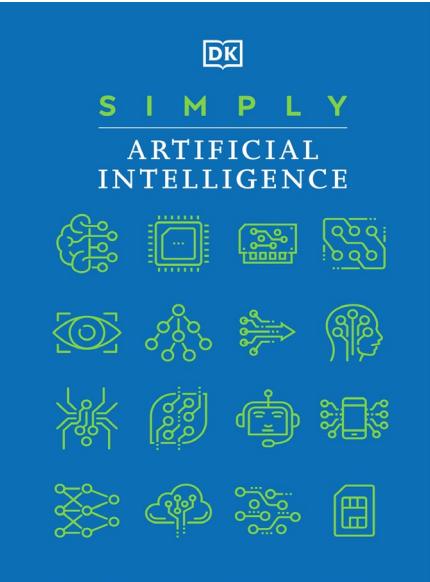


AI-taxonomie is complex



https://www.researchgate.net/publication/359424818_Designing_Neural_Networks_Through_Sensory_Ecology_Biology_to_the_rescue_of_AI_Produced_by_Living-Lab_AiRA_Hub_voor_Data_Responsible_AI_Hogeschool_Rotterdam_Lunch-Lezing_Creating-010_FEB_2022



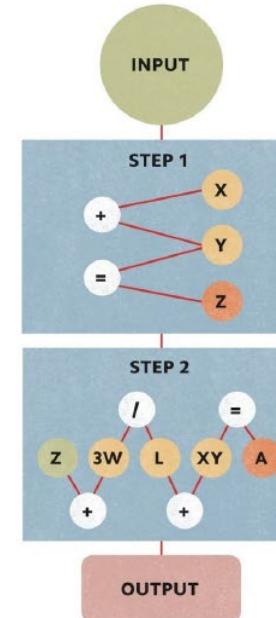


STEP BY STEP

An algorithm is a sequence of instructions for accomplishing a task. It takes an input, such as information or data, and processes it in a series of steps to produce a desired result, or output. The task or process can range from a simple calculation, or following a recipe to make a meal, to solving complex mathematical equations. An algorithm is an example of what mathematicians call an “effective method,” which means it has a finite number of steps and produces a definite answer, or output.

Components of calculation

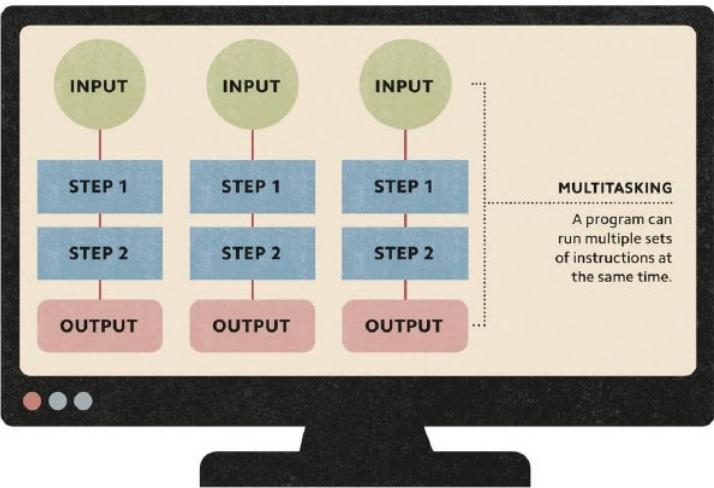
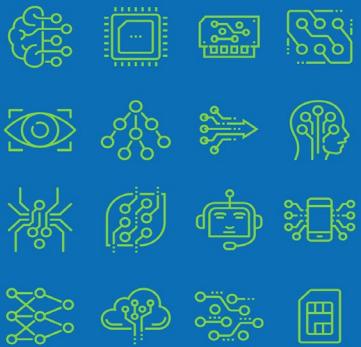
Computations have an input and an output, and multiple steps. They can vary from simple sums to complex equations.



ALGORITHMS IN ACTION

A computation is a calculation that follows the steps of an algorithm (see opposite). The most straightforward example of computation is arithmetic calculation. For example, if you add together a pair of three-digit numbers in your head, you follow a series of steps, or an algorithm, to achieve this calculation. Computations use symbols to represent numbers, but symbols can represent almost anything else (see p.36). With the right symbols and the right algorithms, immensely complex computation becomes possible.

SIMPLY

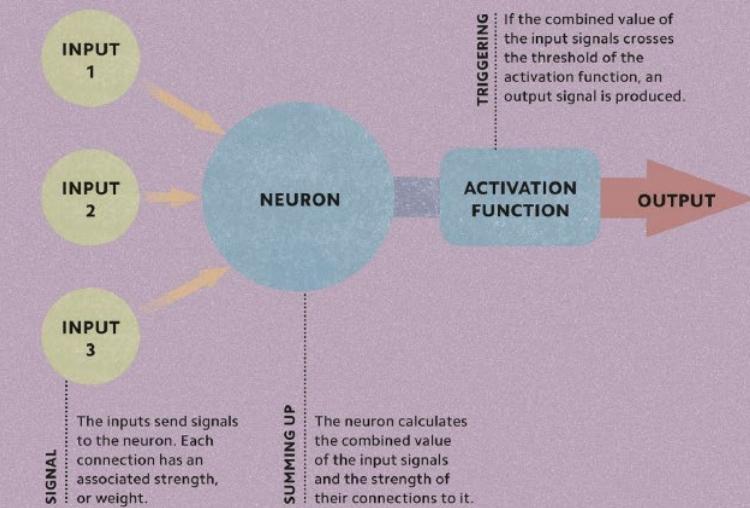
ARTIFICIAL
INTELLIGENCE

INSTRUCTING COMPUTERS

A program is a sequence of instructions written in code that enables a computer to perform one or more tasks. Charles Babbage (see opposite) imagined the first program. He was inspired by the design of a certain silk loom, which had parts that moved up or down in response to a pattern of holes punched into a card. Babbage recognized that these holes could store instructions to operate the cogs and levers of a machine he was designing: the "Analytical Engine". Modern computers work on the same principle, following sequences of instructions, which are usually written in binary code (see p.13).

ARTIFICIAL NEURONS

Each of the 86 billion neurons in the human brain is effectively a tiny processor, receiving electrical signals (inputs) from other neurons and sending out signals of its own (outputs). McCulloch and Pitts (see opposite) realized that neurons can act as logic gates—devices that can switch on and off (see p.13), depending on the input. The scientists described an imaginary neuron called a "threshold logic unit". This neuron works by first adding the values of its inputs (signals from other neurons) and then multiplying that value by a variable called a "weight" (see p.78)—this is the strength of a connection between neurons. If the input signals exceed a certain value (see p.79), the neuron is triggered to send an output signal. This triggering is called the "activation function".



Defenities / Omschrijvingen / Richtlijnen voor veilig gebruik

“Generatieve AI”

Generative AI according to Google

Machine learning (ML) model that can take what it has learned from multimodal-examples it has been provided to create new content, such as text, images, music, and code.

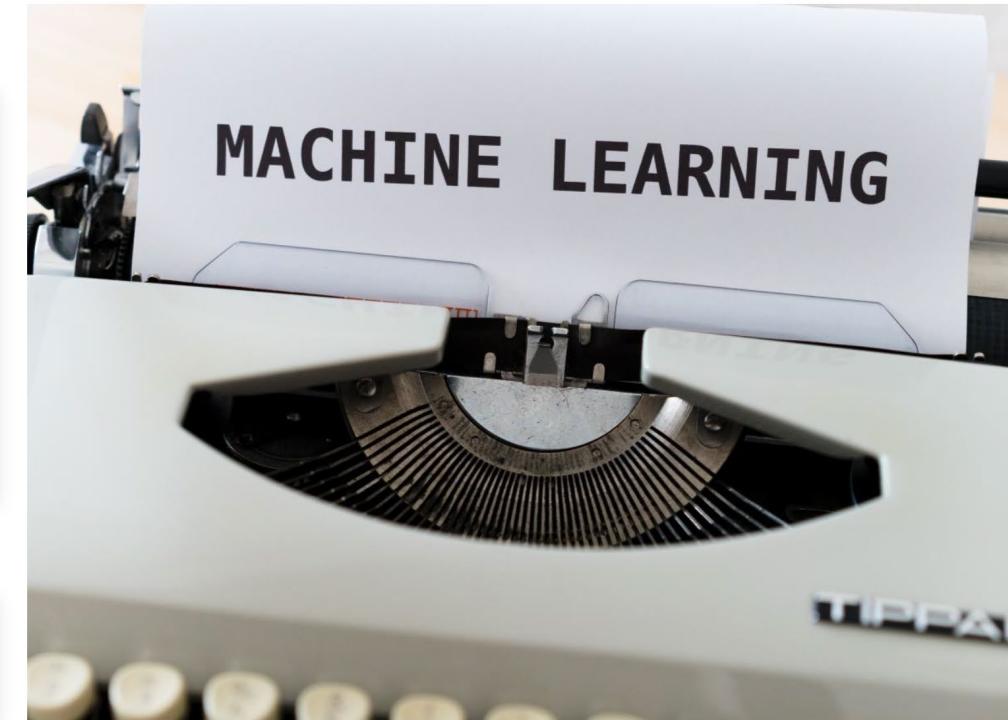
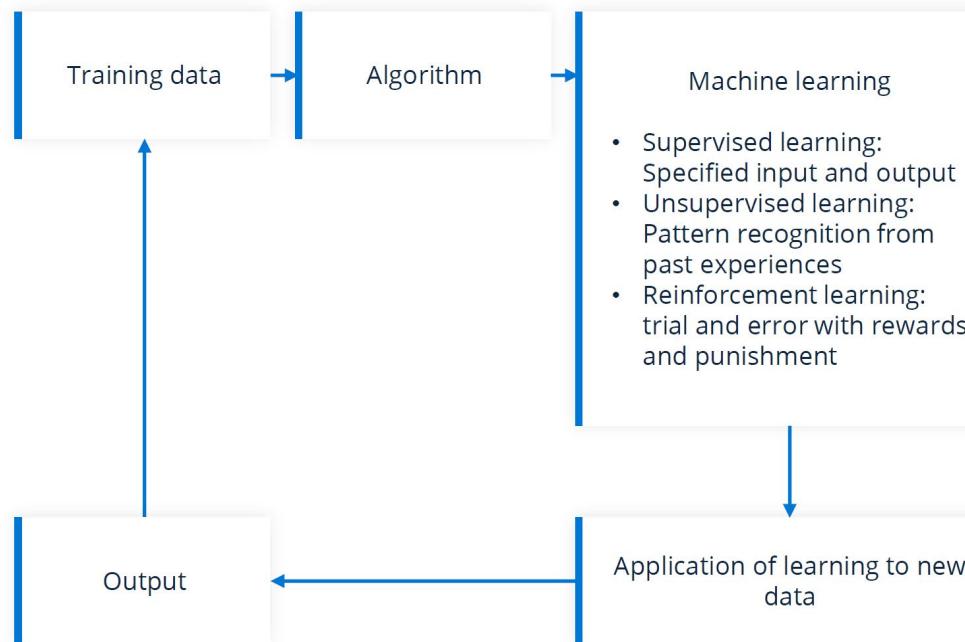
These models learn through observation and pattern matching, also known as training.

Generative AI models are neither information databases nor deterministic information retrieval systems, because they are prediction engines.

There are three types of machine learning applications

Machine learning (2/2)

Illustration of the machine learning process



Generative AI according to TechTarget

Generative AI is a type of artificial intelligence technology that can produce various types of content, including text, imagery, audio and synthetic data.

The recent buzz around generative AI has been driven by the simplicity of new user interfaces for creating high-quality text, graphics and videos in a matter of seconds.

The technology, it should be noted, is not brand-new. Generative AI was introduced in the 1960s in chatbots.

But it was not until 2014, with the introduction of generative adversarial networks, or GANs --a type of machine learning algorithm-- that generative AI could create convincingly authentic images, videos and audio of real people.

Top Strategic Technology Trends for 2022: Generative AI

Published 18 October 2021 - ID G00756059 - 10 min read

By Analyst(s): Anthony Mullen, Nicole Greene, Brent Stewart, Marc Halpern, Soyeb Barot

Initiatives: Artificial Intelligence; Evolve Technology and Process Capabilities to Support D&A

Generative AI is a disruptive technology that can generate artifacts that previously relied on humans, delivering innovative results without the biases of human experiences and thought processes. IT leaders must use the appropriate governance to exploit its transformative potential.

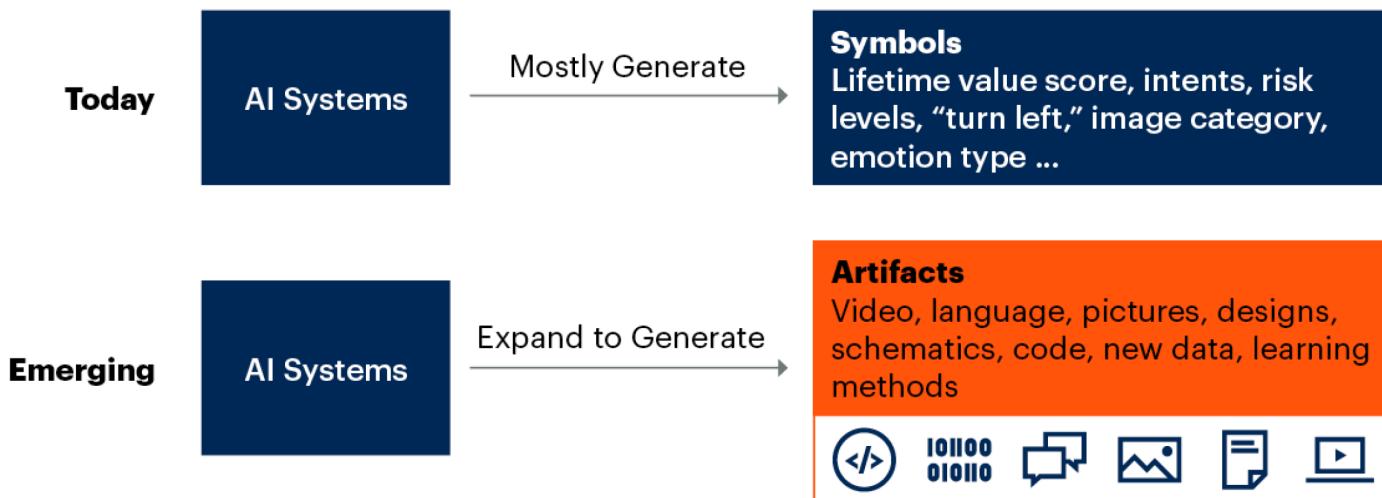
Additional Perspectives

- [Summary Translation + Localization: Top Strategic Technology Trends for 2022: Generative AI](#)
(20 December 2021)

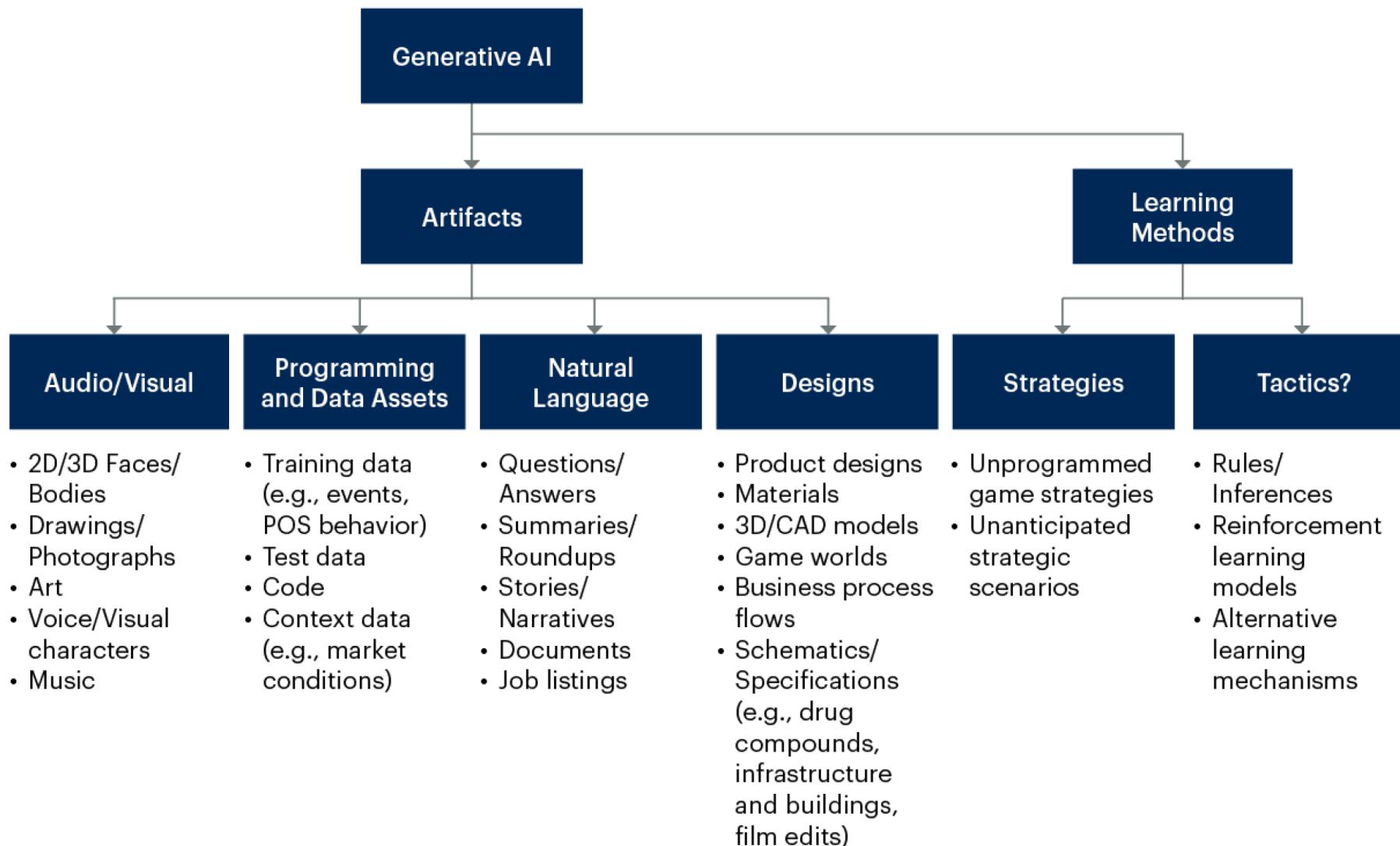
Generative AI expands the output of AI systems to include high-value artifacts such as video, narrative, training data and even designs and schematics (see Figure 1).

Figure 1: Expanding the Output of AI Systems

Expanding the Output of AI Systems



What Types of Artifacts Can Generative AI Create?



Source: Gartner

756059_C

Quick Answer: How Will Prompt Engineering Impact the Work of Data Scientists?

Published 9 June 2023 - ID G00794952 - 6 min read

Afraz Jaffri

Initiatives: [Analytics, BI and Data Science Solutions](#); [Evolve Technology and Process Capabilities to Support D&A](#)

Generative AI models produce outputs based on natural language inputs or “prompts.” Prompts are generally seen as simple instructions, but data and analytics leaders should utilize data scientists’ expertise to build prompts capable of more complex tasks and accelerate their own productivity.

Additional Perspectives

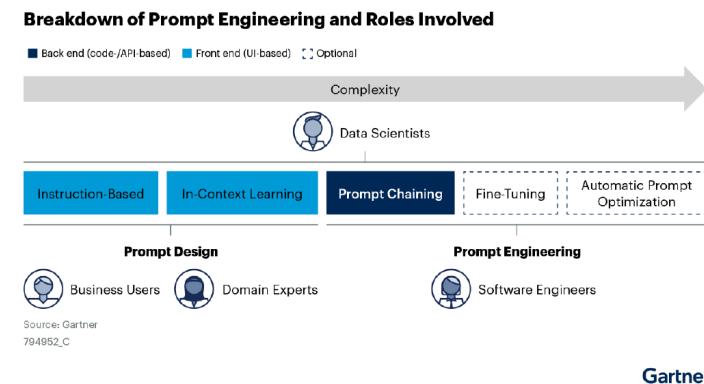
- [Summary Translation: Quick Answer: How Will Prompt Engineering Impact the Work of Data Scientists?](#)
(11 August 2023)

Quick Answer

How will prompt engineering impact the work of data scientists?

- Prompt engineering can customize large language models (LLMs) as part of the workflow to power generative AI applications. Data scientists’ underlying knowledge of LLM architecture makes them well-suited to this task.
- Prompt engineering will be used with LLMs to accelerate common machine learning tasks on text and images, such as categorization, summarization, annotation and sentiment analysis.
- Code generation enhanced by prompt engineering will accelerate many tasks, including exploratory data analysis, model selection, and training and model evaluation. These tasks will become increasingly declarative as specialized models perform subtasks in reaching an overall objective.
- Adversarial attacks and unwanted bias can be perpetrated through intentional or unintentional prompt manipulation. The tasks of prompt validation and testing are likely to initially be the responsibility of data scientists.

Figure 1: Breakdown of Prompt Engineering and Roles Involved



Prompt-based approaches involve both prompt design to provide clear definitions of what is required and back-end prompt engineering to automate, chain and configure prompts in a pipeline for use in AI applications. New styles of prompt techniques are being developed by data scientists in academia and industry. Examples include:

- Chain-of-Thought prompting,² where a model is encouraged to show its working or thinking step-by-step; and
- Tree of Thoughts, where multiple prompts are evaluated and refined until a task is complete.³

Data scientists can use prompt design and engineering to unlock capabilities of an LLM that are not well-known or even thought to be possible.

Data scientists will also need to decide when an LLM can be used to go beyond text generation to the execution of processes and utilization of external services. In April 2023, Gartner’s social media analysis found a significant number of conversations from business executives related to AutoGPT, an open-source application that automates tasks by leveraging GPT4 and integrations to tools. Utilizing models in this way will require significant prompt engineering.

MCKINSEY GLOBAL INSTITUTE
NOTES FROM THE AI FRONTIER
INSIGHTS FROM HUNDREDS OF USE CASES

DISCUSSION PAPER
APRIL 2018

Michael Chui | San Francisco
James Manyika | San Francisco
Mehdi Meimand | Chicago
Nicolas Herremans | London
Rita Chung | Silicon Valley
Peter Nel | New York
Sankarpal Malhotra | New York

2018

Exhibit 3

Heat map: Technique relevance to industries

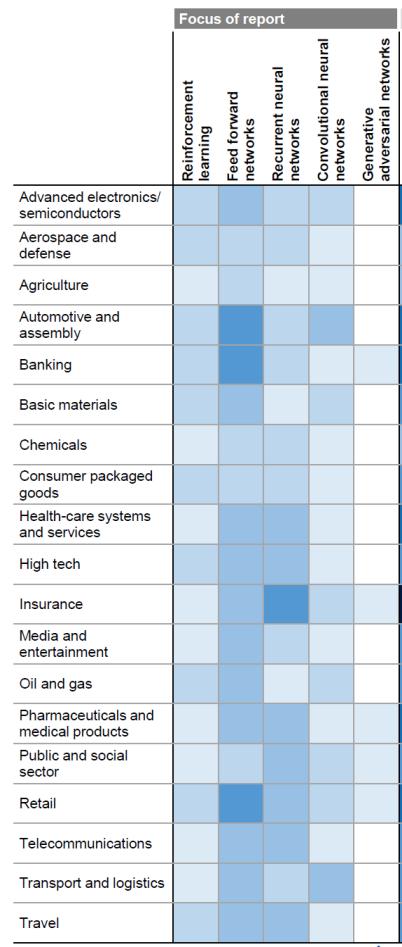
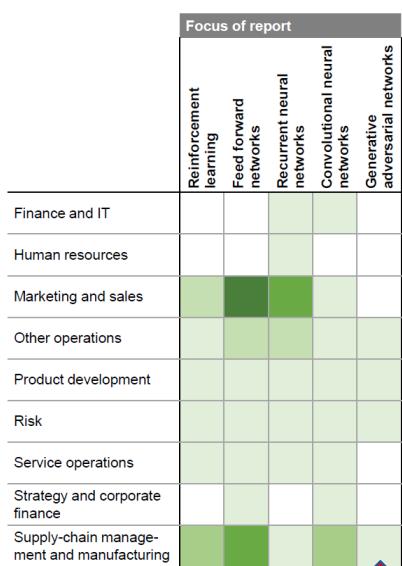


Exhibit 4

Heat map: Technique relevance to functions



The economic potential of generative AI

The next productivity frontier

June 2023

Authors

Michael Chui

James Manyika

Mehdi Meimand

Rita Chung

Nicolas Herremans

Peter Nel

Kurt Smeltz

Alex Sushkevich

Rodrigo Zamora

2023

Generative AI use cases will have different impacts on business functions across industries.

Generative AI productivity impact by business functions¹

Low impact High impact



The road to human-level performance just got shorter

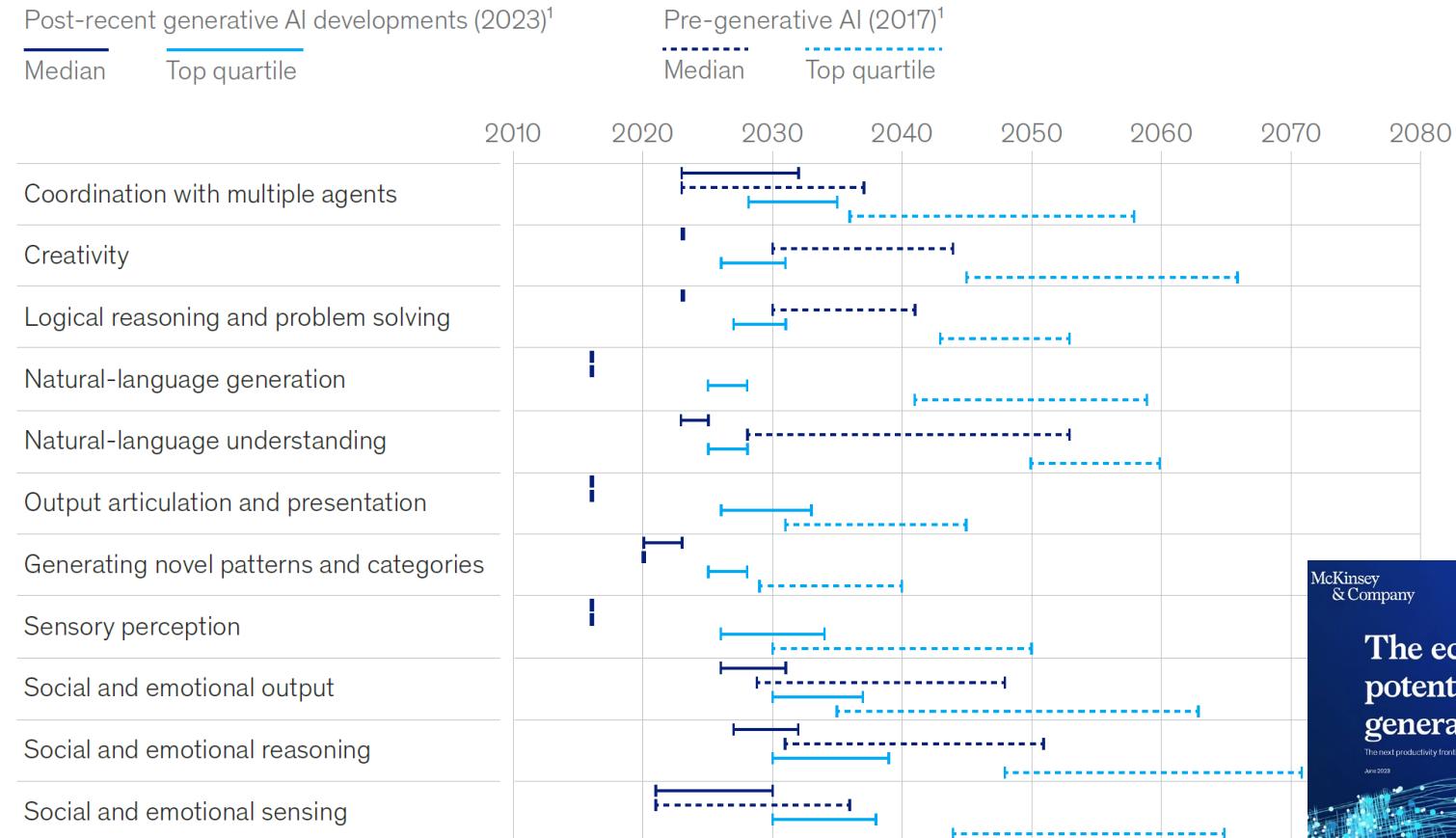
For most of the technical capabilities shown in this chart, gen AI will perform at a median level of human performance by the end of this decade. And its performance will compete with the top 25 percent of people completing any and all of these tasks before 2040. In some cases, that's 40 years faster than experts previously thought.

Source: *The economic potential of generative AI: The next productivity frontier*

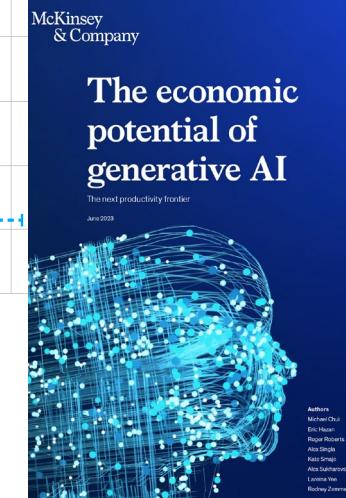
<https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier>

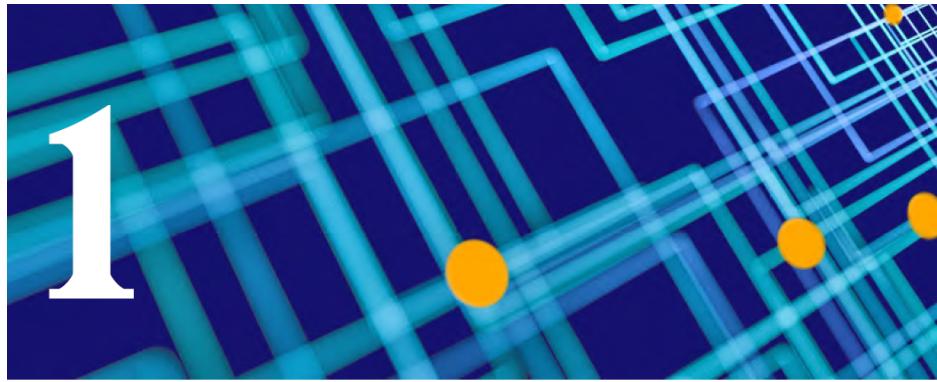
Due to generative AI, experts assess that technology could achieve human-level performance in some capabilities sooner than previously thought.

Estimated range for technology to achieve human-level performance, by technical capability



¹Comparison made on the business-related tasks required from human workers.
Source: McKinsey Global Institute occupation database; McKinsey analysis





Generative AI as a technology catalyst

To grasp what lies ahead requires an understanding of the breakthroughs that have enabled the rise of generative AI, which were decades in the making. ChatGPT, GitHub Copilot, Stable Diffusion, and other generative AI tools that have captured current public attention are the result of significant levels of investment in recent years that have helped advance machine learning and deep learning. This investment undergirds the AI applications embedded in many of the products and services we use every day.

But because AI has permeated our lives incrementally—through everything from the tech powering our smartphones to autonomous-driving features on cars to the tools retailers use to surprise and delight consumers—it’s progress was almost imperceptible. Clear milestones, such as when AlphaGo, an AI-based program developed by DeepMind, defeated a world champion Go player in 2016, were celebrated but then quickly faded from the public’s consciousness.

ChatGPT and its competitors have captured the imagination of people around the world in a way AlphaGo did not, thanks to their broad utility—almost anyone can use them to communicate and create—and preternatural ability to have a conversation with a user. The latest generative AI applications can perform a range of routine tasks, such as the reorganization and classification of data. But it is their ability to write text, compose music, and create digital art that has garnered headlines and persuaded consumers and households to experiment on their own. As a result, a broader set of stakeholders are grappling with generative AI’s impact on business and society but without much context to help them make sense of it.

McKinsey
& Company

The economic potential of generative AI

The next productivity frontier

June 2023



Glossary

Application programming interface (API) is a way to programmatically access (usually external) models, data sets, or other pieces of software.

Artificial intelligence (AI) is the ability of software to perform tasks that traditionally require human intelligence.

Artificial neural networks (ANNs) are composed of interconnected layers of software-based calculators known as “neurons.” These networks can absorb vast amounts of input data and process that data through multiple layers that extract and learn the data’s features.

Deep learning is a subset of machine learning that uses deep neural networks, which are layers of connected “neurons” whose connections have parameters or weights that can be trained. It is especially effective at learning from unstructured data such as images, text, and audio.

Early and late scenarios are the extreme scenarios of our work-automation model. The “earliest” scenario flexes all parameters to the extremes of plausible assumptions, resulting in faster automation development and adoption, and the “latest” scenario flexes all parameters in the opposite direction. The reality is likely to fall somewhere between the two.

Fine-tuning is the process of adapting a pretrained foundation model to perform better in a specific task. This entails a relatively short period of training on a labeled data set, which is much smaller than the data set the model was initially trained on. This additional training allows the model to learn and adapt to the nuances, terminology, and specific patterns found in the smaller data set.

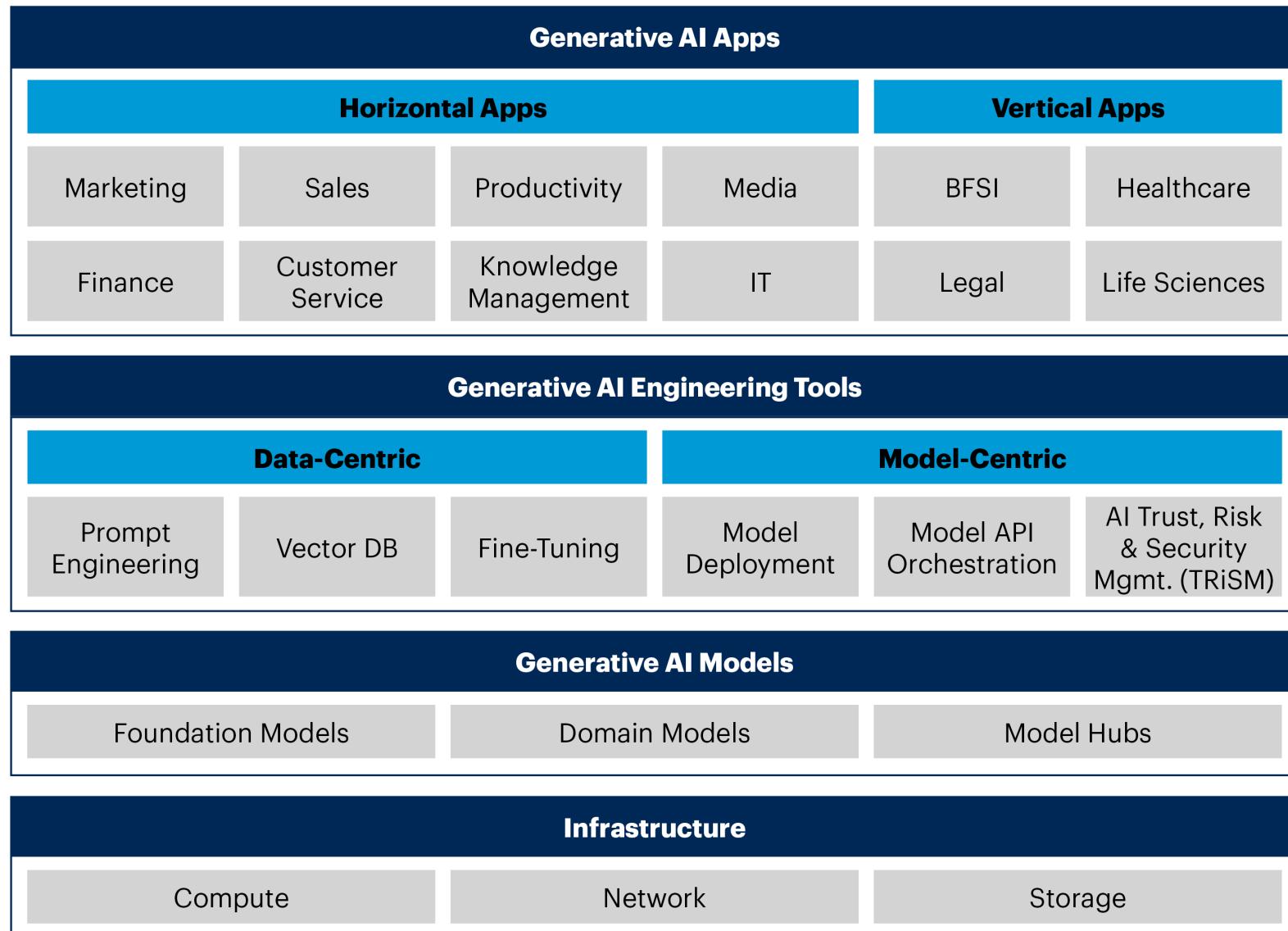
Foundation models (FM) are deep learning models trained on vast quantities of unstructured, unlabeled data that can be used for a wide range of tasks out of the box or adapted to specific tasks through fine-tuning. Examples of these models are GPT-4, PaLM, DALL-E 2, and Stable Diffusion.

Generative AI is AI that is typically built using foundation models and has capabilities that earlier AI did not have, such as the ability to generate content. Foundation models can also be used for nongenerative purposes (for example, classifying user sentiment as negative or positive based on call transcripts) while offering significant improvement over earlier models. For simplicity, when we refer to generative AI in this article, we include all foundation model use cases.

Graphics processing units (GPUs) are computer chips that were originally developed for producing computer graphics (such as for video games) and are also useful for deep learning applications. In contrast, traditional machine learning and other analyses usually run on central processing units (CPUs), normally referred to as a computer’s “processor.”

Large language models (LLMs) make up a class of foundation models that can process massive amounts of unstructured text and learn the relationships between words or portions of words, known as tokens. This enables LLMs to generate natural-language text, performing tasks such as summarization or knowledge extraction. GPT-4 (which underlies ChatGPT) and LaMDA (the model behind Bard) are examples of LLMs.

Generative AI Technology Landscape

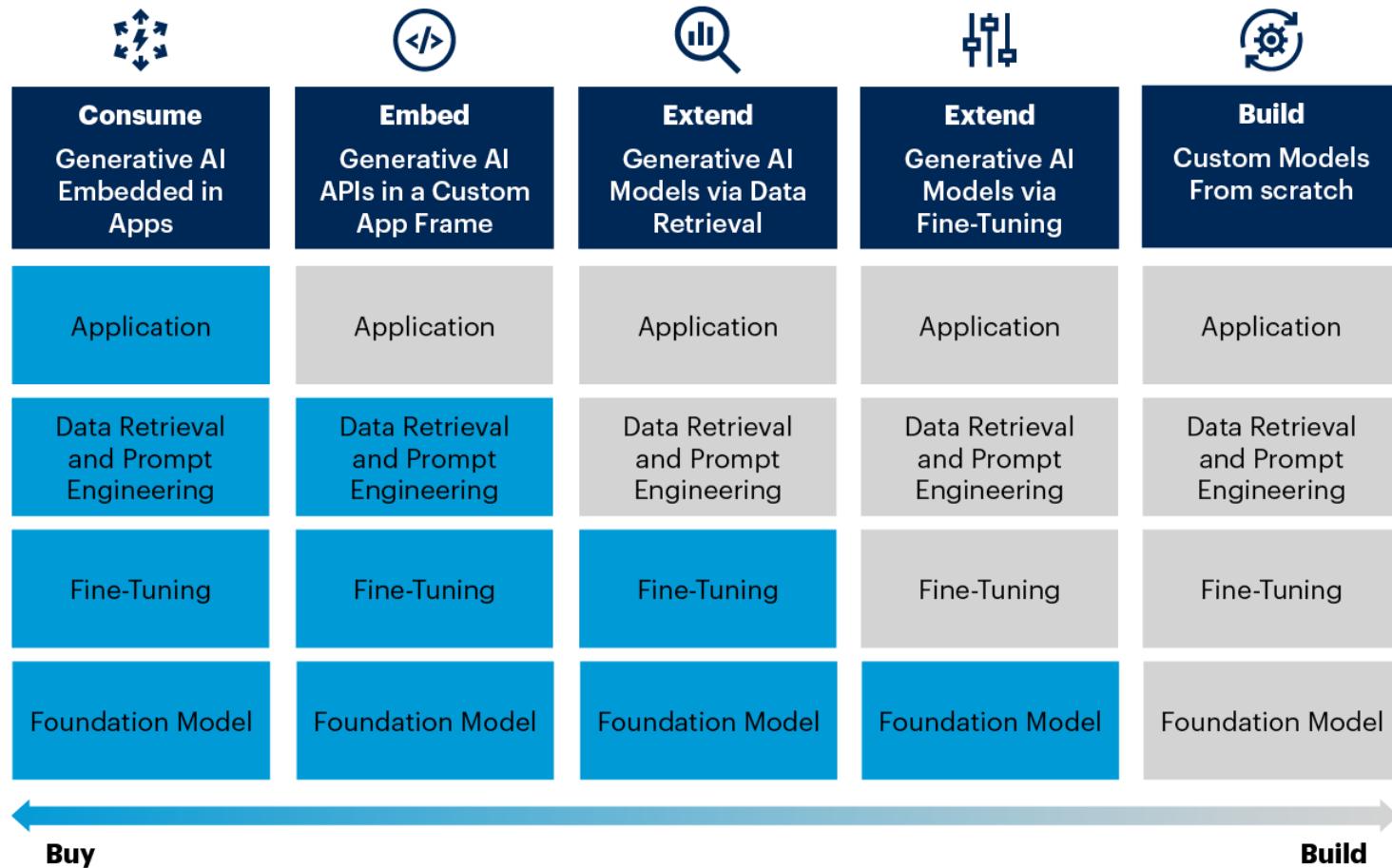


Source: Gartner

793970_C

Generative AI Deployment Approaches

■ Provider-Managed ■ Self-Managed



Source: Gartner

794559_C

Office of the Provost

HOME /

Guidelines for Using ChatGPT and other Generative AI tools at Harvard

Dear Members of the Harvard Community,

We write today with initial guidelines on the use and procurement of generative artificial intelligence (AI) tools, such as OpenAI's ChatGPT and Google Bard. The University supports responsible experimentation with generative AI tools, but there are important considerations to keep in mind when using these tools, including information security and data privacy, compliance, copyright, and academic integrity.

Generative AI is a rapidly evolving technology, and the University will continue to monitor developments and incorporate feedback from the Harvard community to update our guidelines accordingly.

Initial guidelines for use of generative AI tools:

- **Protect confidential data:** You should not enter data **classified as confidential** (Level 2 and above), including non-public research data, into publicly-available generative AI tools, in accordance with the University's **Information Security Policy**. Information shared with generative AI tools using default settings is not private and could expose proprietary or sensitive information to unauthorized parties.
- **You are responsible for any content that you produce or publish that includes AI-generated material:** AI-generated content can be inaccurate, misleading, or entirely fabricated (sometimes called "hallucinations"), or may contain copyrighted material. Review your AI-generated content before publication.
- **Adhere to current policies on academic integrity:** Review your School's student and faculty handbooks and policies. We expect that Schools will be developing and updating their policies as we better understand the implications of using generative AI tools. In the meantime, faculty should be clear with students they're teaching and advising about their policies on permitted uses, if any, of generative AI in classes and on academic work. Students are also encouraged to ask their instructors for clarification about these policies as needed.
- **Be alert for AI-enabled phishing:** Generative AI has made it easier for malicious actors to create sophisticated scams at a far greater scale. Continue to **follow security best practices** and report suspicious messages to phishing@harvard.edu.
- **Connect with HUIT before procuring generative AI tools:** The University is working to ensure that tools procured on behalf of Harvard have the appropriate privacy and security protections and provide the best use of Harvard funds.
 - If you have procured or are considering procuring generative AI tools or have questions, contact HUIT at ithelp@harvard.edu.
 - Vendor generative AI tools must be **assessed for risk by Harvard's Information Security and Data Privacy office prior to use**.

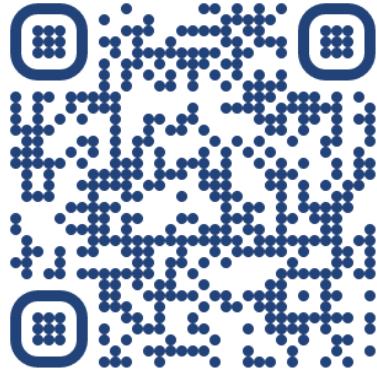
It is important to note that these guidelines are not new University policy; rather, they leverage existing University policies. You can find more information about generative AI, including a survey to collect data on its potential use, **on the HUIT website**, which will be updated as new information becomes available.

<https://provost.harvard.edu/guidelines-using-chatgpt-and-other-generative-ai-tools-harvard>

Artificial intelligence {AI}

*“Taxonomie van het
generatieve
AI-landschap is complex”*

AI-taxonomie is complex



Crothers, E., Japkowicz, N., & Viktor, H. (2022). Machine Generated Text: A Comprehensive Survey of Threat Models and Detection Methods. arXiv preprint <https://doi.org/10.48550/arXiv.2210.07321>

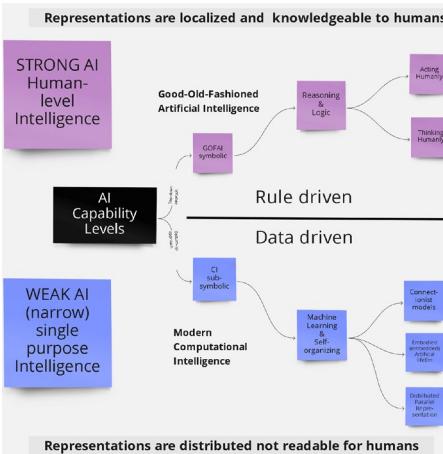
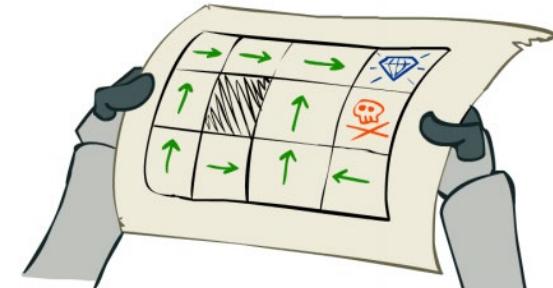
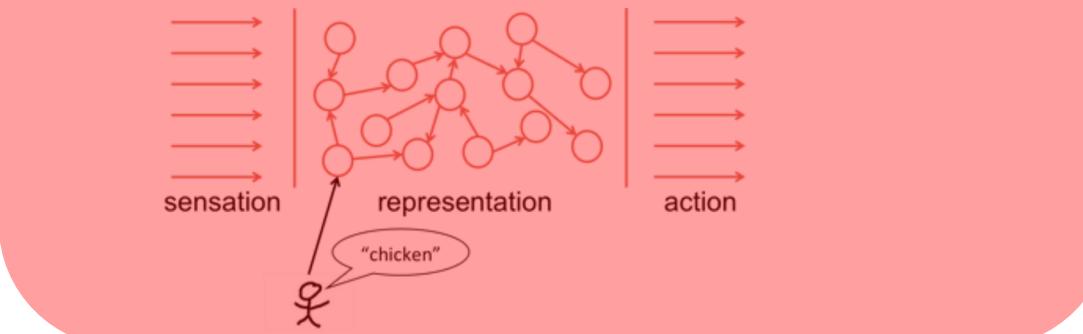
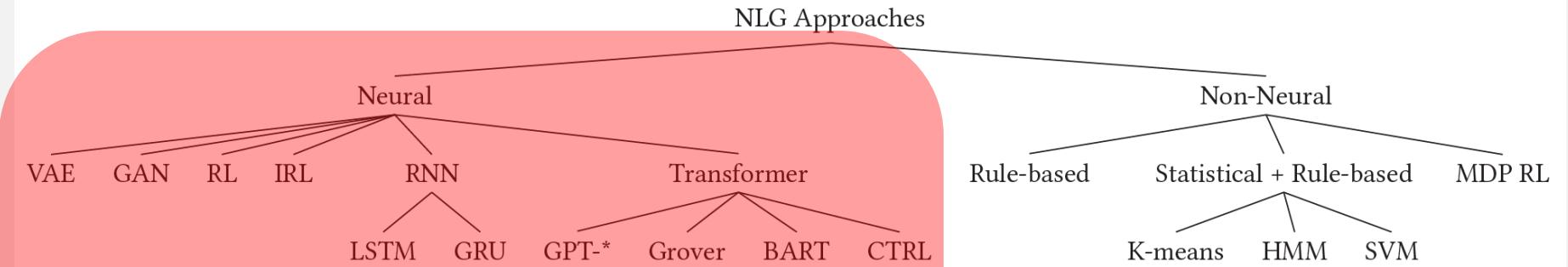
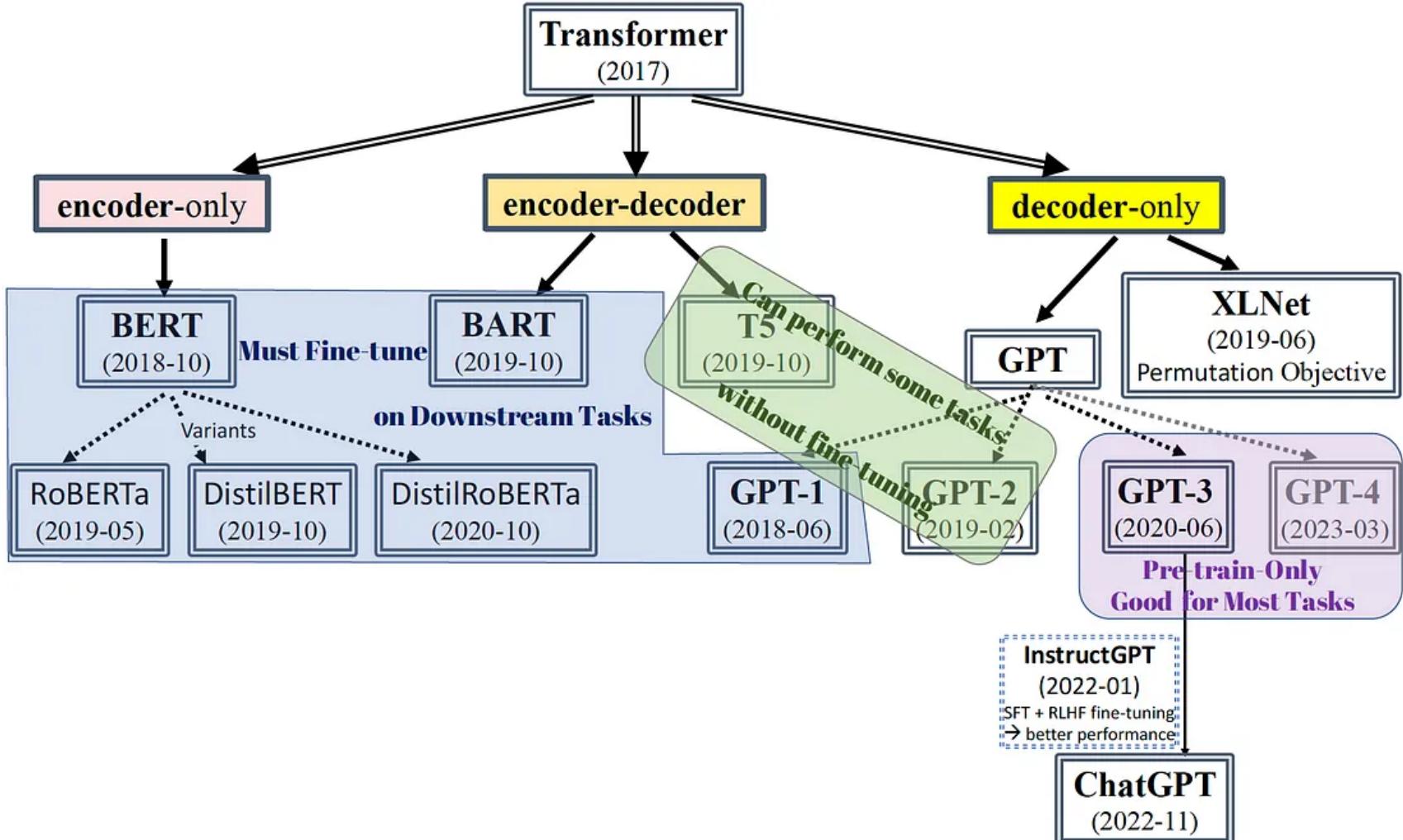


Fig. 1. Taxonomy of major NLG approaches



AI-taxonomie is complex



<https://medium.com/@yulemon/an-in-depth-look-at-the-transformer-based-models-22e5f5d17b6b>

{AI=ML=DL}

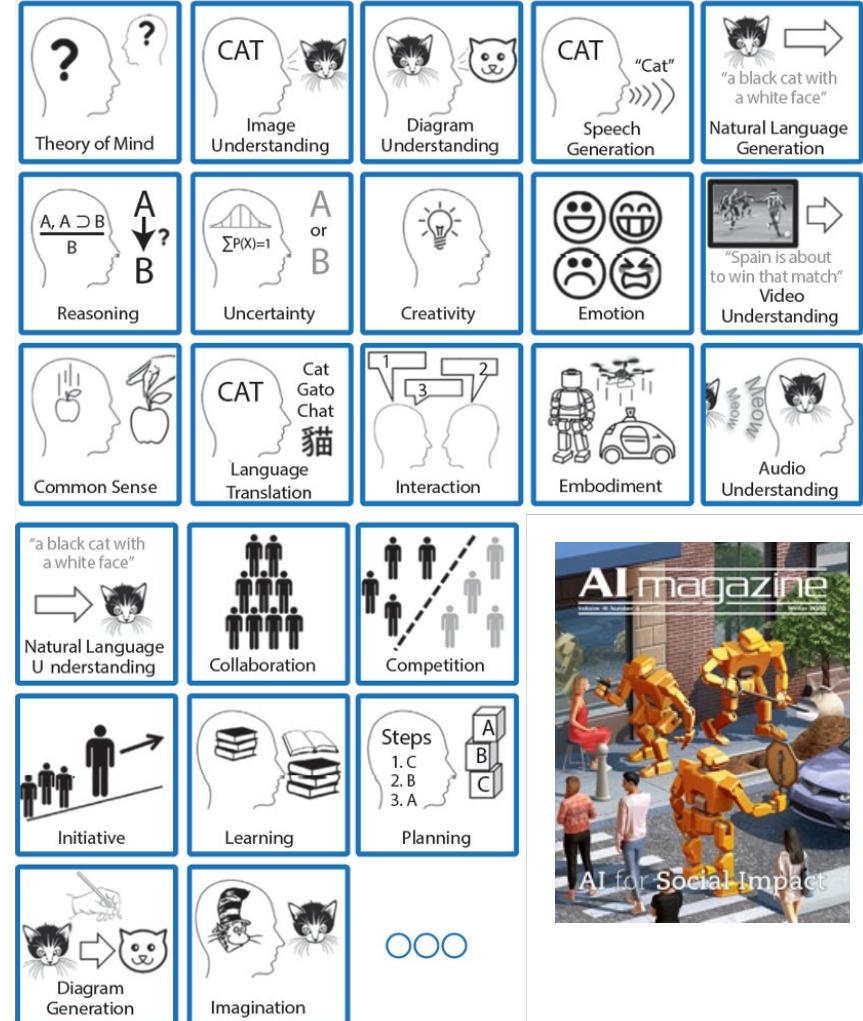
AI enabled through {DL} must be understood as any form of Machine Learning {ML} technology mimicking & automatising tasks which otherwise require

*human perception,
cognition and/or
motor skills*

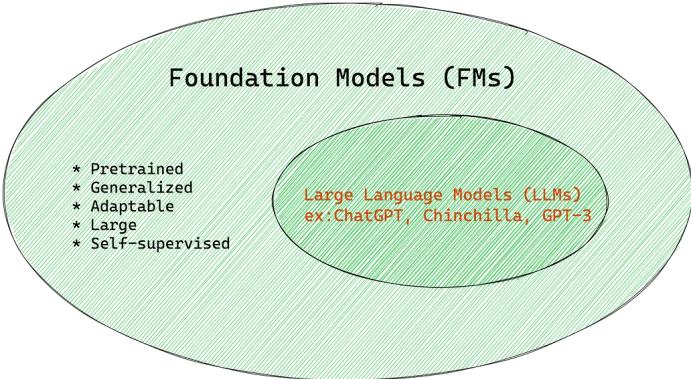


<https://robfvdw.medium.com/the-world-wide-web-ai-safari-b2e4f7f90647>

<https://doi.org/10.1609/aimag.v37i1.2643>



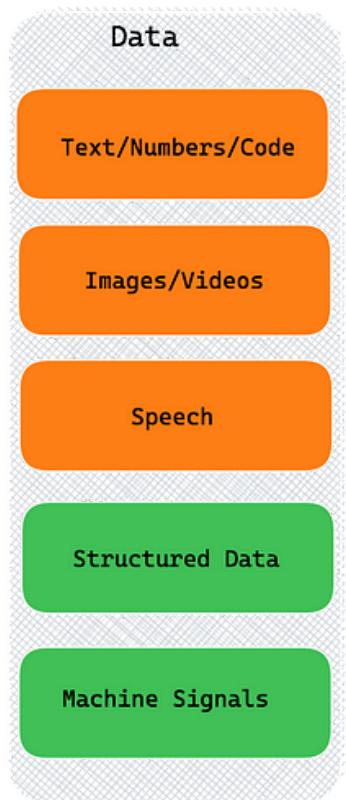
{Foundation Models}



FMs are models trained on broad data (using self-supervision at scale) that can be adapted to a wide range of downstream tasks.
<https://hai.stanford.edu/news/reflections-foundation-models>



<https://thebabar.medium.com/essential-guide-to-foundation-models-and-large-language-models-27dab58f7404>



Training →



(c) 2023 - Babar Bhatti @thebabar

Following Instructions

Question Answering

Information Extraction
Summarization

Image Captioning
Object Recognition

Transcribing Speech

Adaptation

ARTICLE

<https://doi.org/10.1038/s41467-022-30761-2>

OPEN

Towards artificial general intelligence via a multimodal foundation model

Nanyi Fei^{1,2,3}, Zhiwu Lu^{1,2}✉, Yizhao Gao^{1,2}, Guoxing Yang^{1,2}, Yuqi Huo^{2,3}, Jingyuan Wen^{1,2}, Ruihua Song^{1,2}, Xin Gao^{1,2}✉, Tao Xiang⁵, Hao Sun^{1,2}✉ & Ji-Rong Wen^{1,2,3}✉

The fundamental goal of artificial intelligence (AI) is to mimic the core cognitive activities of human. Despite tremendous success in the AI research, most of existing methods have only single-cognitive ability. To overcome this limitation and take a solid step towards artificial general intelligence (AGI), we develop a foundation model pre-trained with huge multimodal data, which can be quickly adapted for various downstream cognitive tasks. To achieve this goal, we propose to pre-train our foundation model by self-supervised learning with weak semantic correlation data crawled from the Internet and show that promising results can be obtained on a wide range of downstream tasks. Particularly, with the developed model-interpretability tools, we demonstrate that strong imagination ability is now possessed by our foundation model. We believe that our work makes a transformative stride towards AGI, from our common practice of “weak or narrow AI” to that of “strong or generalized AI”.



*“grote”
taalmodellen*

Hoe groot is Large?

Huge “foundation models” are turbo-charging AI progress
They can have abilities their creators did not foresee

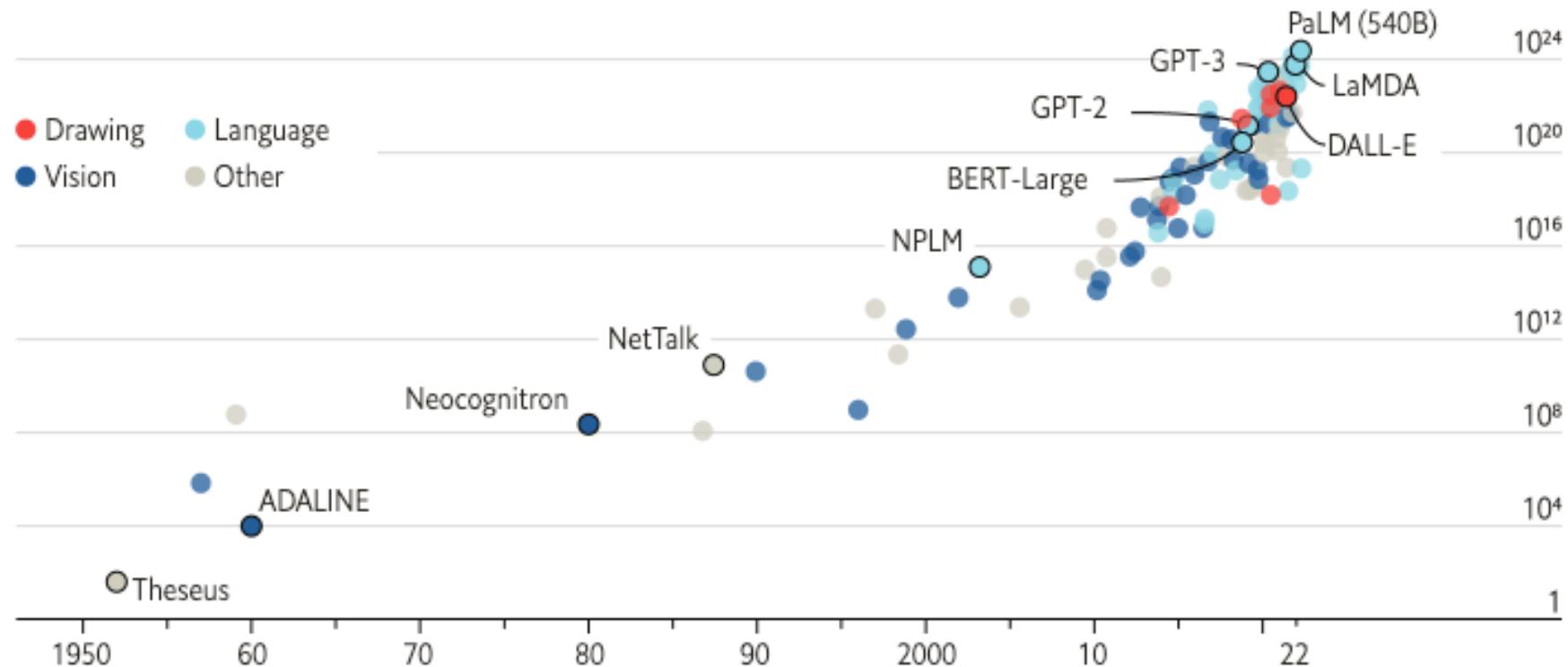
The economist
Jun 11th 2022



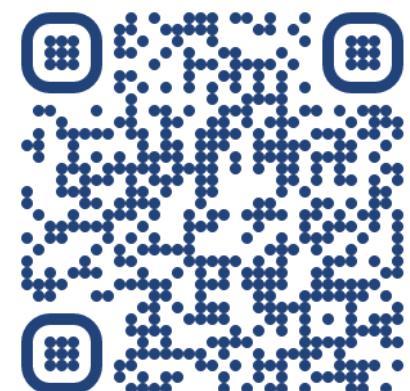
The blessings of scale

AI training runs, estimated computing resources used

Floating-point operations, selected systems, by type, log scale

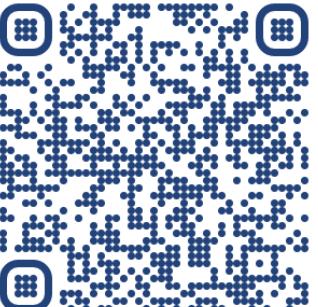


Sources: "Compute trends across three eras of machine learning", by J. Sevilla et al., arXiv, 2022; Our World in Data

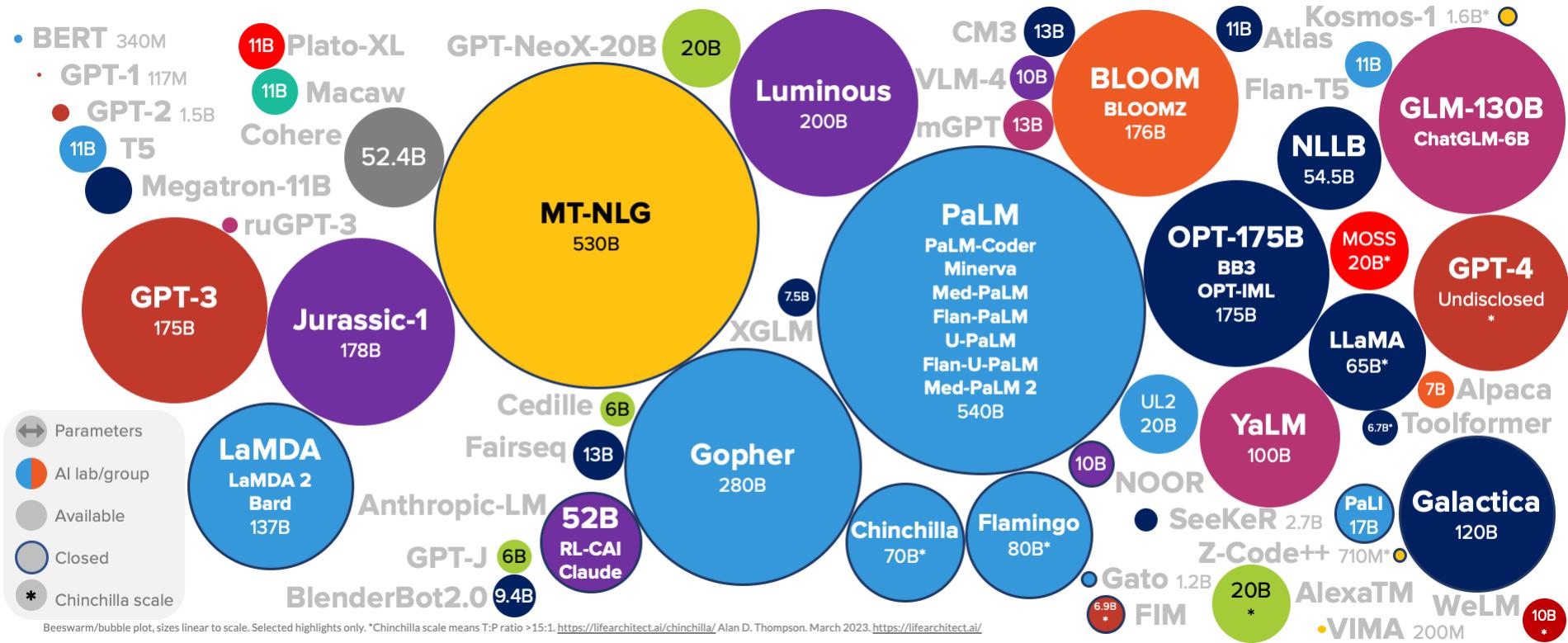


Large Language Models: Het LLM-Landschap

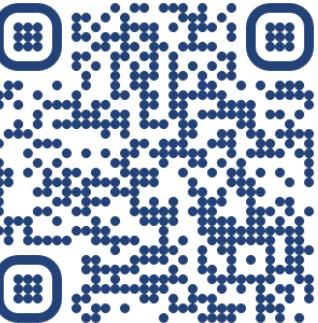
Large Language Models: Het LLM-Landschap



[https://s10251.pcdn.co/pdf/2023-
Alan-D-Thompson-AI-Bubbles-Rev-
7b.pdf](https://s10251.pcdn.co/pdf/2023-Alan-D-Thompson-AI-Bubbles-Rev-7b.pdf)

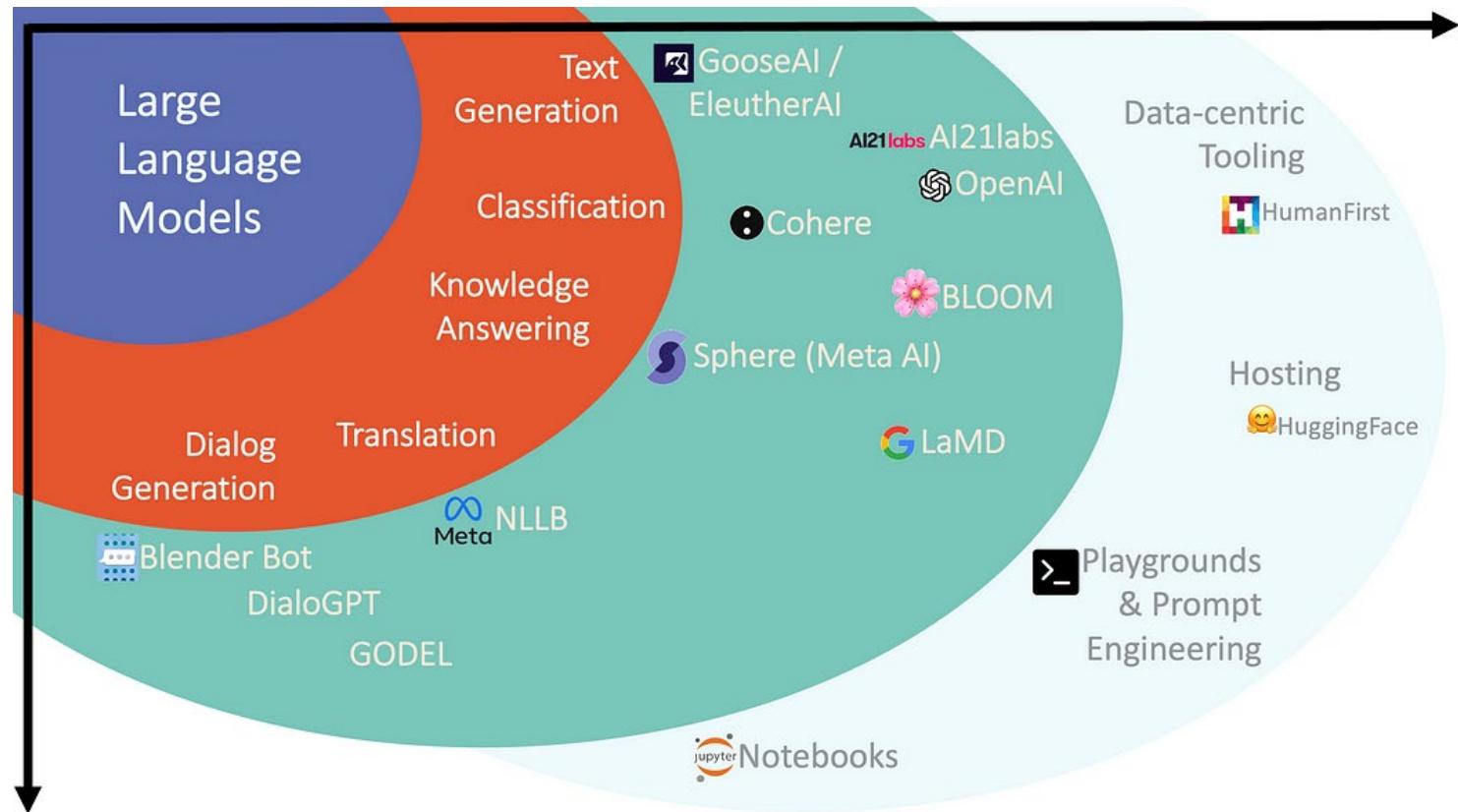


<https://docs.google.com/spreadsheets/d/105KVQW1Hx5ZAkcg8AIRjbQLQzx2wVaLI0SqUu-ir9Fs/edit#gid=1158069878>



<https://huggingface.co/spaces/adt/models-table>

Large Language Models: Het LLM-Landschap



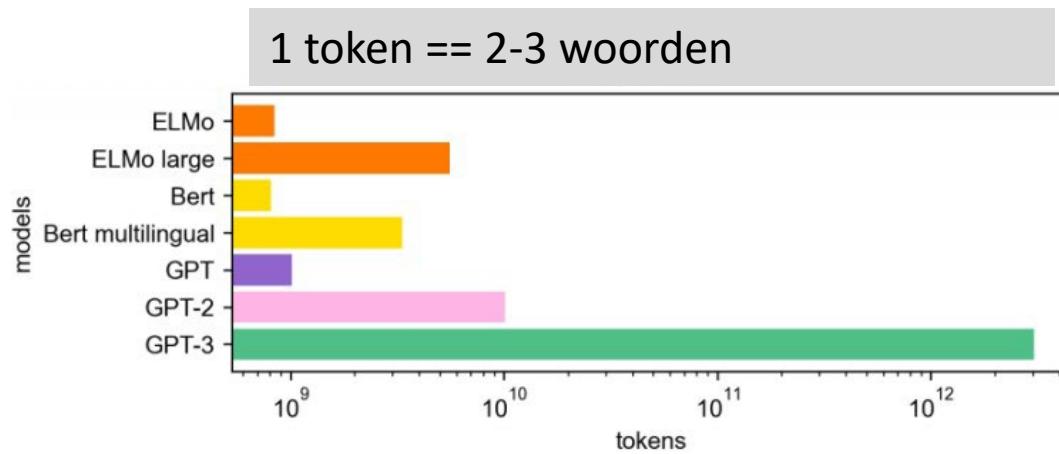
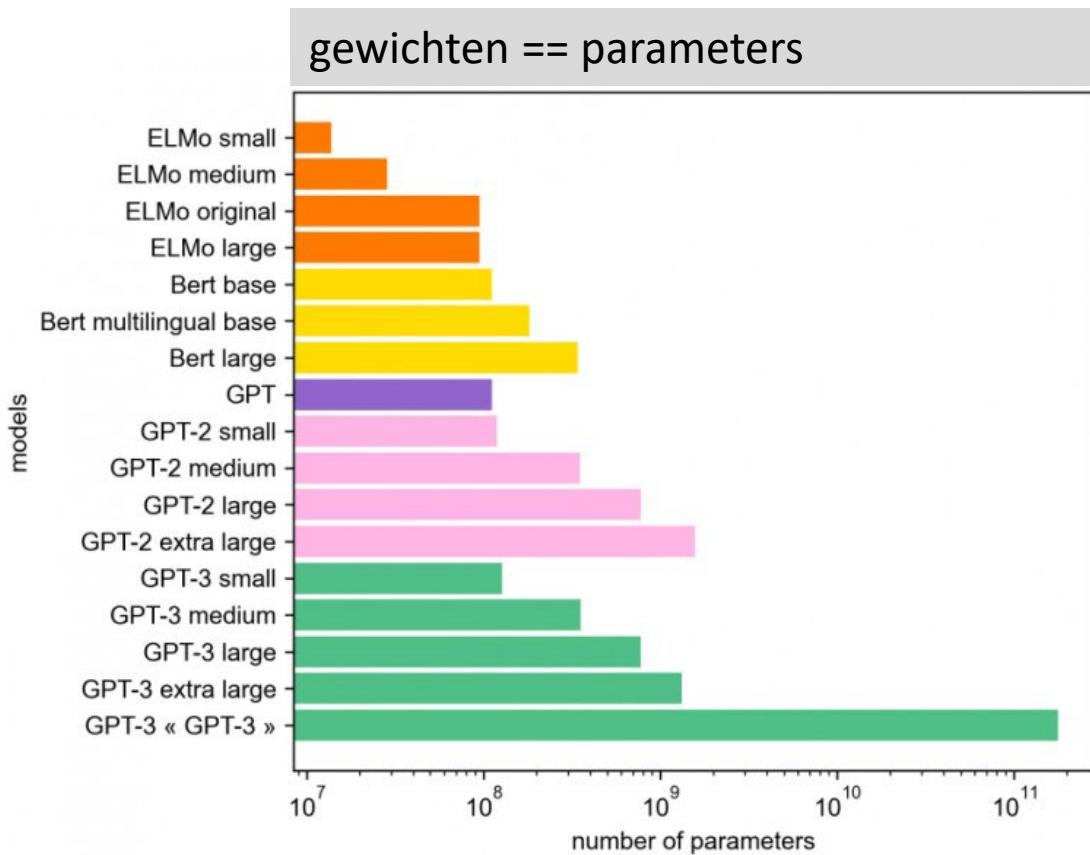
<https://cobusgreiling.medium.com/the-large-language-model-landscape-9da7ee17710b>



HOGESCHOOL
ROTTERDAM

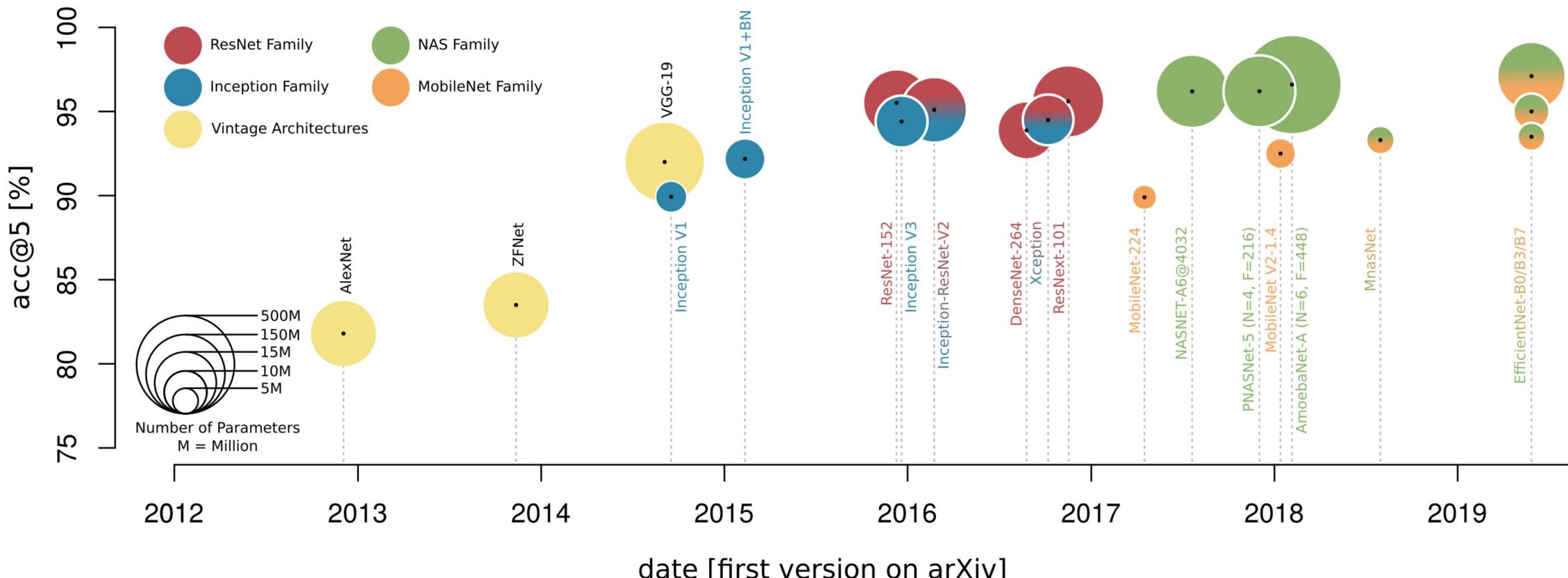
*“waarom zo
groot?”*

Grote training datasets vereist meer gewichten



[https://hellofuture.orange.com/
en/the-gpt-3-language-model-
revolution-or-evolution/](https://hellofuture.orange.com/en/the-gpt-3-language-model-revolution-or-evolution/)

Betrouwbaarheid neemt toe door meer gewichten



How to calculate the number of learnable parameters?

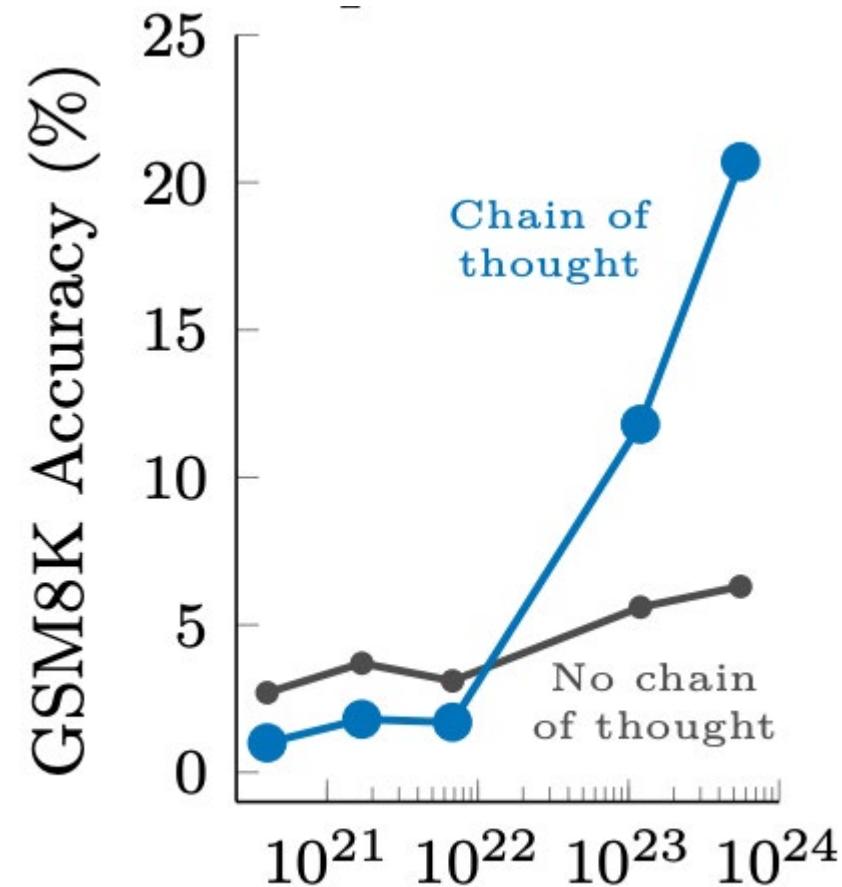
<https://doi.org/10.3390/rs12101667>

Emergente Eigenschaften

Researchers have discovered that GPT-style models have developed “emergent abilities”, like super-powers in arithmetic calculations, that were absent in smaller models.



<https://medium.com/@arushi.prakash307/gpt-style-models-are-unexpectedly-developing-super-powers-4df002186a75>



natural language processing [nlp]

“natuurlijke
taalverwerking”

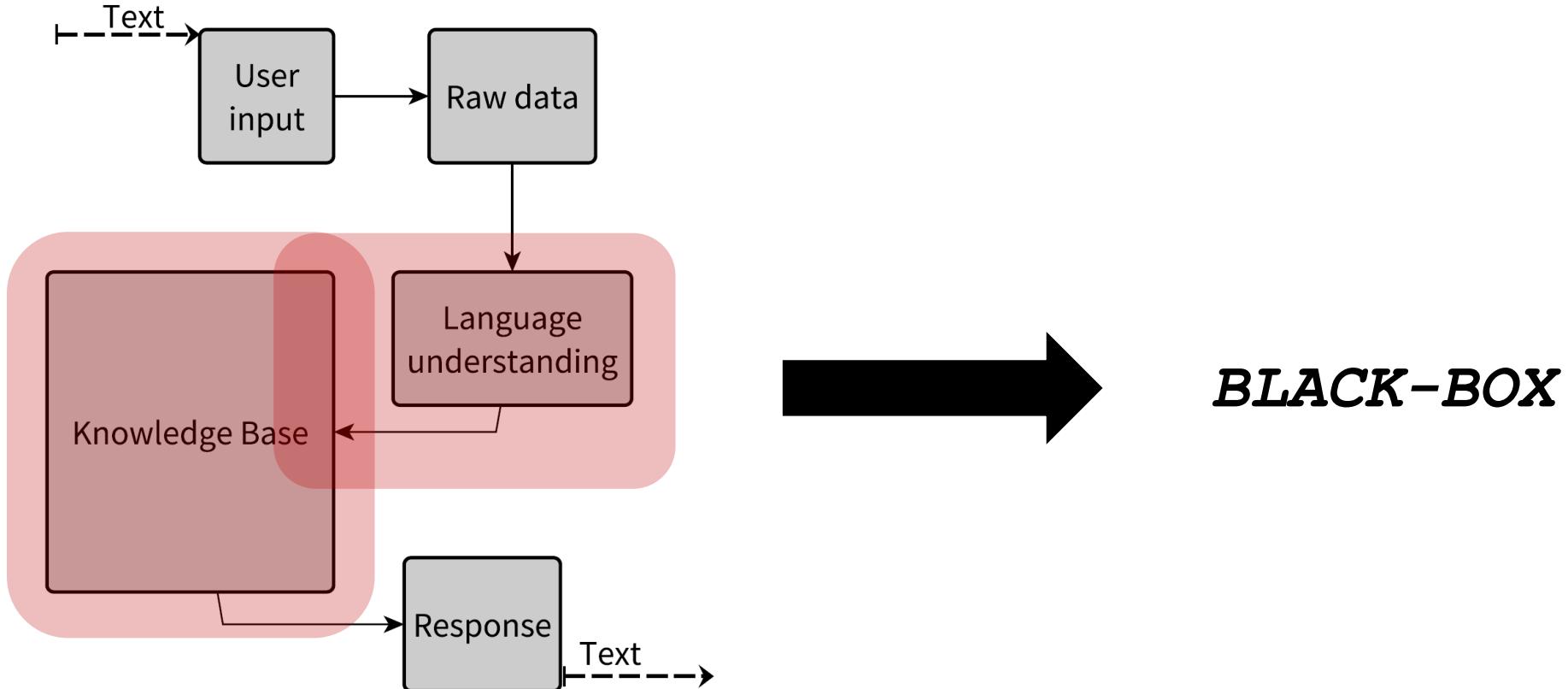
---*Natural Language Processing {NLP}*---

Natuurlijke taalverwerking

begrijpen (NLU) + genereren (NLG)

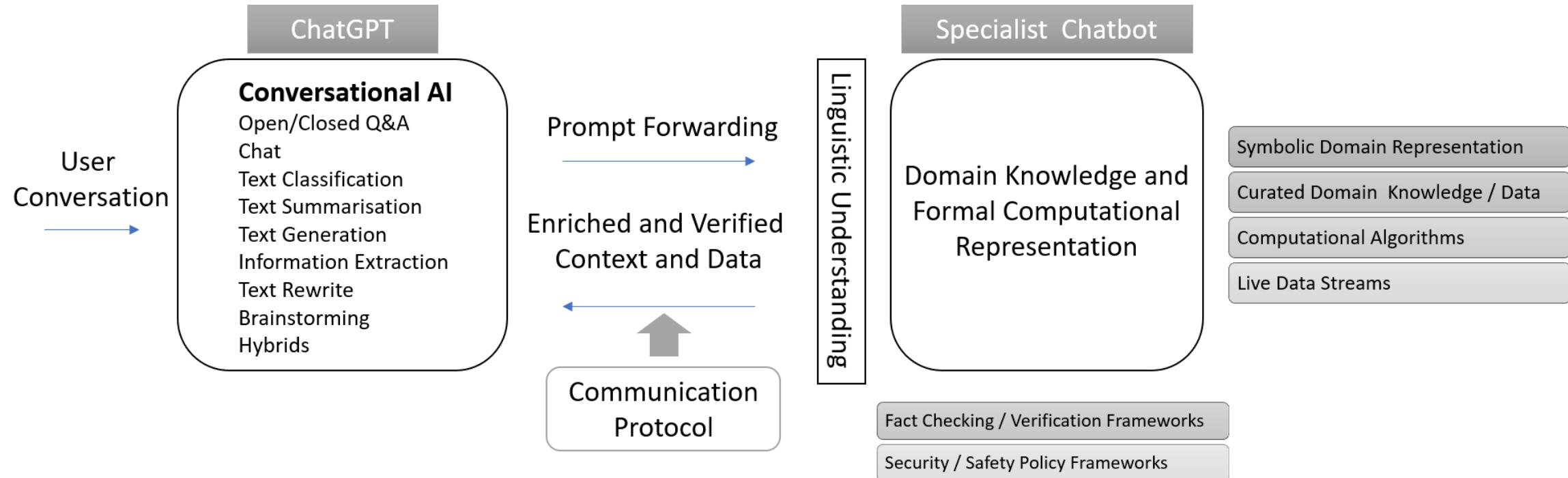
NLP richt zich op het ontwerpen van **conversationele agenten** die vloeiende, coherent en betekenisvolle taal produceren voor menselijke consumptie.

ChatGPT is een Conversationele *tekst-in/tekst-uit* AI-agent



Sánchez-Díaz, X., Ayala-Bastidas, G., Fonseca-Ortiz, P., & Garrido, L. (2018).
A knowledge-based methodology for building a conversational chatbot as an
intelligent tutor. https://doi.org/10.1007/978-3-030-04497-8_14

Fine-tuning vs Prompt engineering



Conferences > 2023 IEEE International Conference on Big Data and Cloud Computing (BCC)

ChatGPT and Generative AI Guidelines for Addressing Academic Integrity and Augmenting Pre-Existing Chatbots

Publisher: IEEE

Cite This

PDF

Daswin De Silva ; Nishan Mills ; Mona El-Ayoubi ; Milos Manic ; Damminda Alahakoon All Authors

635
Full
Text Views



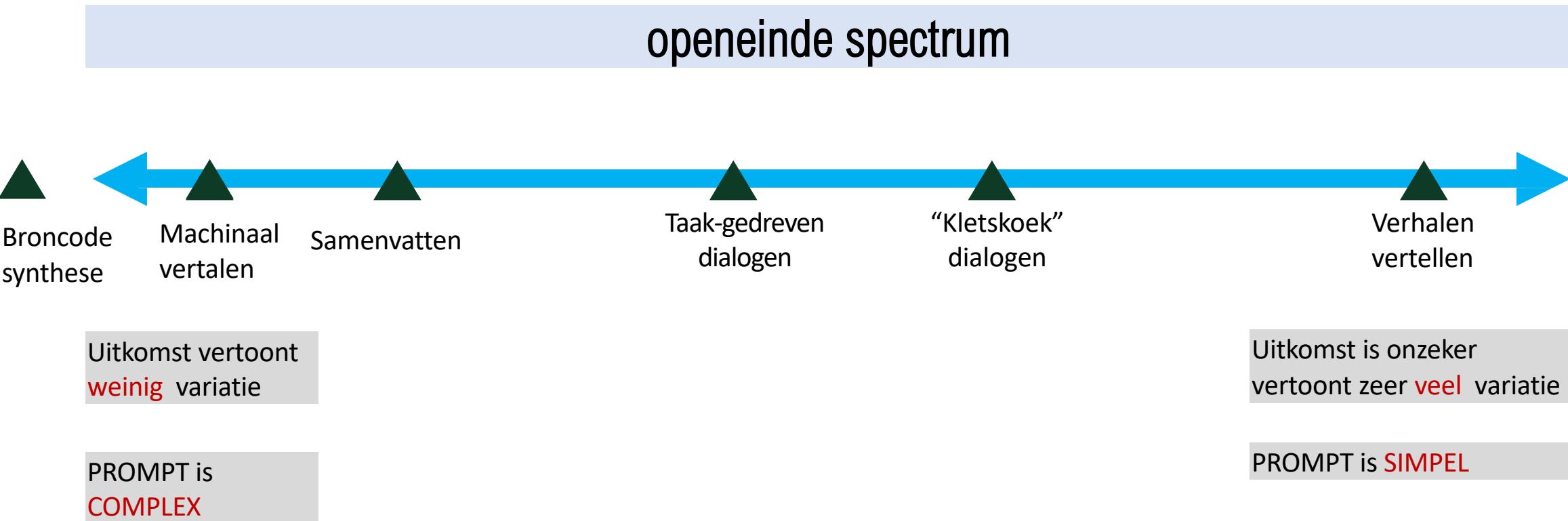
Transformeren

In de context van natuurlijke taalverwerking verwijst "transformeren" naar het proces waarbij de inputreeks van "tokens" wordt omgezet in een reeks gecontextualiseerde representaties.

Natuurlijke taal vormt een tekst-in/tekst-uit “Sequentie naar Sequentie” [Seq2Seq] Transformeer Probleem

	INPUT	OUTPUT
Machine Translation	"De kat is zwart"	The Cat is black
Named Entity Recognition	"Trebor werkt bij IBM"	[<PERSOON>, <X>, <X>, <ORG>]
POS Tagging	"Trebor werkt bij IBM"	[Zelfstn., Werkwoord, Voorzetsel, Zelfstn.]
Speech Recognition		"Ik ben menselijk"
Answering chatbot	Wat is het weer in Rotterdam?"	25 graden en Zonning
DNA Analysis	"cacgcaaaccctttcaggcttcggcggtccct caagctgtccgatgcgcagactacttgtcg.."	{promotor.(0,67), exon:(74,103)}
	Tekst-in woordvolgorde	Tekst-uit woordvolgorde

Categorisering NLG Taken



ChatGPT: Use Cases

CODE GENERATION

JU Give me code for the Dijkstra algorithm in Python

Q Here is the code for the Dijkstra algorithm in Python:

```
from collections import defaultdict
import heapq

def dijkstra(graph, source):
    # Initialize distances and predecessor for each vertex
    distances = defaultdict(lambda: float('inf'))
    predecessor = defaultdict(lambda: None)
    distances[source] = 0

    # Initialize priority queue with all vertices and their distances from
    # the source
    priority_queue = [(0, source)]
    while priority_queue:
        # Extract vertex with minimum distance from the queue
        distance, vertex = heapq.heappop(priority_queue)

        # Update distances and predecessors for all neighbors of the vertex
        for neighbor, weight in graph[vertex]:
            if distances[neighbor] > distance + weight:
                distances[neighbor] = distance + weight
                predecessor[neighbor] = vertex
                heapq.heappush(priority_queue, (distances[neighbor], neighbor))

    return distances, predecessor
```

CONCEPTUAL QUESTIONS

JU What are algorithmic solutions for solving congestion

Q Congestion occurs when a network becomes overloaded, resulting in reduced capacity and slower performance. There are several algorithmic solutions for solving congestion in networks, including:

1. Routing algorithms: These algorithms aim to route traffic around congested areas or to balance the load among different routes. Some examples include Dijkstra's algorithm, Bellman-Ford algorithm, and Link-State Routing Protocol.
2. Flow control algorithms: These algorithms aim to regulate the flow of traffic in order to prevent congestion. Examples include window-based flow control (such as TCP), rate-based flow control (such as the leaky bucket algorithm), and priority-based flow control (such as strict priority queuing).
3. Resource allocation algorithms: These algorithms aim to allocate resources (such as bandwidth or processing power) to different tasks or users in a way that minimizes congestion. Examples include max-min fairness, proportional fairness, and weighted fair queueing.
4. Traffic engineering algorithms: These algorithms aim to optimize the use of network resources in order to reduce congestion. Examples include QoS routing, traffic grooming, and multicast tree construction.
5. Congestion pricing algorithms: These algorithms aim to incentivize users to reduce their demand for network resources by charging them for congestion. Examples include tolls for using certain roads or time-of-day pricing for using the internet.

MATH QUESTIONS

Q Find the derivative of the function $f(x) = 3x^2 + 2x - 1$. Describe your working

Q To find the derivative of the function $f(x) = 3x^2 + 2x - 1$, we need to use the rules of differentiation. The derivative of a function is a measure of how the function is changing at a particular point.

The derivative of a constant term (such as -1 in this case) is always 0 , so we can ignore that part of the function. The derivative of a term with a coefficient of 1 (such as $2x$ in this case) is just the coefficient times the variable (in this case, $2x$), so the derivative of $2x$ is 2 .

The derivative of a term with a coefficient other than 1 (such as $3x^2$ in this case) is the coefficient times the derivative of the term with the coefficient of 1 (in this case, 3 times x), so the derivative of $3x^2$ is $3 \cdot 2x = 6x$.

Now that we have found the derivative of each term in the function, we can add them up to get the derivative of the whole function:

$$f'(x) = 3x + 2 = 6x + 2$$

This is the final answer. The derivative of the function $f(x) = 3x^2 + 2x - 1$ is $f'(x) = 6x + 2$.

Conferences > 2023 IEEE Global Engineering ... ⓘ

Engineering Education in the Era of ChatGPT: Promise and Pitfalls of Generative AI for Education

Publisher: IEEE

Cite This



Junaid Qadir All Authors

4

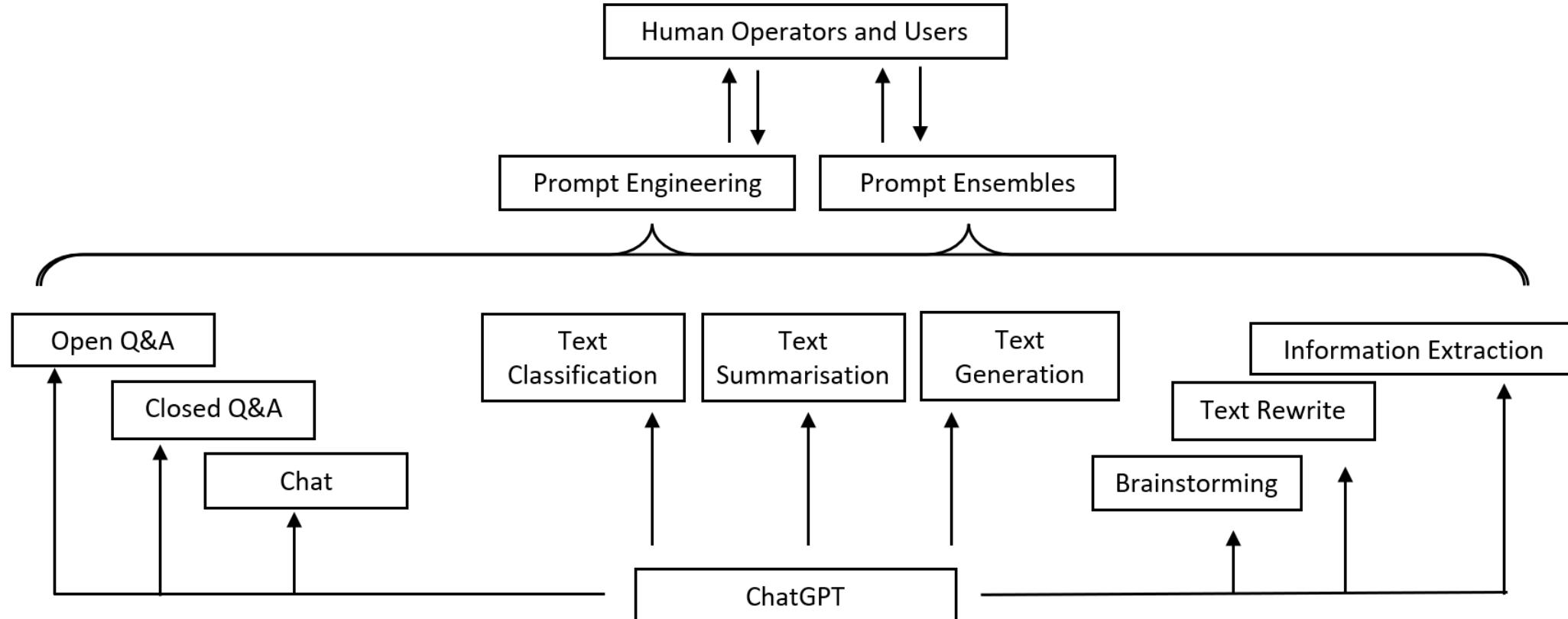
Cites in
Papers

3380

Full
Text Views



ChatGPT: Use Cases



Conferences > 2023 IEEE International Confe... ⓘ

ChatGPT and Generative AI Guidelines for Addressing Academic Integrity and Augmenting Pre-Existing Chatbots

Publisher: IEEE

Cite This

PDF

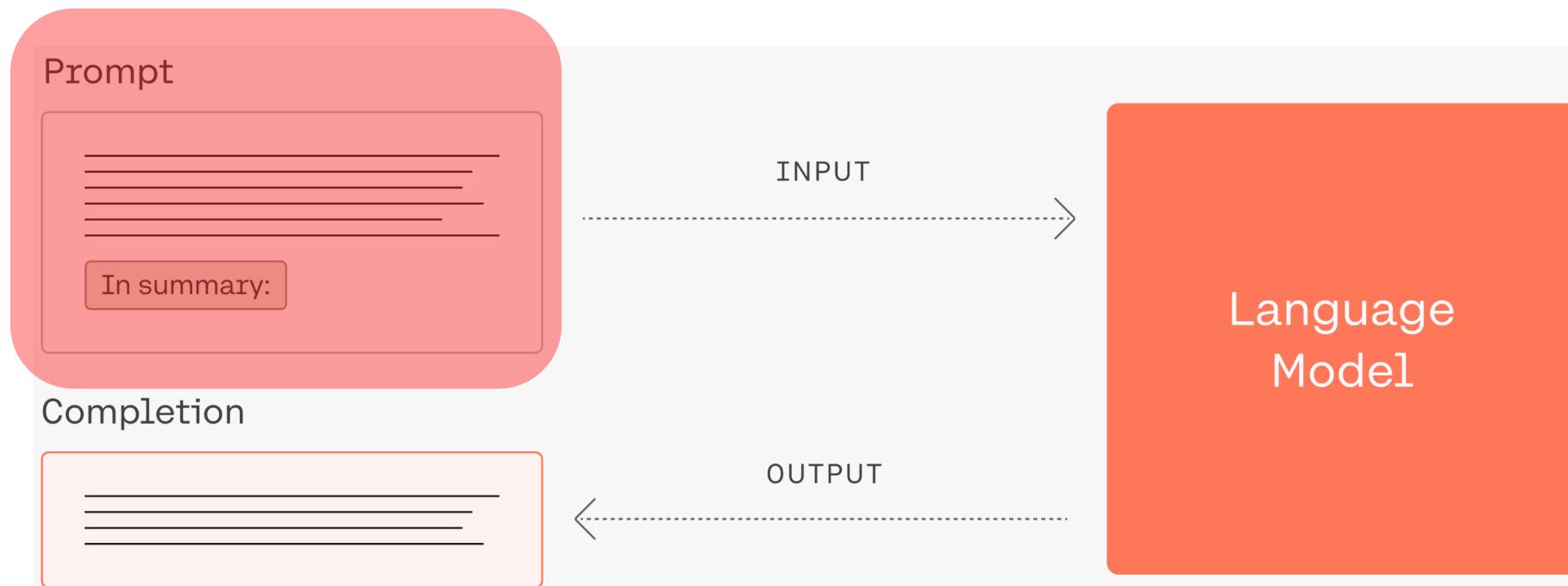
Daswin De Silva ; Nishan Mills ; Mona El-Ayoubi ; Milos Manic ; Damminda Alahakoon All Authors

635
Full
Text Views



“Prompt Engineering
vs
Fine-tuning”

*Conversationele AI-agenten worden aangestuurd via “**prompts**”*



<https://docs.cohere.com/docs/prompt-engineering>

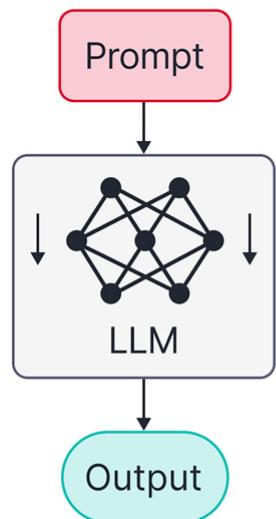


<https://docs.cohere.com/docs/introduction-to-large-language-models>

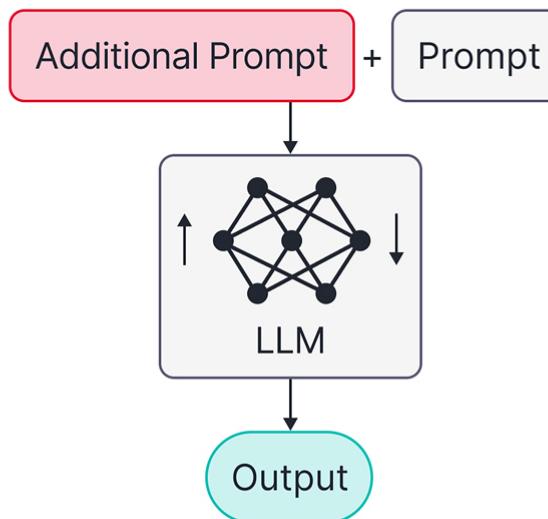
Vier manieren om LLM's te benutten



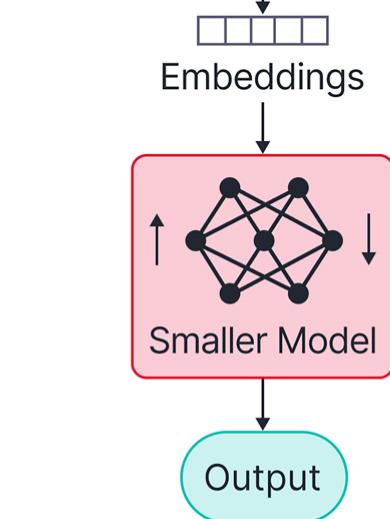
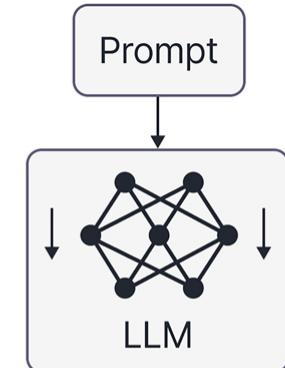
HOGESCHOOL
ROTTERDAM



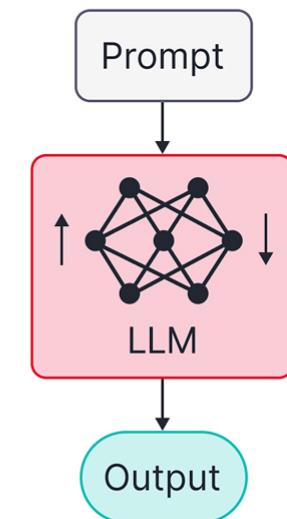
(a) Prompt Engineering



(b) Prompt Tuning



(c) LLM Embeddings as input



(d) Model Fine Tuning

PROMPT

Een prompt is een natuurlijke taal instructie gericht aan een **conversationele AI-agent** met als doel om de werking ervan te verbeteren of te verfijnen.

PROMPT-ENGINEERING

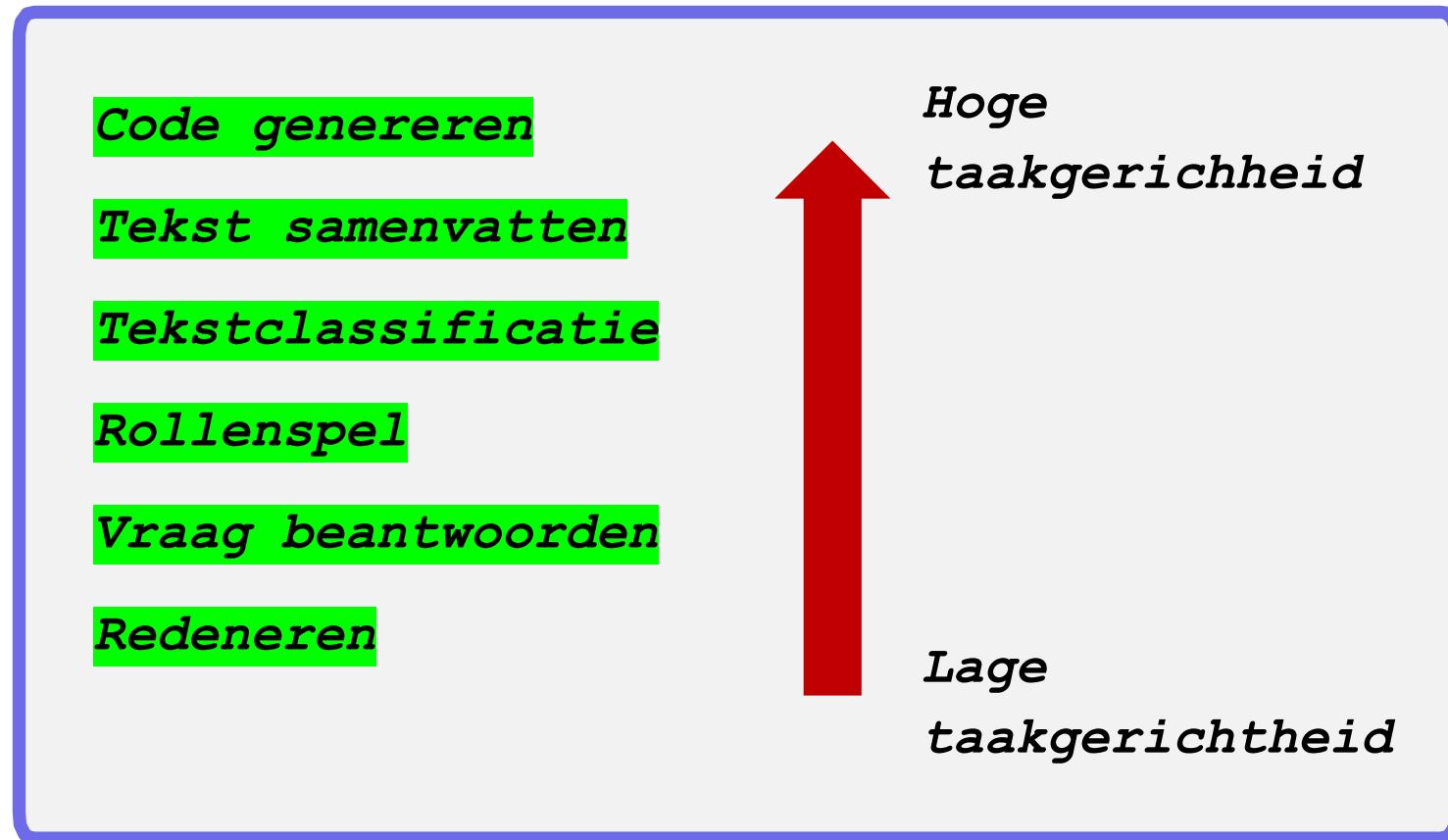
Het creatieve proces van het schrijven van een effectief ***prompt-recept*** wordt in het Engels "***prompt engineering***" genoemd.

Het schrijven van prompt-recepten

---pseudo-Code---

is een talige manier van het programmeren van "bevroren" voorgetraind taalmodellen.

Taakgerichtheid van prompts



Wat is het belang van Prompt Recepten Schrijven?

*Sturen van de mate van taakgerichtheid door
reduceren van variatie in het antwoord zodat de
kans groter wordt dat de uitkomst correct is.*

Prompt Recept Structuur

Een prompt is opgebouwd uit de volgende elementen:

Instructie(s)

Context

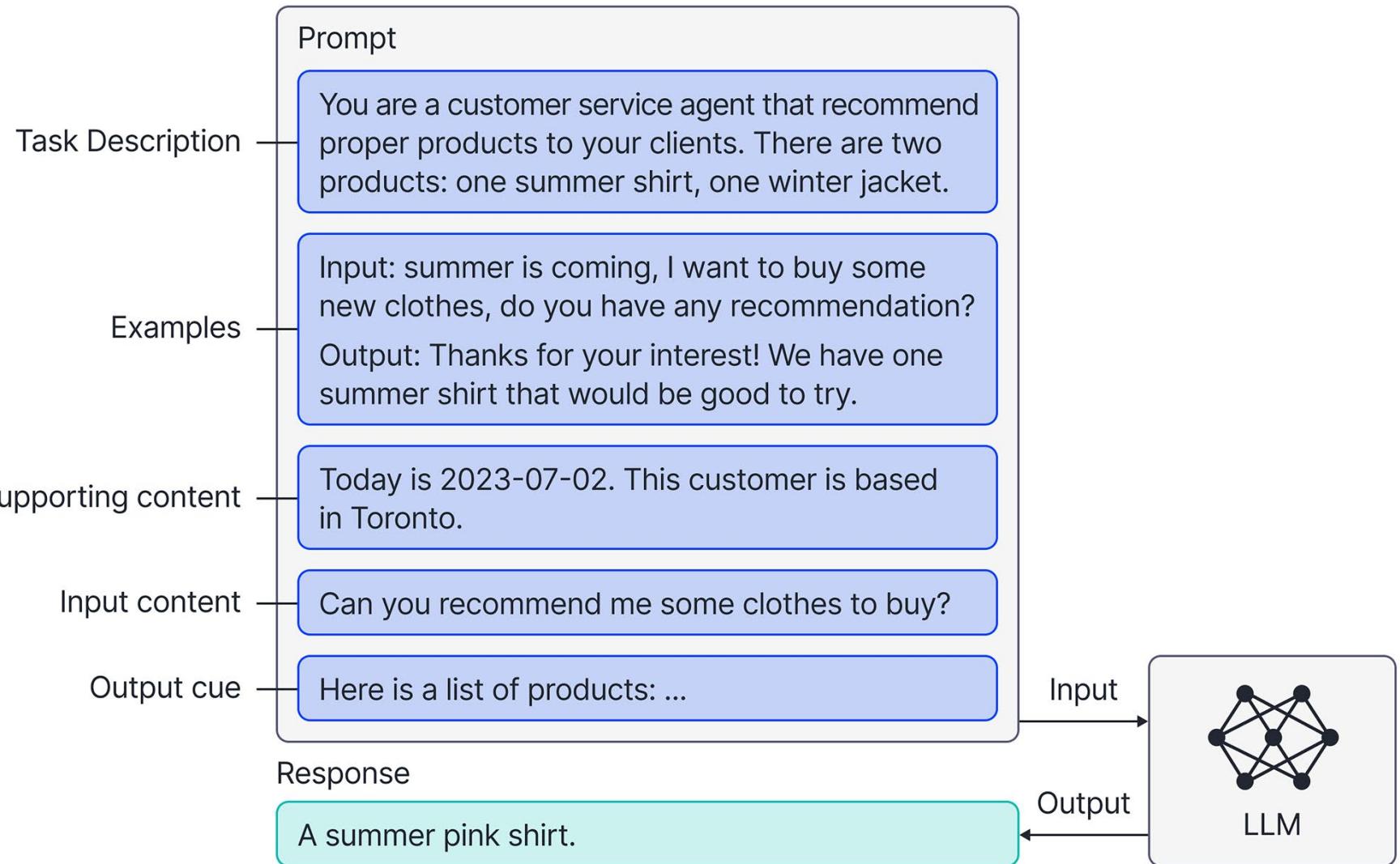
Invoergegevens

Uitvoer-indicator

Classificeer de onderstaande tekst als neutraal, negatief of positief

Text: Ik vond het eten wel zozo.

Sentiment:





<https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/prompt-engineering>

Microsoft | Learn Documentation Training Credentials Q&A Code Samples Assessments Shows

Azure Product documentation ▾ Architecture ▾ Learn Azure ▾ Develop ▾ Resources ▾ Troubleshooting

Filter by title

Azure OpenAI Service Documentation

Overview

- What is Azure OpenAI?
- Quotas and limits
- Models
- Legacy models
- Pricing ⓘ
- What's new
- Azure OpenAI FAQ

Quickstarts

Concepts

- Abuse monitoring
- Content filtering
- Embeddings
- Red teaming large language models (LLMs)

Intro to prompt engineering

- Prompt engineering techniques
- System message templates
- Using your data (preview)

How-to

Tutorials

Responsible AI

Reference

Resources

Learn / Azure / AI Services /

Introduction to prompt engineering

Article • 07/18/2023 • 2 contributors ⚙️ Feedback

In this article

- Basics
- Best practices
- Space efficiency
- Next steps

GPT-3, GPT-3.5, and GPT-4 models from OpenAI are prompt-based. With prompt-based models, the user interacts with the model by entering a text prompt, to which the model responds with a text completion. This completion is the model's continuation of the input text.

While these models are extremely powerful, their behavior is also very sensitive to the prompt. This makes prompt construction an important skill to develop.

Prompt construction can be difficult. In practice, the prompt acts to configure the model weights to complete the desired task, but it's more of an art than a science, often requiring experience and intuition to craft a successful prompt. The goal of this article is to help get you started with this learning process. It attempts to capture general concepts and patterns that apply to all GPT models. However it's important to understand that each model behaves differently, so the learnings may not apply equally to all models.

Basics

This section covers the basic concepts and elements of GPT prompts.

Text prompts are how users interact with GPT models. As with all generative language models, GPT models attempt to produce the next series of words that are most likely to follow from the previous text. It's as if we're saying *What is the first thing that comes to your mind when I say <prompt>?*. The examples below demonstrate this behavior. Given the first words of famous content, the model is able to accurately continue the text.

Prompt	Completion
Four score and seven years ago our	fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal. [...]
"Call me Ishmael."	"Some years ago—never mind how long precisely—having little o no money in my purse, and nothing particular to interest me on shore, I thought I would sail about a little and see the watery part of the world." [...]



HOGESCHOOL
ROTTERDAM

<https://github.com/promtslab/Awesome-Prompt-Engineering>

Awesome-Prompt-Engineering Public

Watch 49 Fork 186 Star 2.3k

main 1 branch 0 tags Go to file Add file Code

monk1337 Merge pull request #9 from ArtificialZeng/main ... 45ddcda 2 weeks ago 116 commits

_source Create data_format.py 8 months ago

LICENSE Initial commit 8 months ago

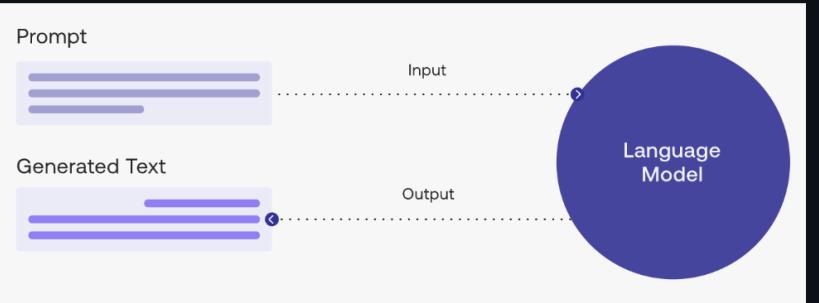
README.md Merge pull request #9 from ArtificialZeng/main 2 weeks ago

code-of-conduct.md Create code-of-conduct.md 8 months ago

contributing.md Update contributing.md 8 months ago

README.md

Awesome Prompt Engineering 🤖



This repository contains a hand-curated resources for Prompt Engineering with a focus on Generative Pre-trained Transformer (GPT), ChatGPT, PaLM etc

Prompt Engineering Course is coming soon..

awesome License Apache 2.0 PRs welcome Discord Community Open in Colab

Table of Contents

- Papers
- Tools & Code
- APIs

About

This repository contains a hand-curated resources for Prompt Engineering with a focus on Generative Pre-trained Transformer (GPT), ChatGPT, PaLM etc

[discord.gg/m88xfYMbK6](#)

machine-learning text-to-speech
deep-learning prompt openai
prompt-toolkit gpt text-to-image
few-shot-learning text-to-video gpt-3
prompt-learning prompt-tuning
prompt-engineering prompt-generator
promptengineering prompt-based-learning
chatgpt chatgpt-api

Readme Apache-2.0 license
Code of conduct
Activity 2.3k stars
49 watching 186 forks
Report repository

Releases

No releases published

Packages

No packages published

Contributors 11



Prompt recept ontwerp overwegingen

Agent' dialoogsturing aanwijzingen / hints

Constructive Reasoning

(Graesser, 2016; Nguyen & Santagata, 2020)

Transactivity

(Chi & Wylie, 2014; Fiacco & Rose', 2018; Resnick et al., 2010;
Teasley, 1997)

Social Expressions

(Sebo, Stoll, Scassellati, & Jung, 2020)

Rollenspel & Identiteit

Inzicht Iteraties

Theories & Cultural Probes

User testing

Prompt recept ontwerptechnieken
gebaseerd op fine-tuning van het
onderliggende taal-model

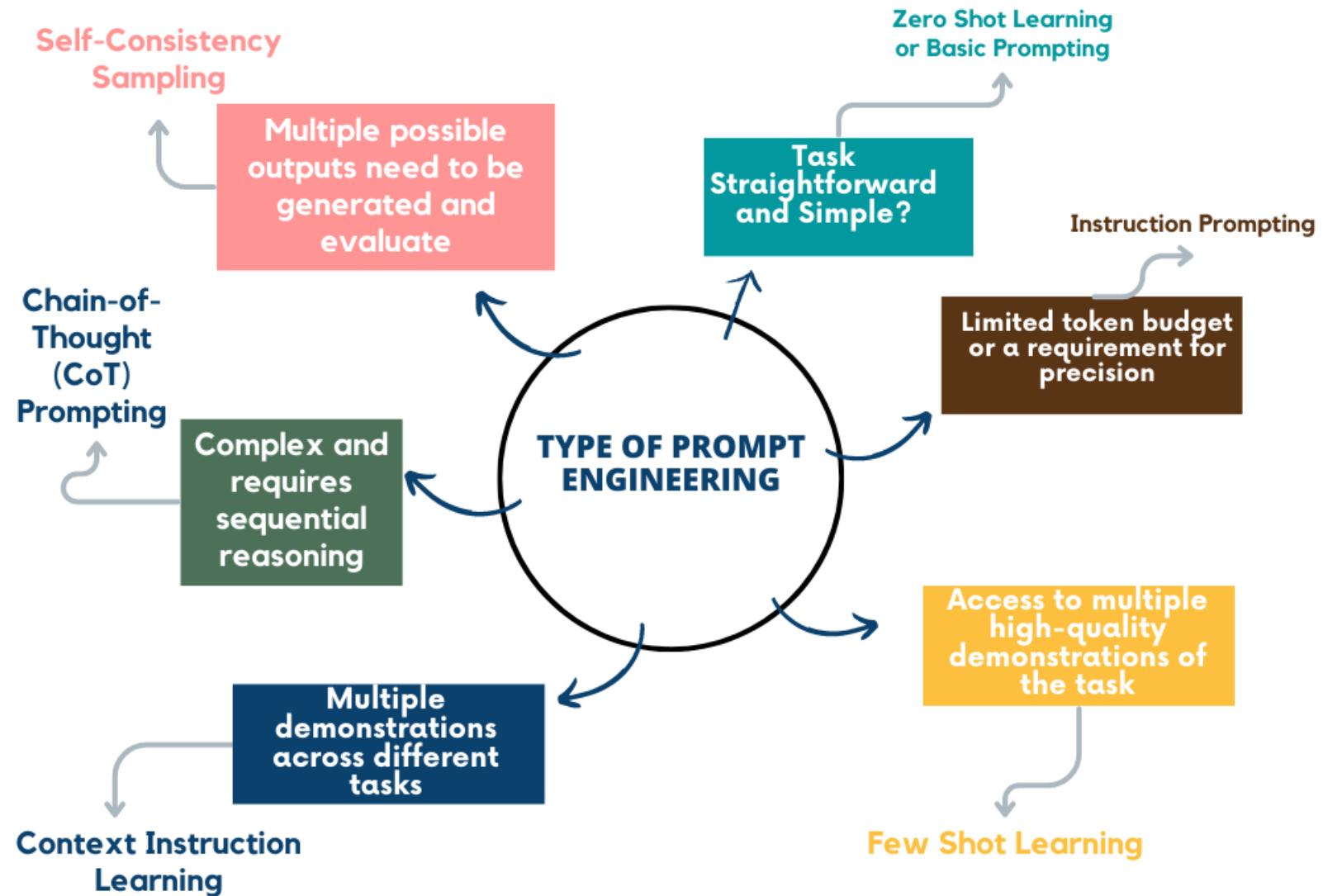
Few-shot prompts (**In Context Leren**)

Chain-of-thought (**CoT**) reasoning

Self-Consistency Sampling

Knowledge Generation Prompting

ReAct



Fine-tuning

De meest geavanceerde methode om een LLM te specialiseren is fine-tuning.

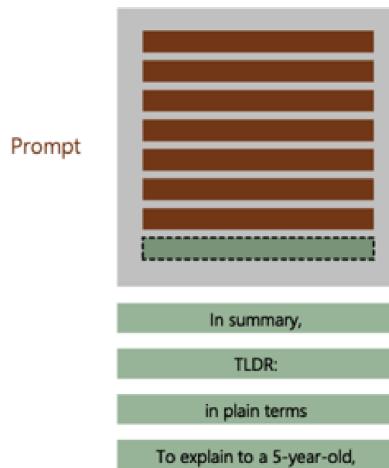
Je kunt wat winst behalen door supervised fine-tuning en dan is het moeilijkste deel misschien wel RLHF (het optimaliseren van een taalmodel met menselijke feedback), wat ons ChatGPT heeft gebracht. Karpathy raadde ten tijde van GPT sterk af om RLHF te doen, meestal heeft OpenAI het in productie genomen.

Google deed er vrij lang over om zijn modellen te RLHF-en, ze hadden al grote modellen maar misschien geen veilige modellen.

Prompting & Fine Tuning

1) Prompt Engineering

1.1) Tuning the task description



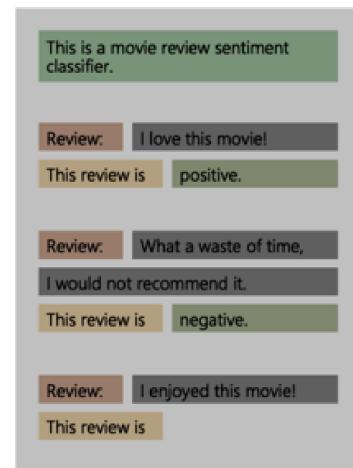
Task description

Example #1

Example #2

Current output

1.2) Adding examples in the prompt



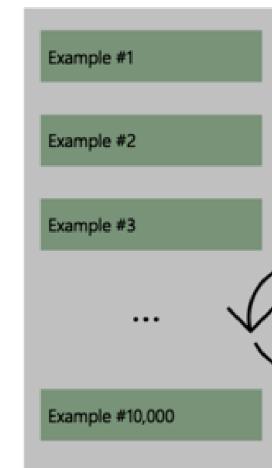
Example input #1

Example output #1

Example input #2

Example output #2

2) Finetune on many examples

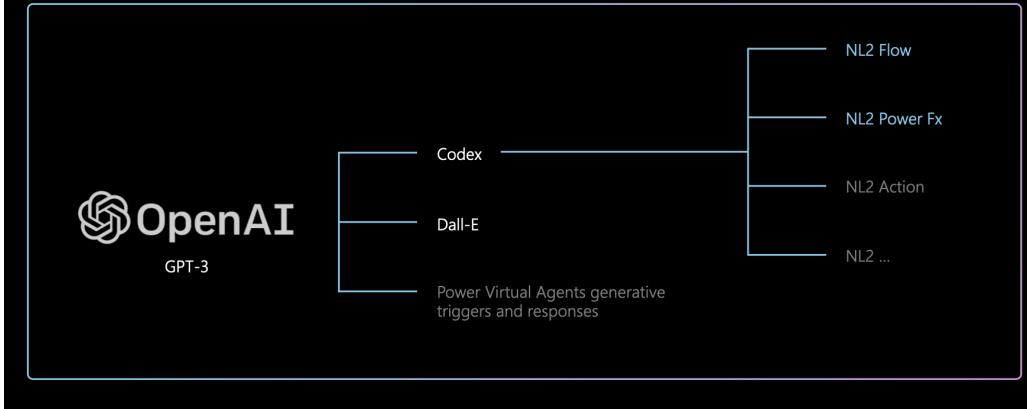


Custom
Language
Model

Train

<https://docs.cohere.com/docs/constructing-prompts>

When to fine tune versus taking base models



<https://www.linkedin.com/pulse/mastering-generative-ai-interactions-guide-in-context-pradeep-menon>

GPT-3 is a state-of-the-art large language model that was fine-tuned to create Codex, a specialized language model designed for generating code.

To accomplish this, the base GPT-3 model was **fine-tuned** on a diverse and extensive dataset that included programming-related texts and code examples from sources such as GitHub repositories, Stack Overflow, and various technical documentation.

As a result, Codex has an improved ability to understand and generate code in multiple programming languages and to solve programming-related tasks more effectively.

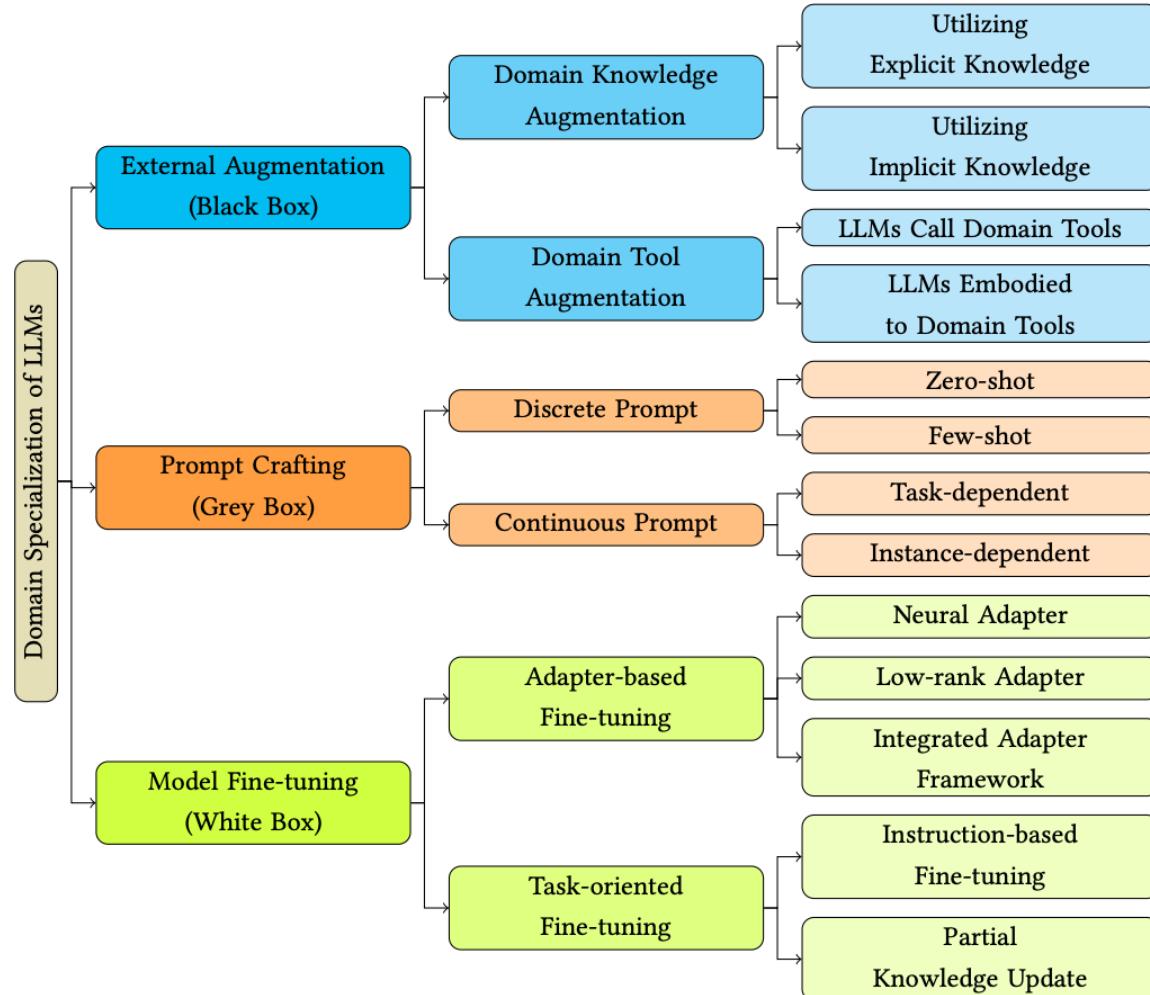
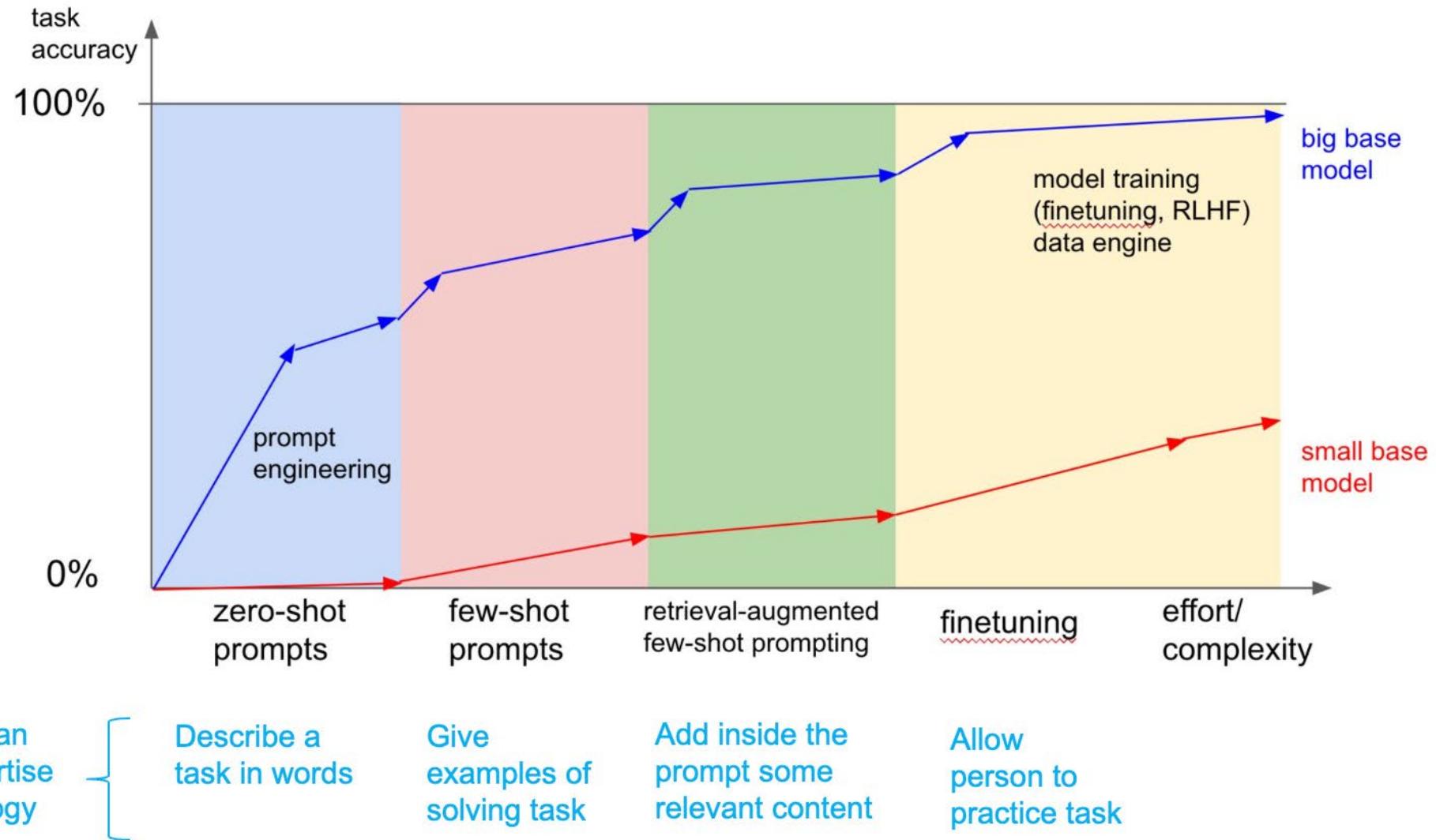
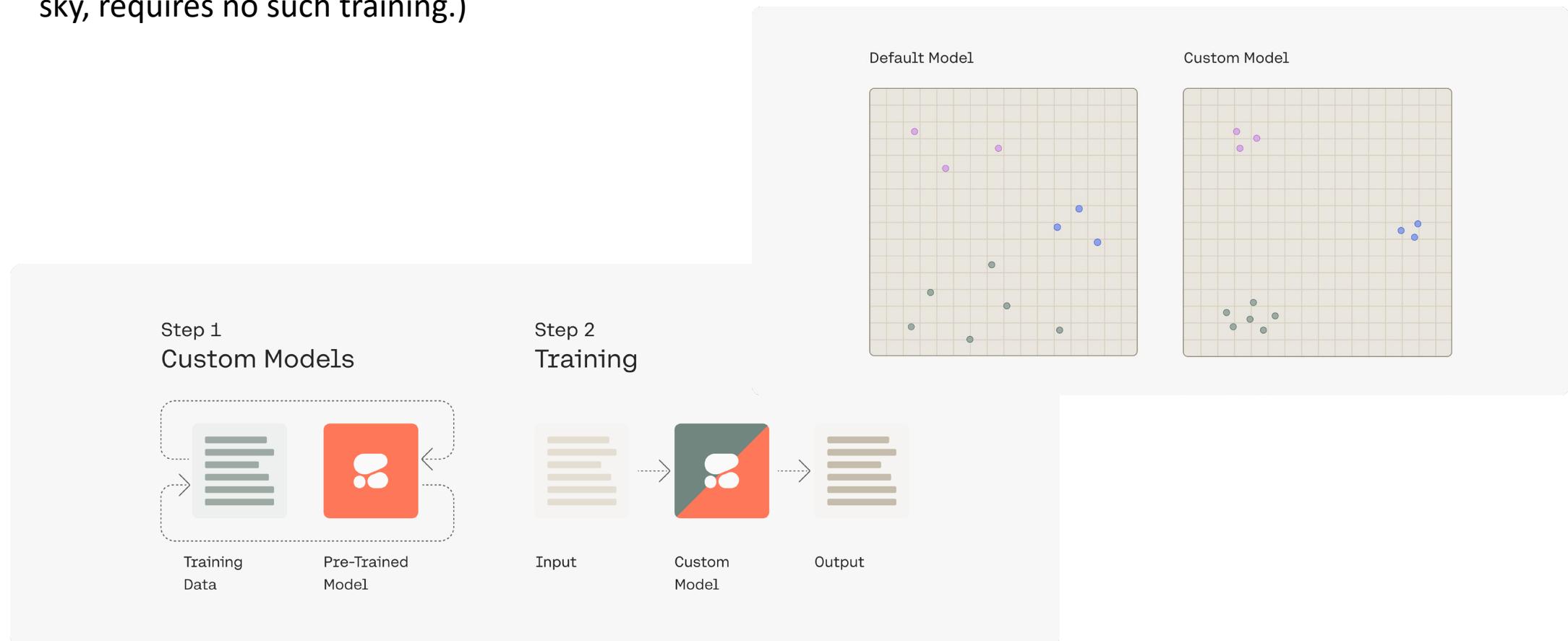


Fig. 1. The taxonomy for current techniques on LLM domain specialization.



When to Train a Custom Model

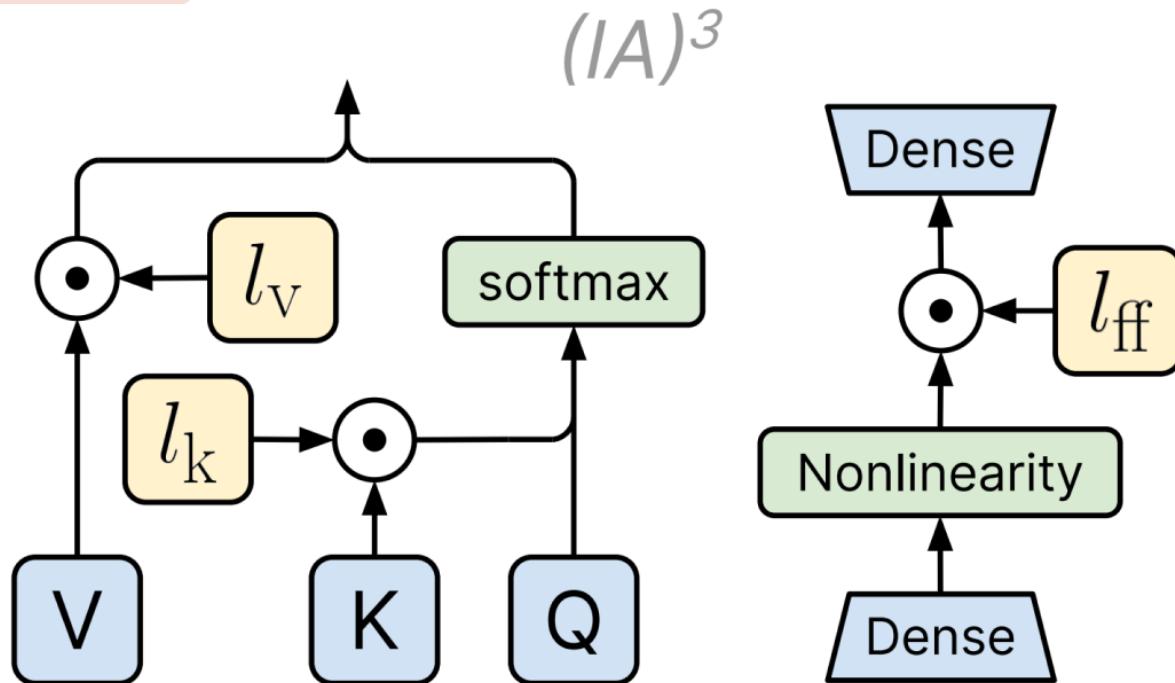
Training large language models is recommended when you want to teach the model a new task, or when you want to utilize specific domain knowledge, such as the different gaits of a horse or your company's unique knowledge base (common knowledge, like the color of the sky, requires no such training.)



Efficient LLM Finetuning with T-Few

We delve into the concept of T-Few finetuning, explore its benefits, and explain our implementation workflow.

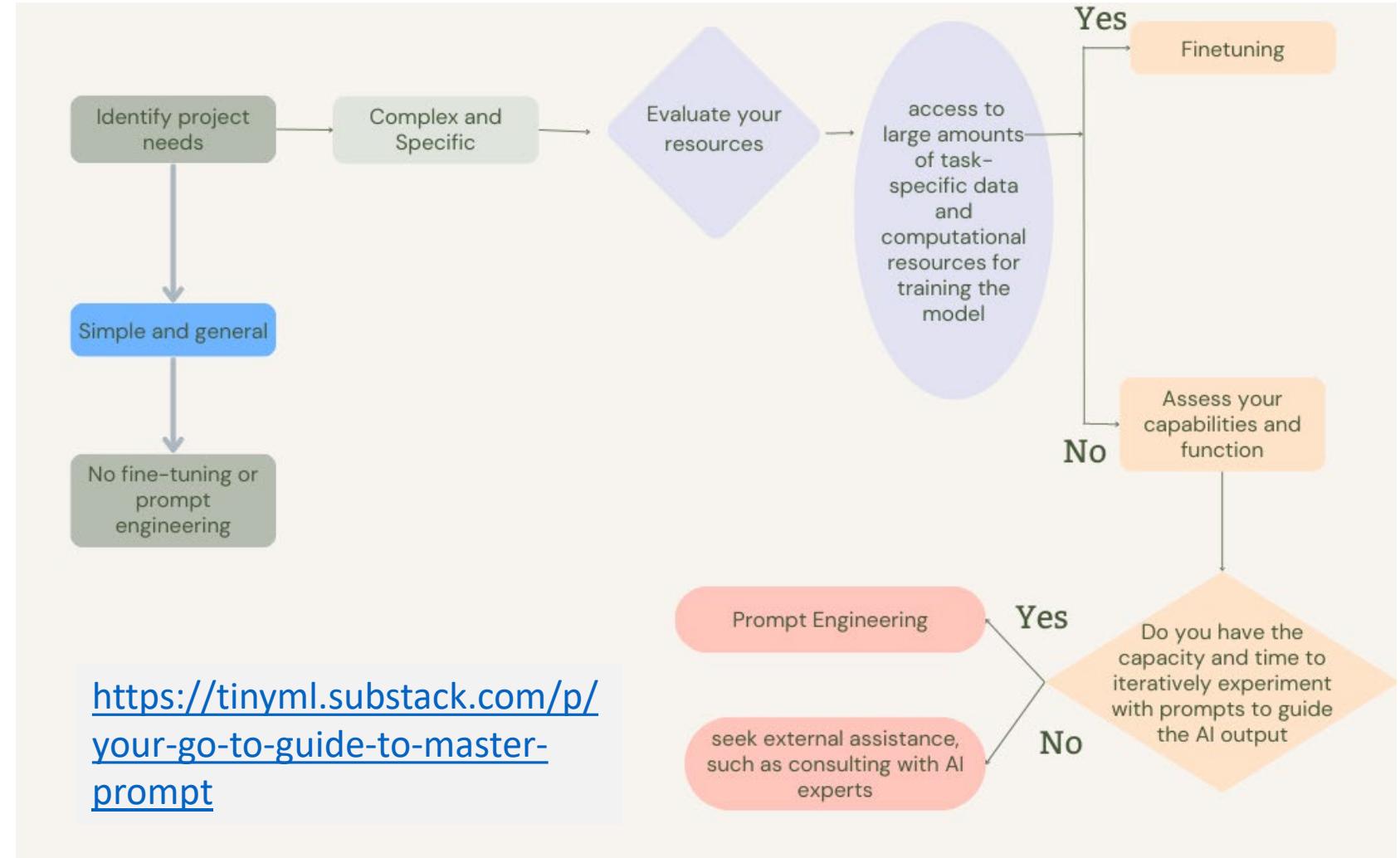
ENGINEERING



T-Few finetuning adds 1D vectors that are multiplied with the K , V , and feed-forward weights during inference. (Source: [Liu et. al, 2022](#))

<https://txt.cohere.com/tfew-finetuning/#:~:text=At%20Cohere%2C%20we%20implement%20the,training%20time%20and%20computational%20resources.>

Prompt Engineering vs Fine-tuning



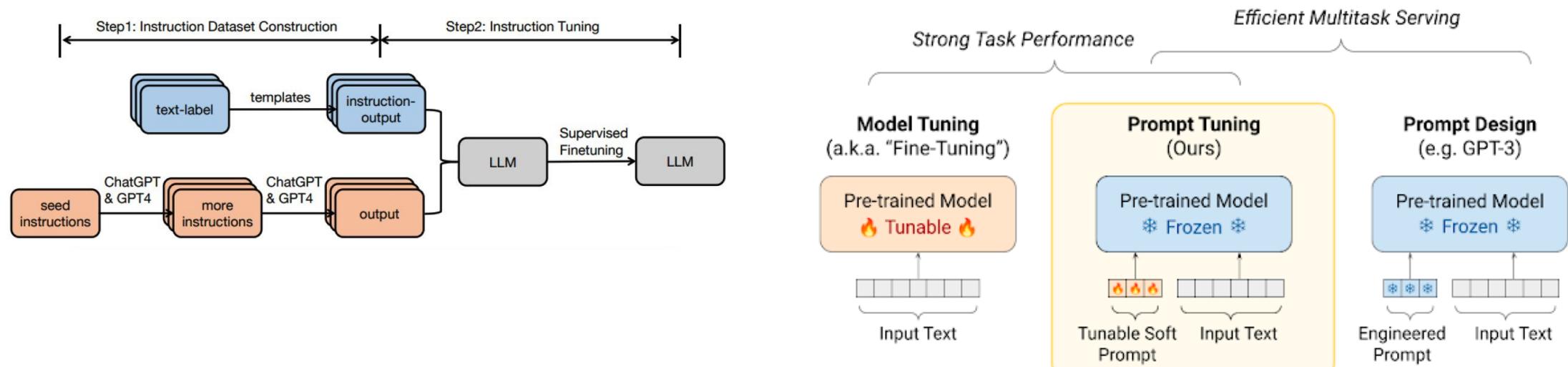
Prompt Engineering vs Fine-tuning

Instruction Tuning for Large Language Models: A Survey

Shengyu Zhang[♦], Linfeng Dong[♦], Xiaoya Li[♦], Sen Zhang[♦]

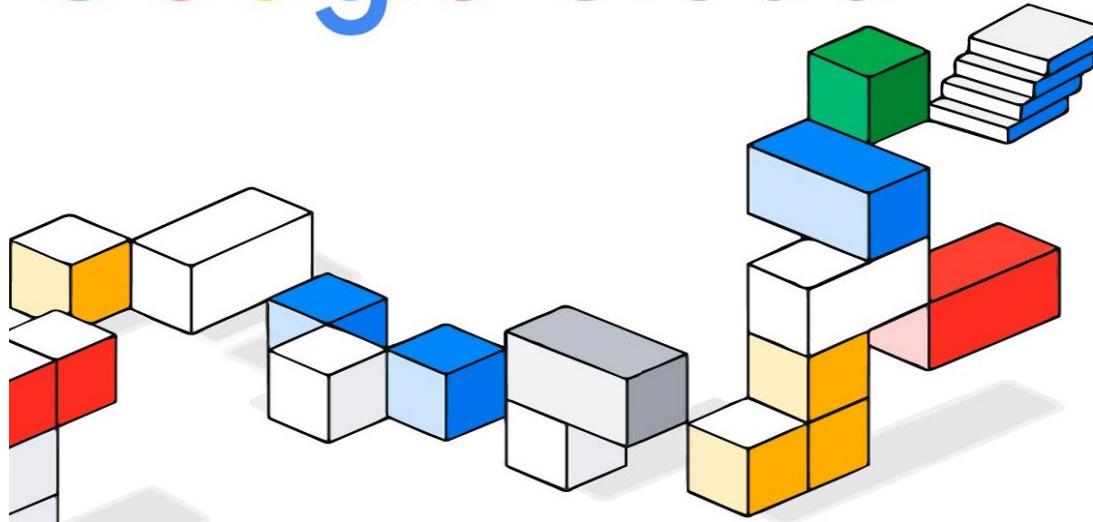
Xiaofei Sun[♦], Shuhe Wang[♦], Jiwei Li^{♦♦}, Runyi Hu[♦]

Tianwei Zhang[▲], Fei Wu[♦] and Guoyin Wang[♦]



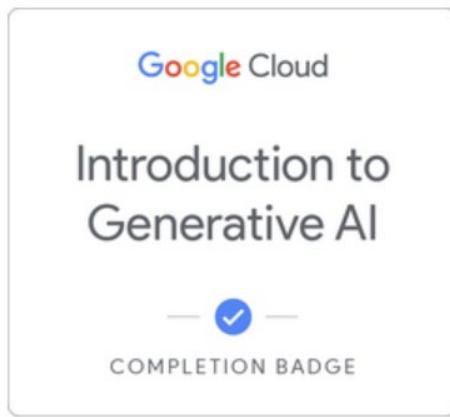
“Online Courses & Open Badges”

Google Cloud



New generative AI training content

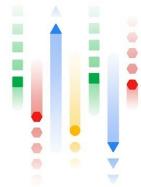
Congratulations! You finished this course.



https://www.cloudskillsboost.google/course_templates/556

<https://www.freecodecamp.org/news/intro-to-generative-ai-10-free-courses-by-google/>

https://docs.google.com/spreadsheets/d/1lHvmTj7SEZU_6vMg7E8yDpBctYdIEYNxc5fJtZzlwCc/edit#gid=0



1. [Introduction to Generative AI](#) – designed to be an overview of what generative AI is and how it differs from machine learning methods.
2. [Introduction to Large Language Models](#) – explores what large language models are, where they are used, and how to use prompt tuning. (If you haven't noticed, prompt writing is being touted as a skill of the future right now.)
3. [Introduction to Responsible AI](#) – an ethical course on what responsible AI is, how it's implemented in Google products, and why it's important. This course introduces Google's 7 AI Principles. I didn't know this was a thing, but there's a [whole page](#) devoted to it. Covering topics from social responsibility, to safety, accountability and privacy design principles, I was happy to see that there is large effort being paid to build in solid principles.
4. [Generative AI Fundamentals](#) – A quiz covering topics from the first three courses.
5. [Introduction to Image Generation](#) – An introduction to diffusion models which are a family of models used in image generation. Some pre-existing knowledge of machine learning, deep learning, convolutional neural nets and/or Python programming is suggested.
6. [Encoder-Decoder Architecture](#) – overview of a machine learning architecture for tasks like machine translation, text summarization, and question answering. Python and Tensorflow knowledge is suggested as a prerequisite.
7. [Attention Mechanism](#) – a technique that allows neural networks to focus on specific parts of an input sequence. Some pre-existing knowledge of machine learning, deep learning, natural language processing, and/or Python programming is suggested.
8. [Transformer Models and BERT Model](#) – Bidirectional Encoder Representations from Transformers...this is what BERT stands for in case you didn't know. You'll learn the main components of the Transformer architecture and intermediate machine learning experience as well, and knowledge of Python and TensorFlow are recommended.
9. [Create Image Captioning Models](#) – how to create an image captioning model using deep learning. Deep learning, machine learning, natural language processing, computer vision and Python are recommended prerequisites.
10. [Introduction to Generative AI Studio](#) – you'll walk through demos of the Generative AI Studio which helps prototype and customize generative AI models. There is a hands-on lab at the end.

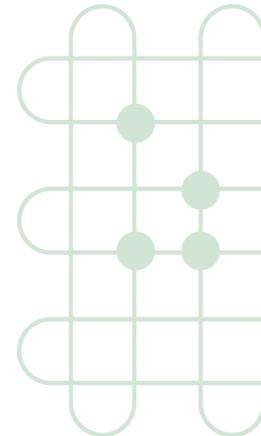
Training a Large Language Model

How does an LLM “learn”?

The technical process of “learning” for LLMs begins with training the model to identify relationships and patterns among words in a large dataset. Through this process, a generative AI model will learn “parameters,” which represent the mathematical relationships in data. Once the model has learned these parameters, it can then use them to generate new outputs based on these parameters.



For example, the [PaLM Research Model \(2022\)](#) learned 540 billion parameters from training on text, which gave it strong capabilities in natural language inference, question and answering, and translation, among other skills. Research is showing that, with the proper training, models with fewer parameters (20B - 200B) can be similarly capable.



What is pre-training? What is fine-tuning?

LLMs are developed in multiple stages, including pre-training and fine-tuning. Pre-training is a way of training an ML model on a variety of data. This gives the model a head start when it is later trained on a smaller dataset of labeled data for a specific task.

Following pre-training, more data can be added to an existing LLM through a process called fine-tuning. Fine-tuning an LLM is the process of adapting a pre-trained LLM to improve its performance on a specific task. The model learns from additional example data to help hone its capabilities.



For example, fine-tuning a general purpose language model can teach it to summarize technical reports in general by using just a few examples of technical reports and accurate summaries.

What is grounding?

Grounding a model refers to the process of linking the abstract concepts in the model to real world entities. Developers use a variety of techniques for grounding generative AI models, including training with real-world data, simulating interactive environments, or even using equipment that can provide actual sensory input. Grounding an LLM can help equip it to better understand language and other abstract concepts in the context of the real world, which may be helpful for tasks such as natural language processing or improving the factuality of model responses.



“Conclusie”

Edubadges: Gen-AI LeerDoel [short term]

Moet gericht zijn op het ervaring op doen met

“prompt engineering”

“fine-tuning”

“veilig-werken / Cyber security (GDPR / DPIA)”

van reeds bestaande Grote Taalmodellen

Professionals die dit beheersen en kunnen toepassen binnen het domein waar ze werkzaam zijn, zullen van grote waarde zijn binnen de organisaties waar ze werkzaam zijn.

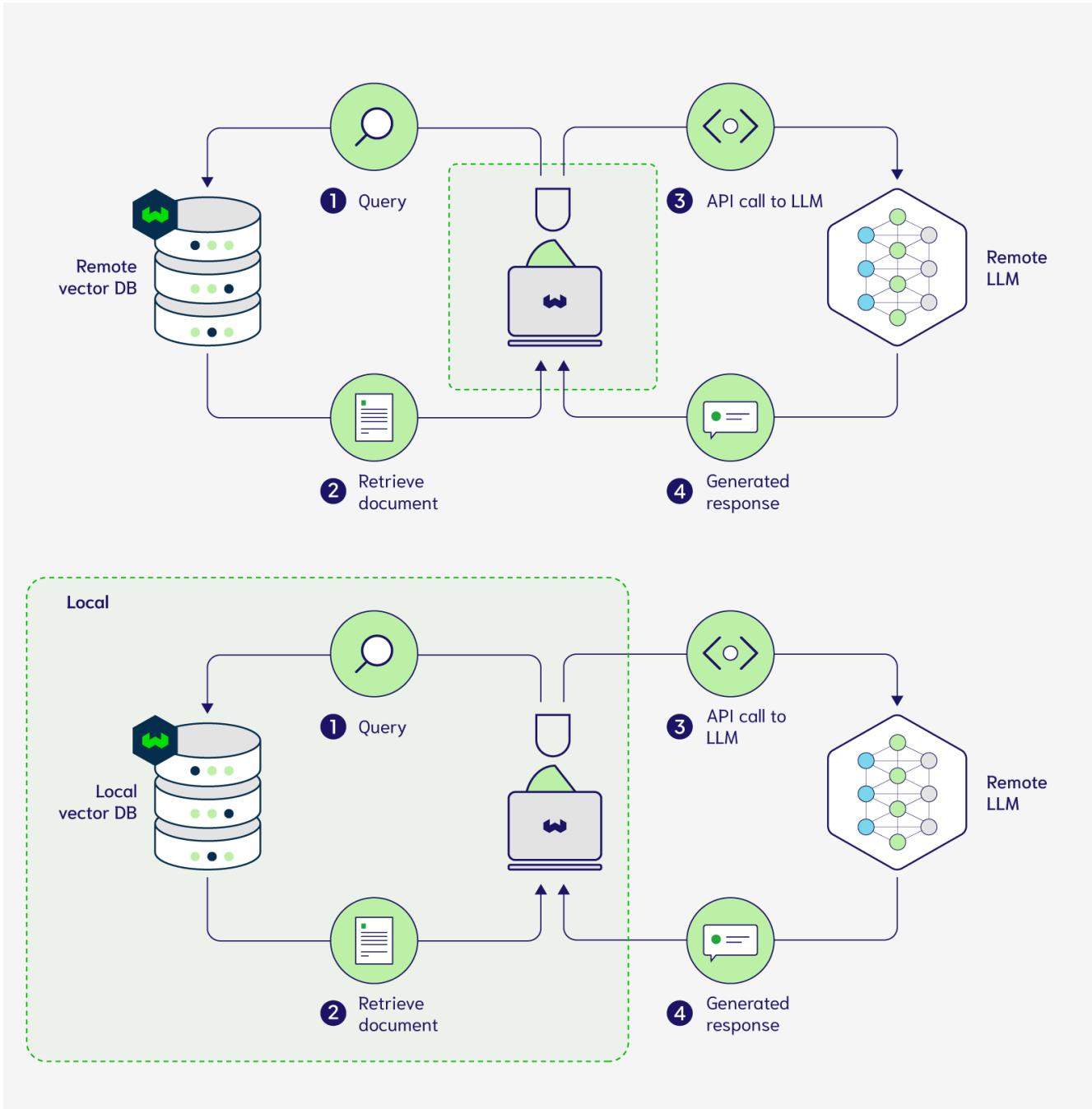
Edubadges: Gen-AI LeerDoel [Long term]

Moet gericht zijn op

“Ontwikkelen van custom-made Private Grote Taalmodellen ”
“lokaal kunnen benutten van publieke Grote Taalmodellen
voor private doeleinden”

van reeds bestaande Grote Taalmodellen

Professionals die dit beheersen en kunnen toepassen binnen het domein waar ze werkzaam zijn, zullen van grote waarde zijn binnen de organisaties waar ze werkzaam zijn.



<https://weaviate.io/blog/private-lm>



HOGESCHOOL ROTTERDAM

overtref jezelf

Who AM I ?

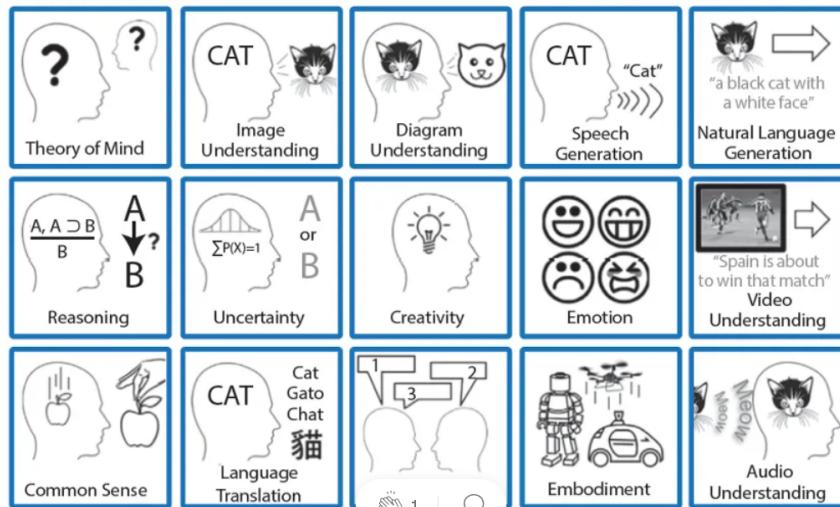


Serengeti Plains. By [Kristin Moger](#)

The World-Wide-Web AI-Safari

Artificial Intelligence: a Human Centred View

AI aims to **mimic & automatise** tasks which otherwise require human perception, cognition and/or motor skills – e.g. pattern recognition, learning, logical reasoning & planning, decision making, problem solving, designing, creativity, likelihood estimation, language acquisition, multi-sensory interfacing, actuated body movement control, locomotion & manipulation, sentiment analysis, and generalisation (see refs [1]...[3]).



RobFvdW

4 Followers

My research anc
biological & psyc
to smart informa

Follow



Who AM I ?



Hogeschool Rotterdam | HR · Instituut voor Communicatie, Media en Informatietechnologie (CMI)
PhD

About

Publications (48)

Network

Projects (8)

Contact

About

48

Publications

83,070

Reads ⓘ

685

Citations

Introduction

Binnen het Creating-010 onderzoeksthema Design in the 21st Century onderzoeken we de valorisatie van Machinaal Lerende (ML) Neurale-Netwerk (NN) algoritmen. De uitkomsten ervan moeten het ontwerp van Human-Centred (HC) data-producten mogelijk maken binnen toepassingsdomeinen zoals Smart & Social City, Zorg-Tech, Smart Logistics & Maritime Innovation. <https://robfvdw.medium.com>

Skills and Expertise

- (Brain)
- (Animal Behavior)
- (Neural Networks)
- (Ethology)
- (Biophysics)
- (Data Science)
- (Data Visualization)
- (Behavioral Analysis)
- (Behavioral Neuroscience)
- (Machine Learning)

Publications

Publications (48)

Designing Neural Networks Through Sensory Ecology "Biology to the rescue of AI"

Produced by Living-Lab: AiRA, Hub voor Data & Responsible AI, Hogeschool Rotterdam

Lunch-Lezing Creating-010 FEB 2022

atietechnologie (CMI) > Robert Frans van der Willigen

Current institution

Hogeschool Rotterdam

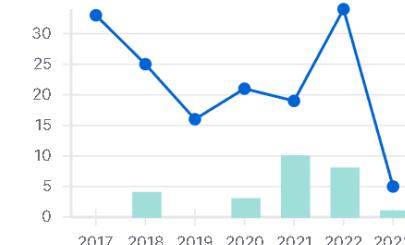


Instituut voor Communicatie, ...

Current position
Data Scientist

Citations since 2017

26 Research Items 153 Citations



[Learn about citations on ResearchGate](#)

Co-authors

Top co-authors



Who AM I ?



Rob

robvdw

Edit profile

8 followers · 0 following

Hogeschool Rotterdam

Netherlands

https://www.researchgate.net/profile/Roberto_Van_Der_Willigen

Highlights

PRO

Organizations



Repositories 15 Projects Packages

Psignifit-MEX Public

Mex files for Matlab 2010b on Win64 / Mac OS X 10.6.7

1 star

1,619 contributions in the last year



Learn how we count contributions

2023

2022

2021

Contribution activity

March 2023

Created 391 commits in 3 repositories

HR-ChatGPT/ChatGPT-UITGELEGD 381 commits

HR-ChatGPT/chatgpt.uitgelegd 8 commits

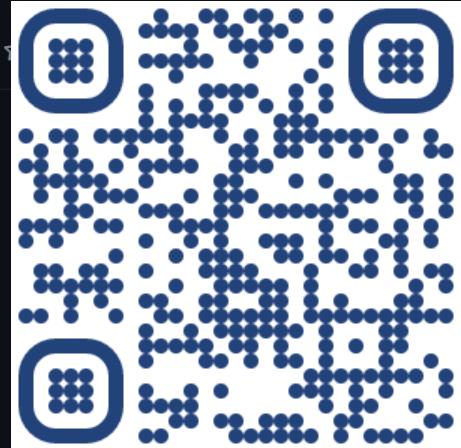
HR-Data-Supported-Healthcare/Corpora-that-Support-Clinical-Reasoning 2 commits

Joined the Hogeschool Rotterdam organization



Hogeschool Rotterdam

De wildgroei aan gezondheidsdossiers heeft een overvloed aan klinisch tekst opgeleverd die vaak onbenut blijft. AI in de vorm van NLP kan uitkomst bieden.

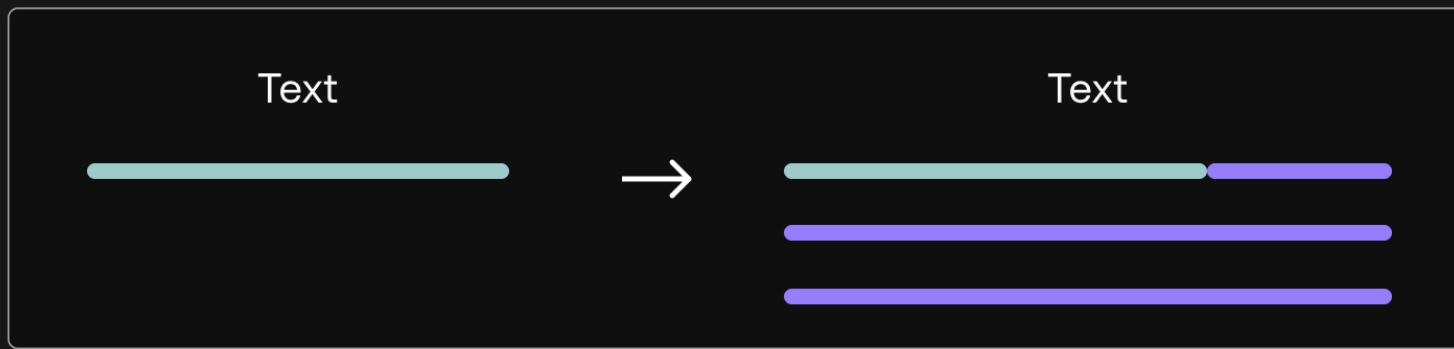


Customize your pins

Contribution settings ▾



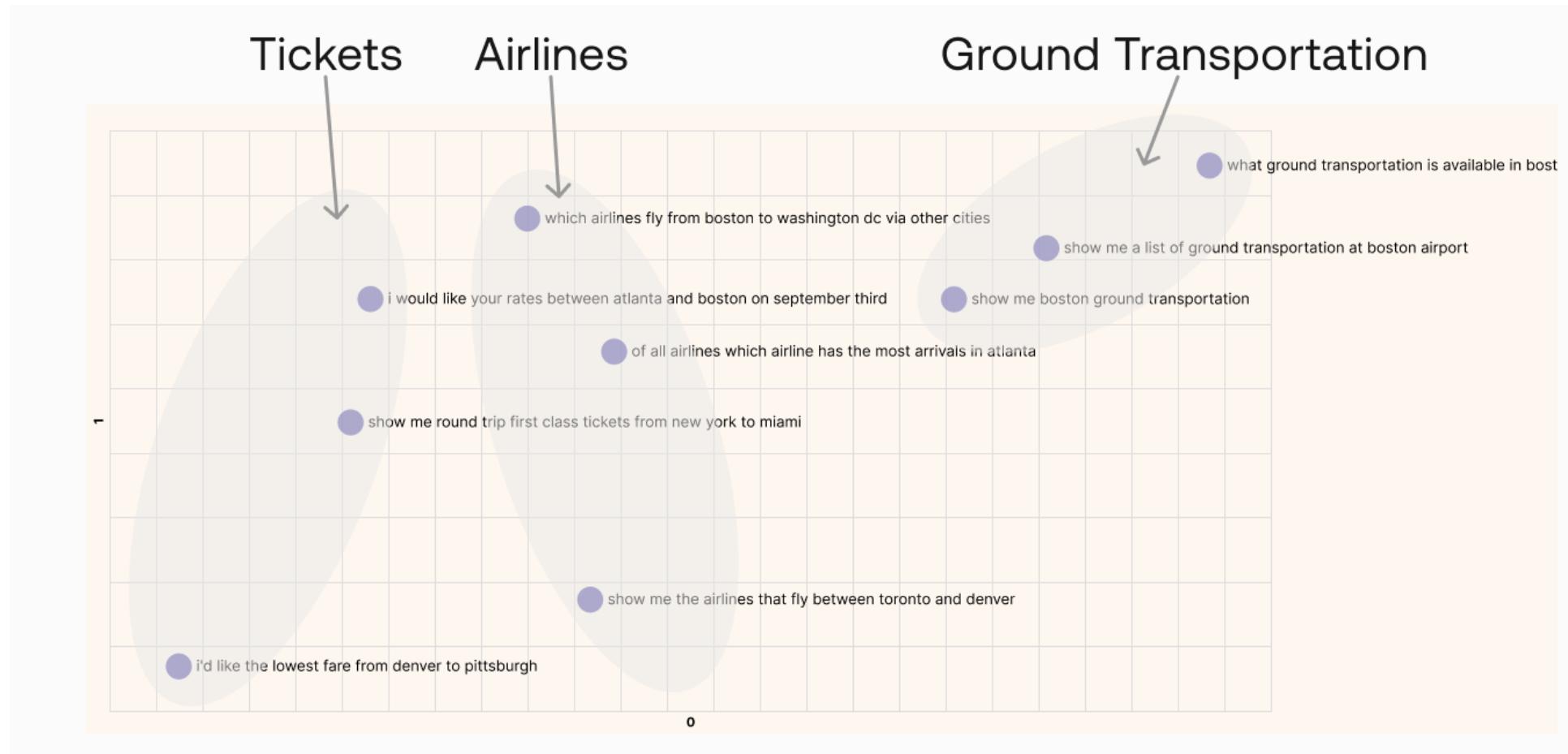
Text Generation



Text Representation



<https://docs.cohere.com/docs/embeddings-visualizing-data>





HOGESCHOOL
ROTTERDAM