

Digital Humans

AI-POWERED
DIGITAL HUMANS



Proof-of-Concept [PoC]

Talige Prototyping
&
Itereren
met Azure Open service

https://www.reply.com/contents/AI-powered_Digital_Humans - English.pdf

Les 04





European Parliament

Artificial intelligence: How does it work, why does it matter, and what can we do about it?

Table of contents

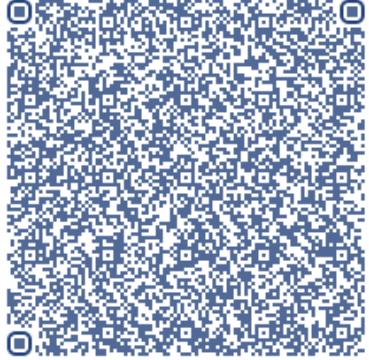
Executive summary.....	III
Glossary	VI
1. What is artificial intelligence?	1
2. How does it work?	2
2.1 First wave: symbolic artificial intelligence.....	2
2.2 Second wave: machine learning and data-driven artificial intelligence	3
2.3 Speculative future waves: towards artificial superintelligence?	13
3. Why does it matter?.....	18
3.1 Current opportunities and challenges	18
3.2 Speculative opportunities and challenges.....	29
4. What can we do about it?.....	32
4.1 Regulatory and economic options.....	32
4.2 Technology development and application options.....	42
4.3 Societal and ethics options.....	49
5. Conclusions.....	60
Key references	61

[https://www.europarl.europa.eu/RegData/etudes/STUD/2020/641547/EPRS_STU\(2020\)641547_EN.pdf](https://www.europarl.europa.eu/RegData/etudes/STUD/2020/641547/EPRS_STU(2020)641547_EN.pdf)

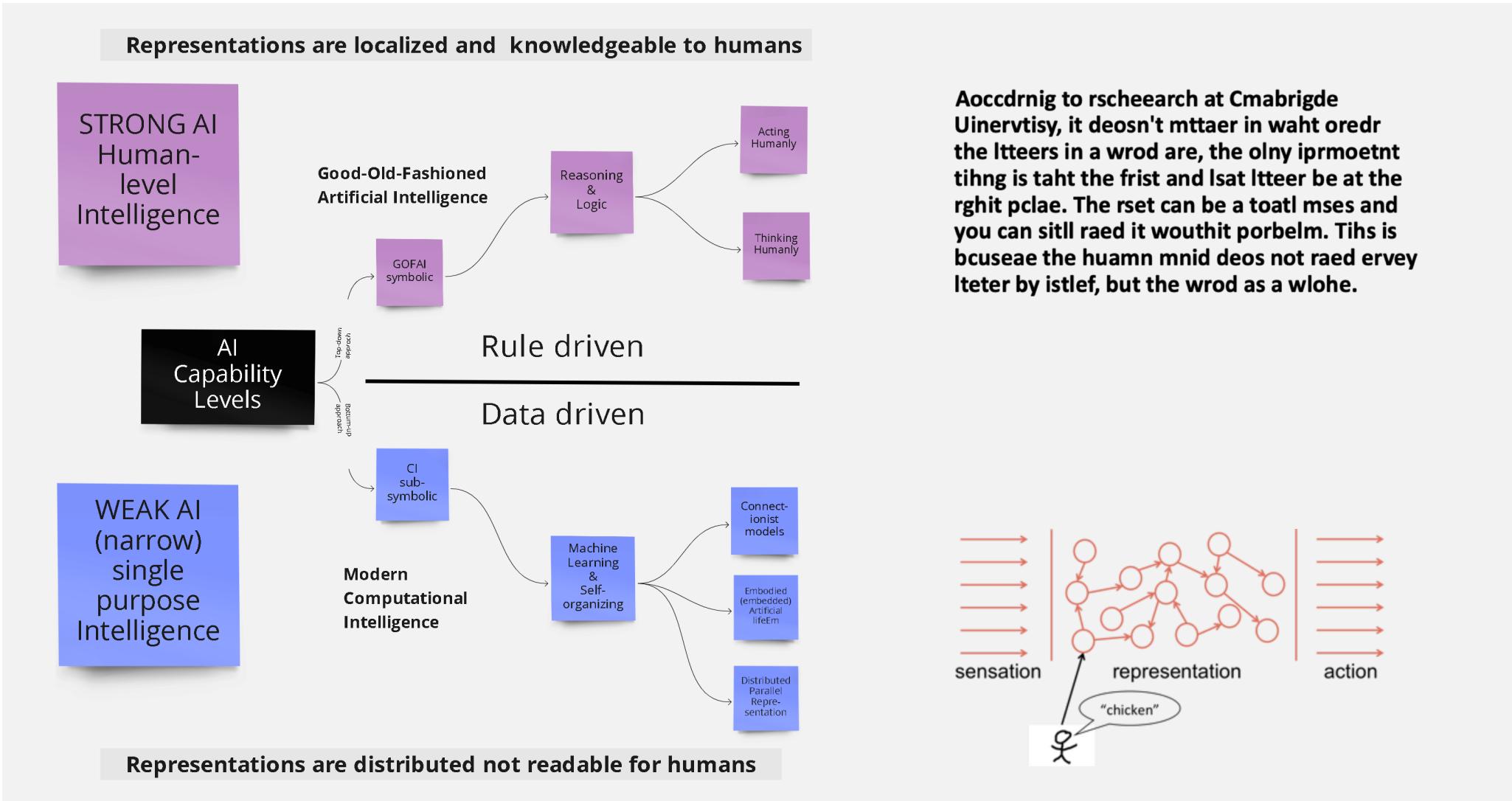


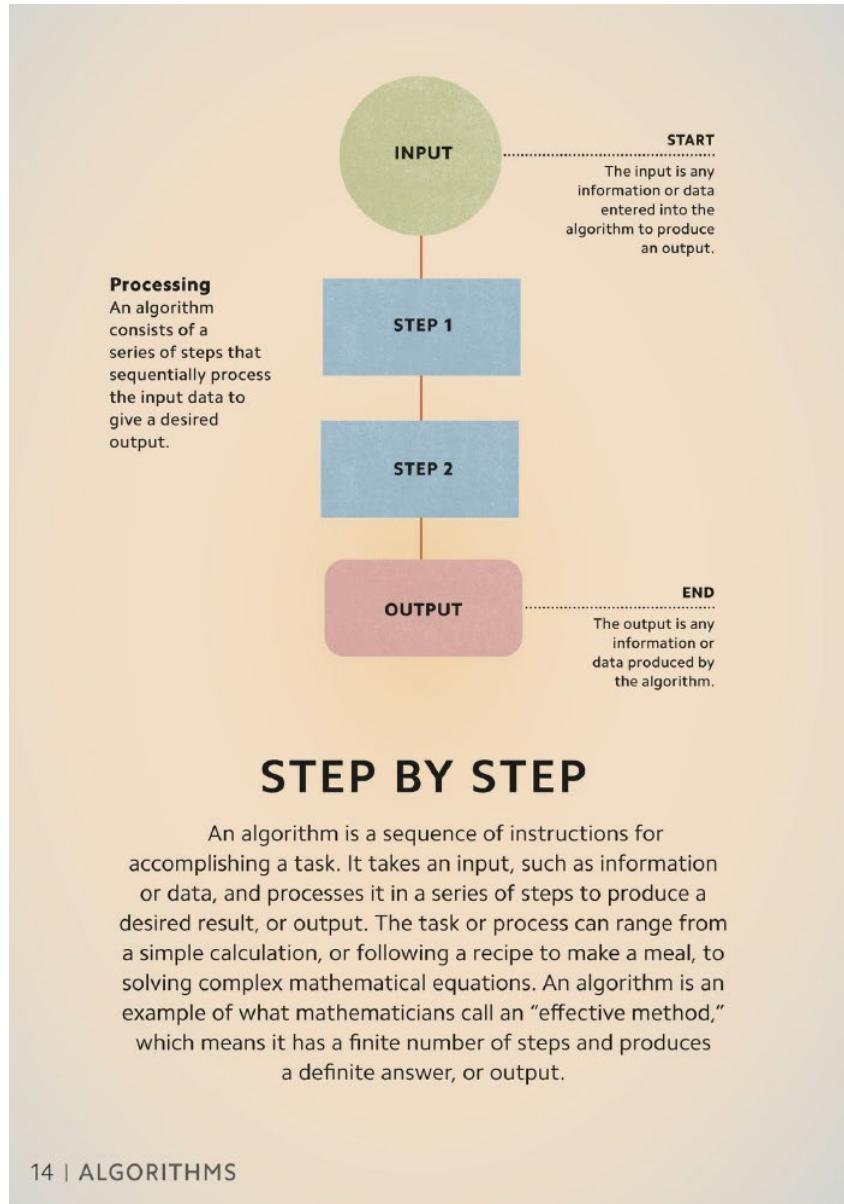
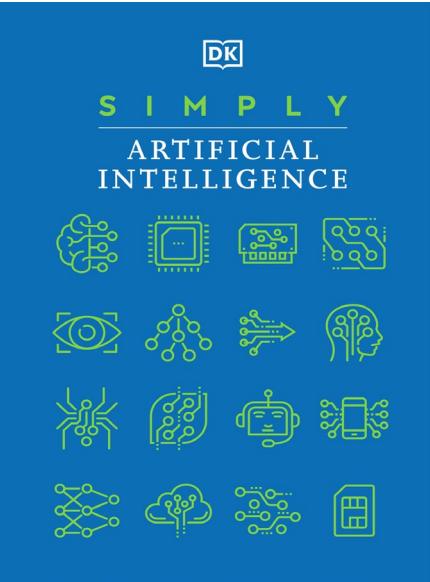
HOGESCHOOL
ROTTERDAM

AI-taxonomie is complex



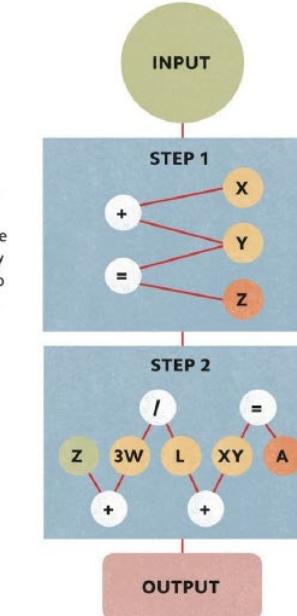
https://www.researchgate.net/publication/359424818_Designing_Neural_Networks_Through_Sensory_Ecology_Biology_to_the_rescue_of_AI_Produced_by_Living-Lab_AiRA_Hub_voor_Data_Responsible_AI_Hogeschool_Rotterdam_Lunch-Lezing_Creating-010_FEB_2022





STEP BY STEP

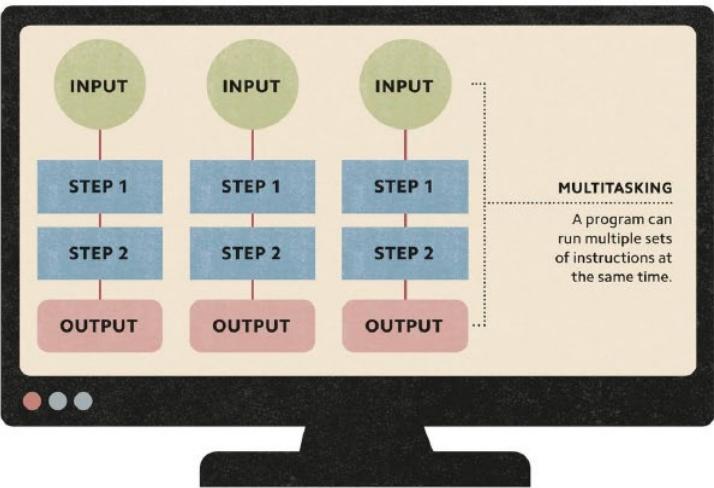
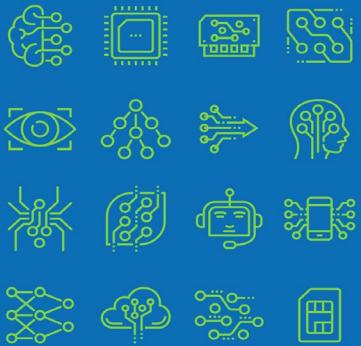
An algorithm is a sequence of instructions for accomplishing a task. It takes an input, such as information or data, and processes it in a series of steps to produce a desired result, or output. The task or process can range from a simple calculation, or following a recipe to make a meal, to solving 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.



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

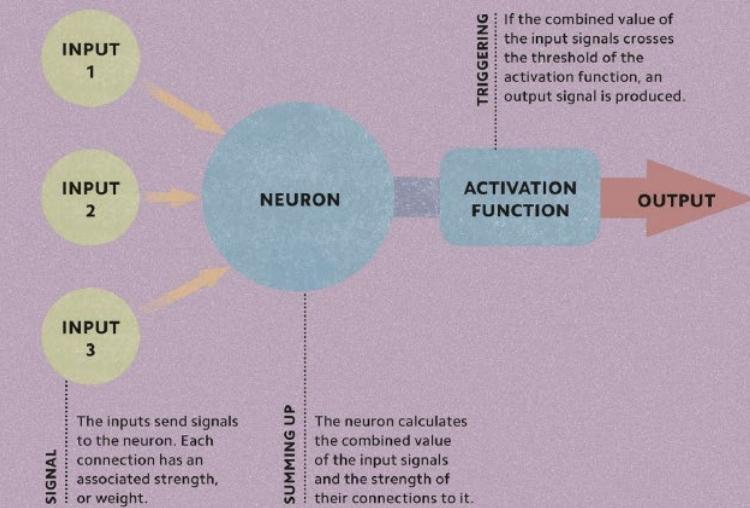
ARTIFICIAL
INTELLIGENCE

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".



Definities / 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.

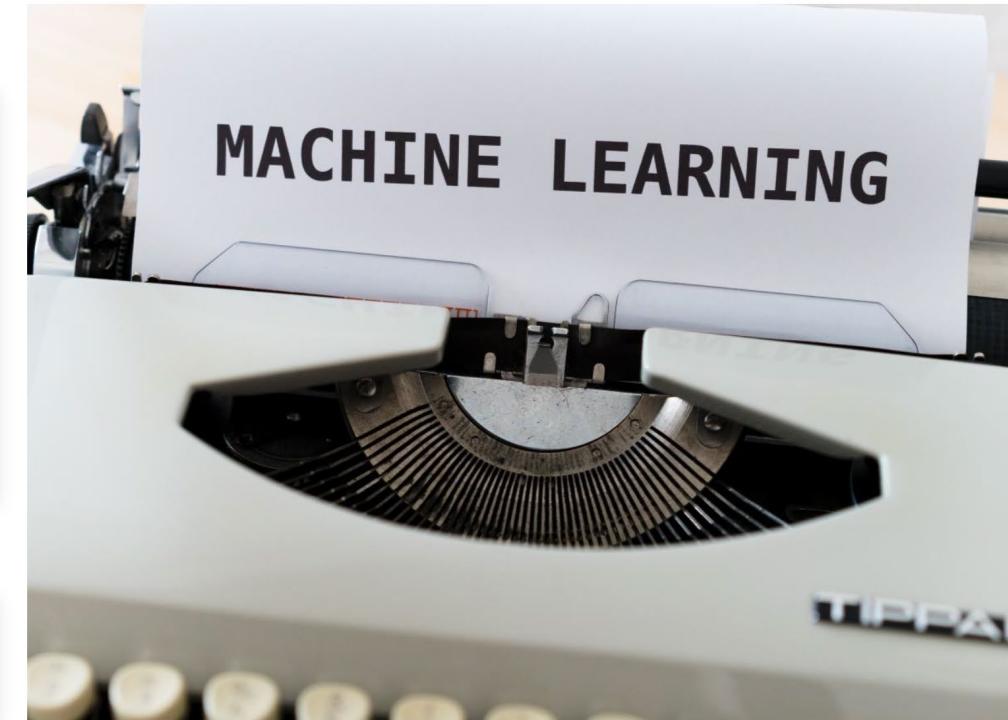
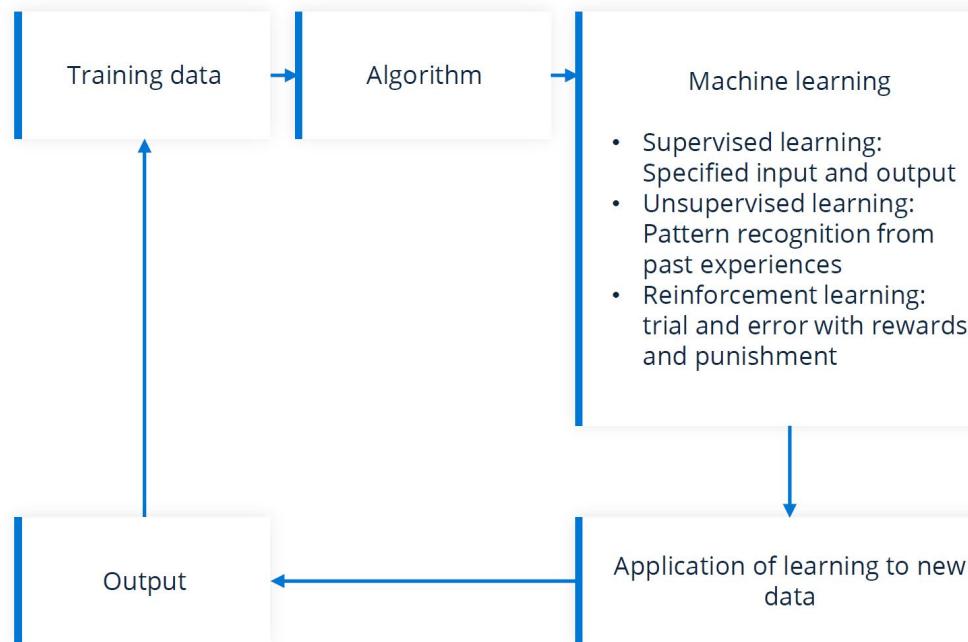
These models learn through observation and pattern matching, also known as training.

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.

Generative AI-Powered life-like Digital Humans

Deep Voodoo

DGene

ElevenLabs

MARZ

Papercup

Respeecher

Runaway

Reply

Strada

Wonder Dynamics

Ready Player Me

Avatars.me

Vocaloids

UneeQ

≡ ⌂ INSIDER Newsletters Log In ⌂

US MARKETS OPEN Dow Jones +0.05% Nasdaq -0.26% S&P 500 -0.05% META +0.4% TSLA -2% BABA -1.92%

Premium HOME > MEDIA

12 AI startups that are raising hundreds of millions in funding to disrupt Hollywood

Lucia Moses Updated Sep 15, 2023, 10:34 PM CEST



Michelle Yeoh in "Everything Everywhere All At Once." A24

<https://www.businessinsider.com/ai-startups-netflix-marvel-lucasfilm-hollywood-deepfake-dubbing-vc-funding-2023-3?r=US&IR=T>

Animated “digital Humans”

*Hoe bouw
en test je
zulke AI-technologie?*

*Taxonomie implementatie
technologie niveau's voor
“digital Humans” bevat 3
lagen: *Talig / Fysiek / Digitaal**

Talig : Conversationele Agenten

Fysiek: Robotica / Cybernetica / “Reinforced Learning”

Digitaal: AI-gedreven Video “Dubbing” & “Motion Capture”

Game Engines [+Omniverse Connectors]

<https://www.fxguide.com/quicktakes/nvidia-ace-enables-easier-interactive-avatars/>

<https://blogs.nvidia.com/blog/2023/04/21/epic-benefits-omniverse-connector-unreal-engine/>

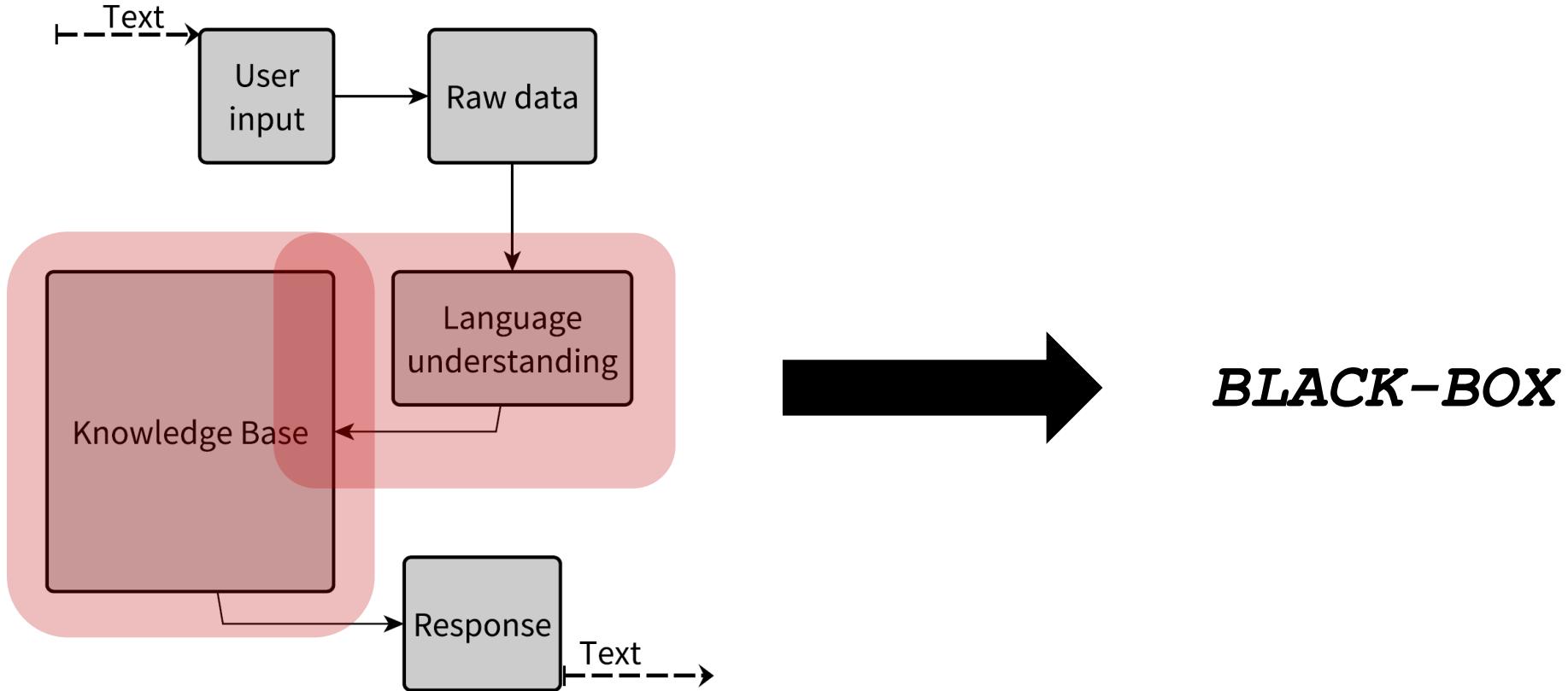


HOGESCHOOL
ROTTERDAM

TALIGE IMPLEMENTATIE

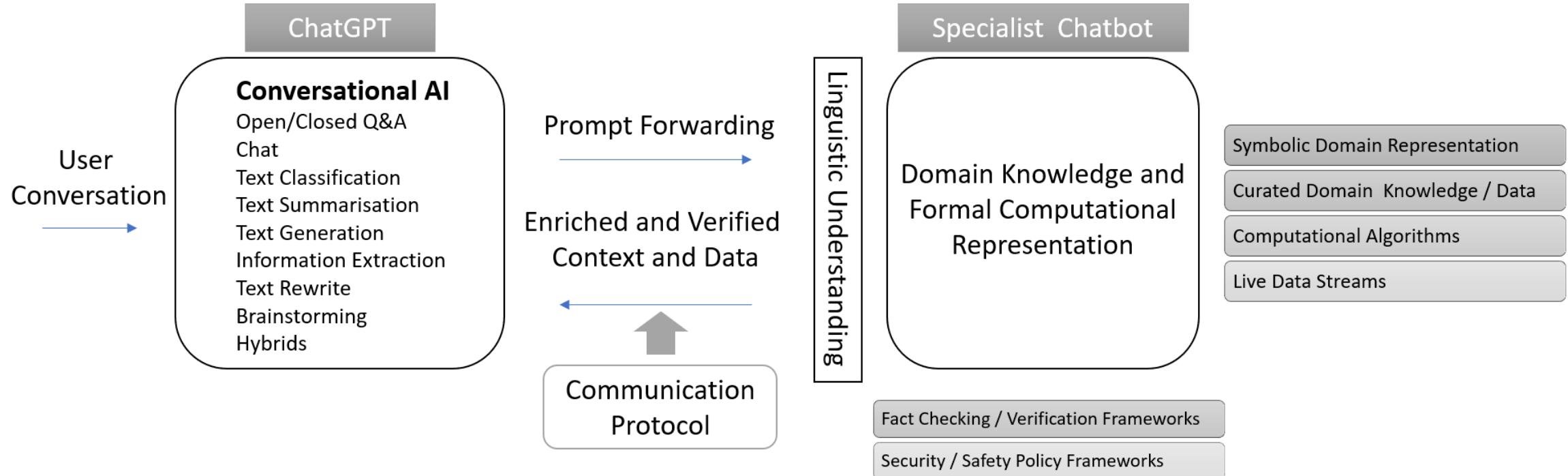
Digital human

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

Conversational Agents



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

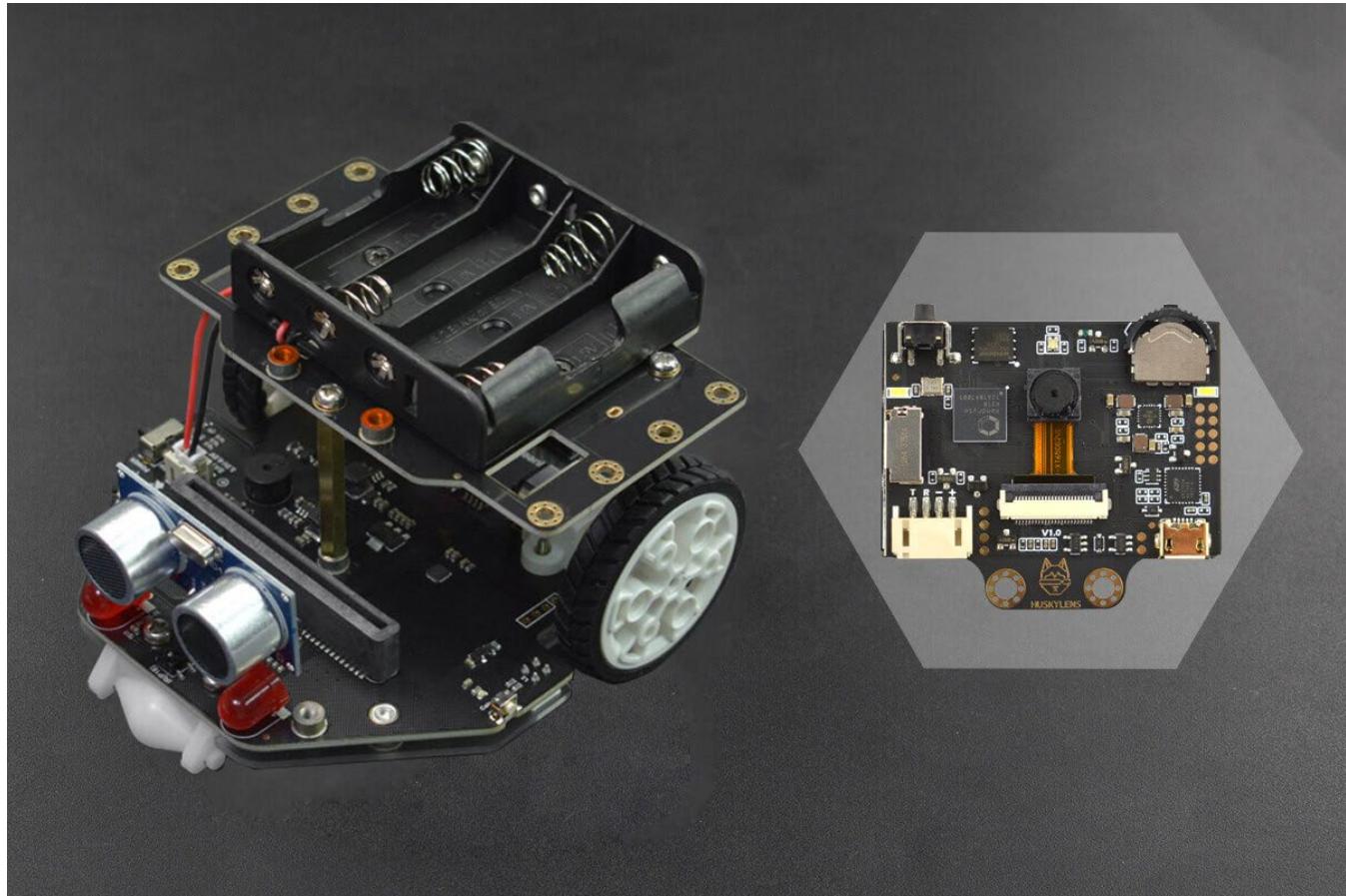
Daswin De Silva ; Nishan Mills ; Mona El-Ayoubi ; Milos Manic ; Damminda Alahakoon All Authors

635
Full
Text Views



ROBOT IMPLEMENTATIE

Digital human



[https://wiki.dfrobot.com/SKU_MBT0021-
EN Maqueen Plus STEAM Programming Educational Robot](https://wiki.dfrobot.com/SKU_MBT0021-EN_Maqueen_Plus_STEAM_Programming_Educational_Robot)

DIY Line Tracking Robot with HuskyLens and Romeo



HuskyLens Team

Jul 07.2020

7

10

61888

Easy

Project Introduction

In this project, we will demonstrate the line tracking function of HuskyLens. We will install HuskyLens to a Devastator tank mobile robot, then HuskyLens will control the robot to perform line tracking.

Press the learning button of HuskyLens, then it can learn and remember the features of an object line. There is no need to adjust parameters again and again, it is easier to use and more powerful than other ordinary sensors.

HARDWARE LIST

1 x Gravity: HUSKYLENS - An Easy-to-use AI Machine Vision Sensor

Link

1 x Devastator Tank Mobile Robot Platform

Link

1 x Romeo V2- an Arduino Robot Board (Arduino Leonardo) with Motor Driver

Link

1 x 7.4V Lithium Battery

Other Romeo board and tank robot also can be used, such as follows:

[Romeo BLE - Arduino Robot Control Board with Bluetooth 4.0](#)

[Devastator Tank Mobile Robot Platform \(Metal DC Gear Motor\)](#)

<https://community.dfrobot.com/makelog-308279.html>

pxt-DFRobot_MaqueenPlus_v20 Public Watch 12

master 1 branch 19 tags Go to file Add file Code

tangjie133 Merge pull request #5 from almasy/master ... 17b72f8 on Jun 27 57 commits

.github/makecode 更新 main.blocks 2 years ago

_locales Update Czech translations 4 months ago

.gitignore add 2 years ago

Gemfile add 2 years ago

LICENSE 格式修改 7 months ago

Makefile add 2 years ago

README.md 更新 README.md 7 months ago

_config.yml add 2 years ago

icon.png 格式修改 7 months ago

maqueenPlusV2.ts add cs and fr 5 months ago

pxt.json Update Czech translations 4 months ago

test.ts 更新 README.md, _locales/zh-cn/maqueenPlusV2-strings.json, _locales/... 7 months ago

tsconfig.json add 2 years ago

README.md

Maqueen Plus V2

Introduction

This is the latest version of Maqueen Plus, a programming robot for STEAM education. Optimized with more expansion ports, larger capacity power supply and larger body, the Maqueen Plus V2.0 can be perfectly compatible with more peripheral components like HuskyLens AI camera and Maqueen Mechanic kits, which makes it an accessible STEAM robot teaching tool for primary and secondary students. Besides, it can be not only suitable for classroom teaching, but also can be used for after-school extended exercises and robot competitions. Besides all the functions of Maqueen Lite, it offers richer and more flexible functions and stronger performance. Whether you have ever used Maqueen series products or not, you'll find it very easy to get started.

https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

DIGTALE IMPLEMENTATIE

Digital human

GAME ENGINE



MetaHuman SDK

AR VR Lab

37 37 reviews written | 145 of 148 questions answered

Automated solution for creating realistic face animation from audio file

AUDIO TO
FACIAL ANIMATION
METAHUMAN SDK

Unavailable

Supported Platforms



Supported Engine Versions

5.0 - 5.2

Description

Reviews

Questions

MetaHumanSDK is a set of tools for creation of an immersive interaction with a digital human. Our service creates facial animation from an audio file or text and the plugin includes connectivity modules of a synthesized voice from Google or Azure (text to speech) and offers the option of creating an interactive chat with the connection to Dialog flow (Google) with the possibility of a live dialogue with a digital human.

[VIDEO TUTORIAL](#)
[OFFICIAL TRAILER](#)To get the latest news and feedback and find friends join our [DISCORD COMMUNITY](#).

<https://www.unrealengine.com/marketplace/en-US/item/66b869fa0d3748e78d422e59716597b6>



NVIDIA
OMNIVERSE

CREATING ANIMATED DIGITAL HUMANS FOR OMNIVERSE

John Martin II, Vice President, Reallusion
Elvis Huang, Head of Innovation, Reallusion



Character Creator Connector for Omniverse

NEWS LIBRARY EXCHANGE COLLABORATION LEARN

Search

ALL

APPS

CONNECTORS

3DS MAX CONNECTOR
3DS MAX CONNECTOR DAILY
MAYA CONNECTOR
MAYA CONNECTOR DAILY
REVIT CONNECTOR
CONNECT SAMPLE
UE4 4.24 CONNECTOR
UE4 4.25 CONNECTOR
UE4 4.26 CONNECTOR
ARCHICAD CONNECTOR
RHINO CONNECTOR
CC CONNECTOR
SKETCHUP CONNECTOR

REALLUSION CHARACTER CREATOR | OMNIVERSE

3.41.1-e **INSTALL**



About Omniverse REALLUSION CHARACTER CREATOR

Description

Character Creator Omniverse Connector is a plugin for exporting digital humans and motions from Character Creator 3 to an NVIDIA Omniverse Neucleus Server.

The Character Creator Connector allows users to export character, accessory, motion, and material content to USD and MDL. The file structure is well-organized, which allows users to easily replace files to achieve collaborative works.

Character Creator 3 and Omniverse in combination help a variety of industries such as AEC, M&E, Video Games, to access and deploy characters as task performers, virtual hosts or citizens for simulations and visualizations.

Tags

Character Creation, Character Animation, Game Character, Content Creation, Connector

Learn more

[RELEASE NOTES](#) [DOCUMENTATION](#) [TUTORIALS](#) [FORUMS](#)

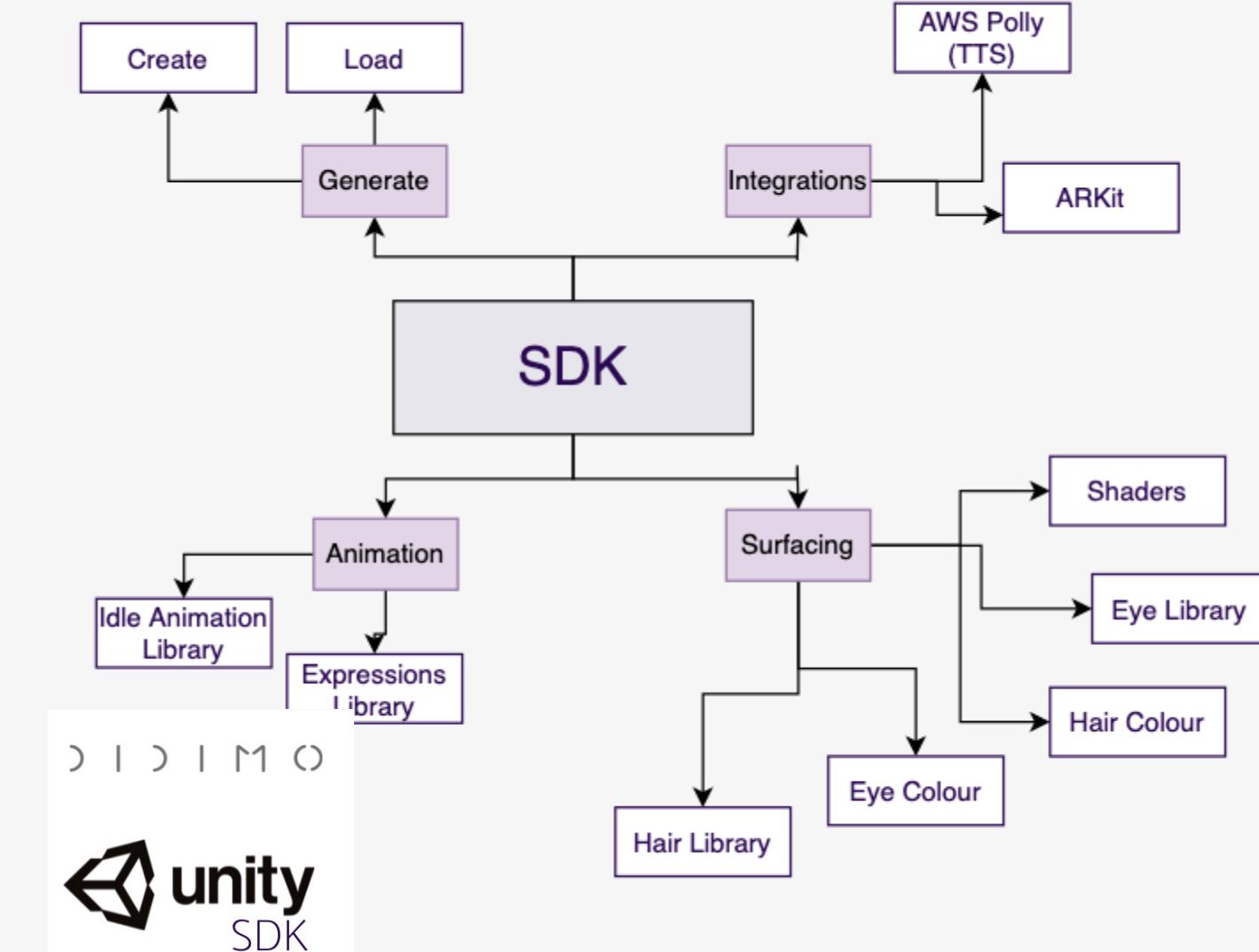
Developer
Reallusion

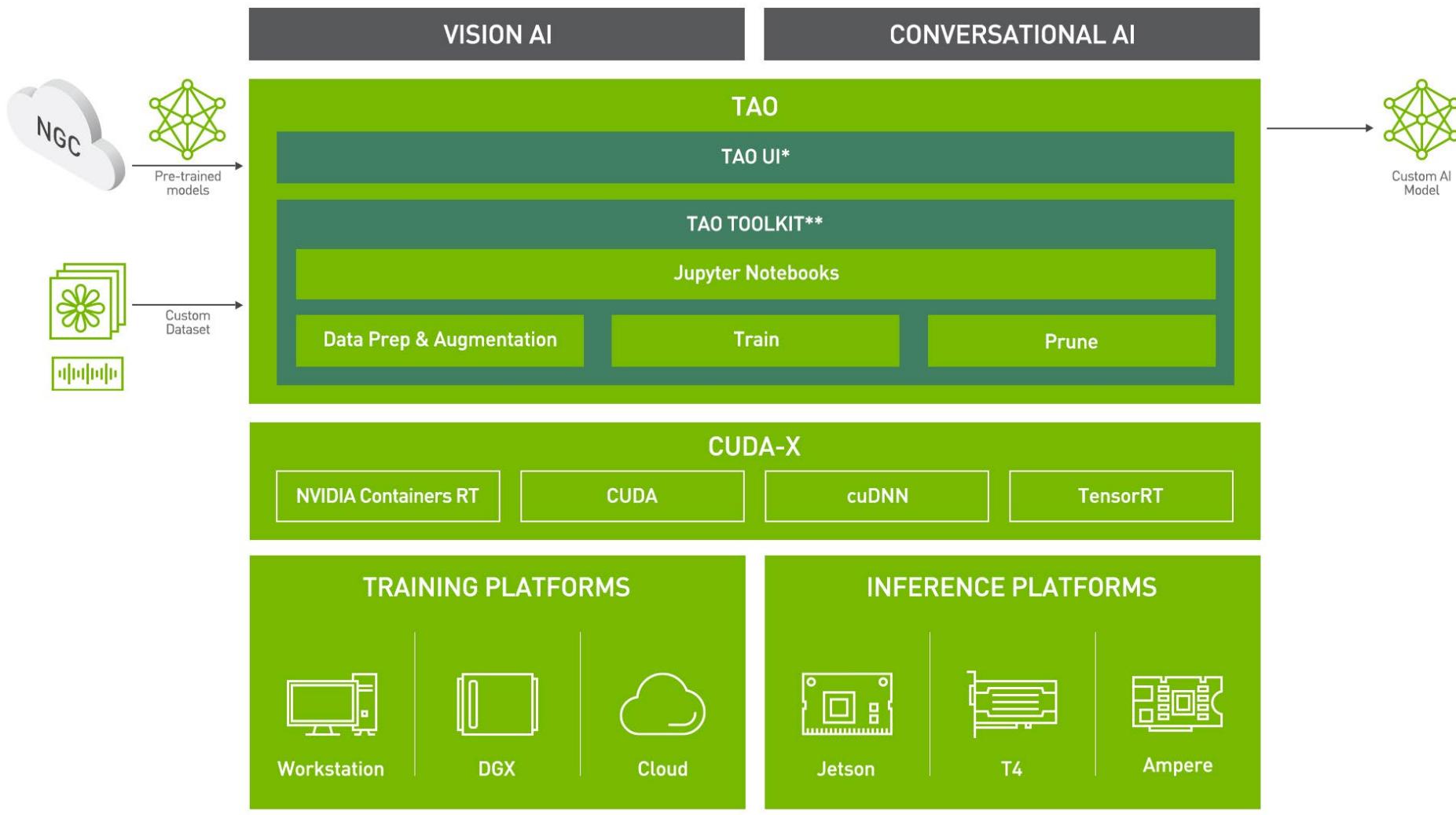
Version
3.41.1-e

Platforms

<https://www.nvidia.com/en-us/on-demand/session/gtcspring21-s31882/>

*Coding & SDKs zijn een
vereiste voor alle
implementatie niveau's*

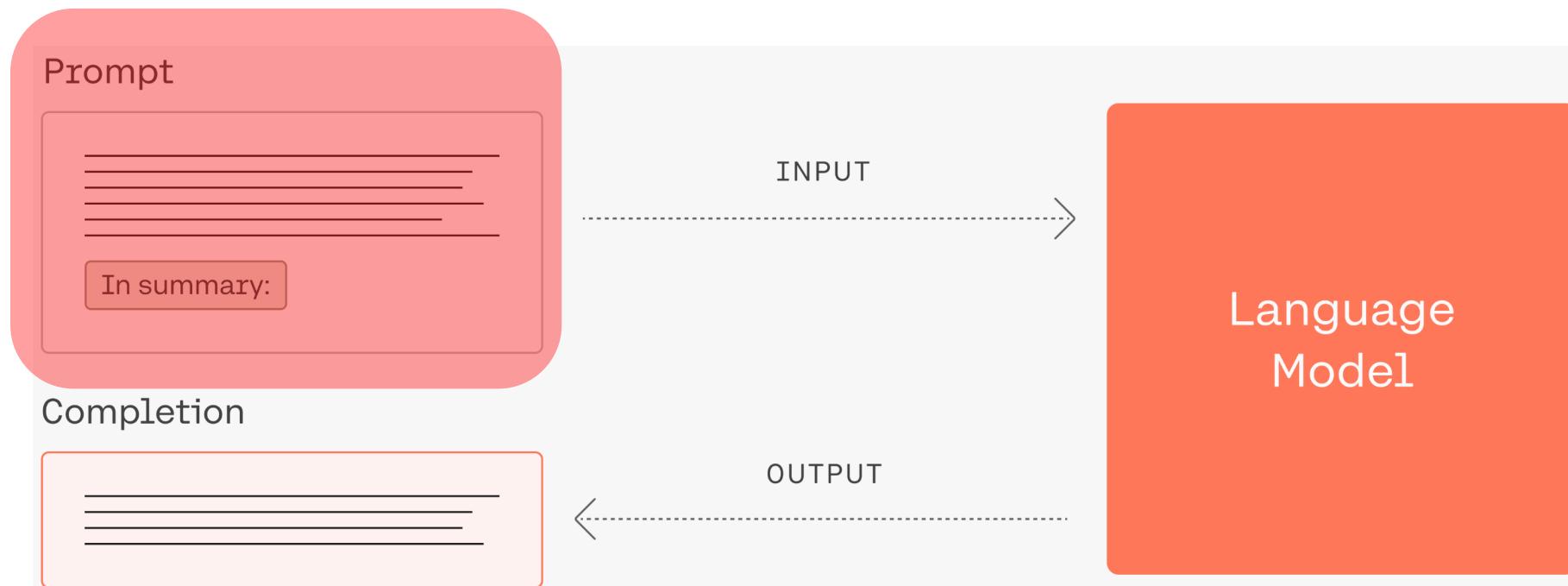




<https://developer.nvidia.com/blog/fast-track-your-production-ai-with-pre-trained-models-and-tao-toolkit-3-0/>

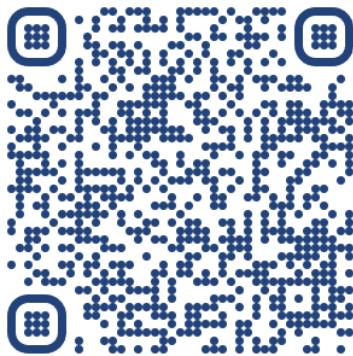
*How to create a
Proof of Concept [PoC]
for Human Dialogues
with large language Models*

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



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

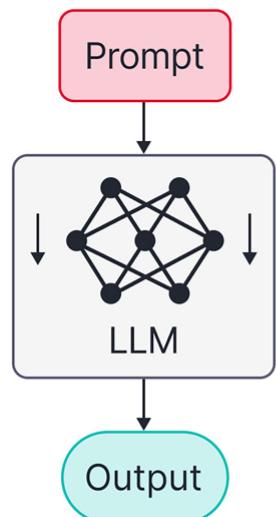
<https://docs.cohere.com/docs/introduction-to-large-language-models>



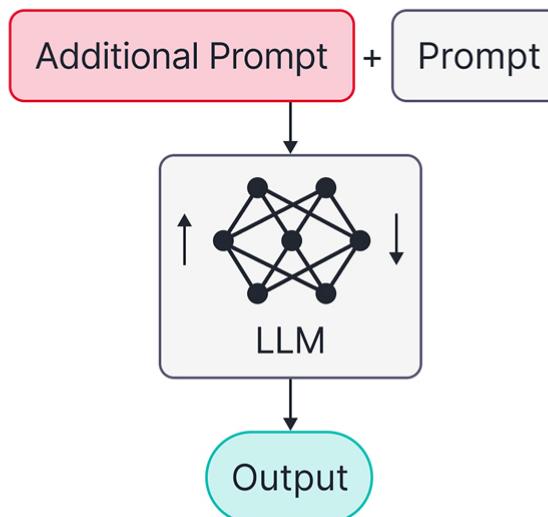
Vier manieren om LLM's te benutten



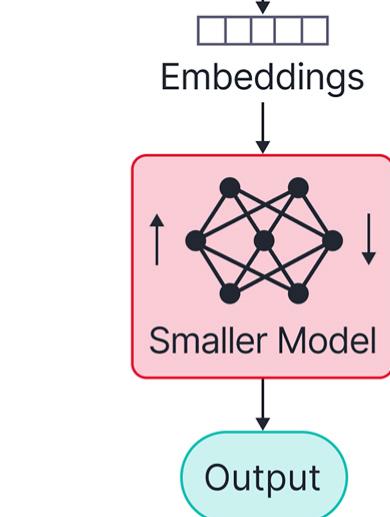
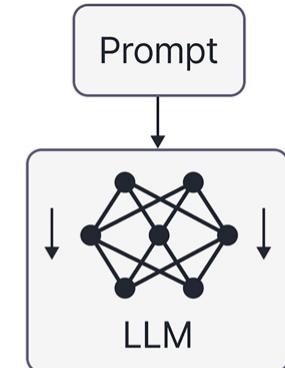
HOGESCHOOL
ROTTERDAM



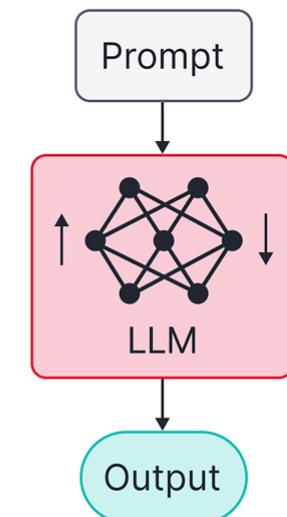
(a) Prompt Engineering



(b) Prompt Tuning

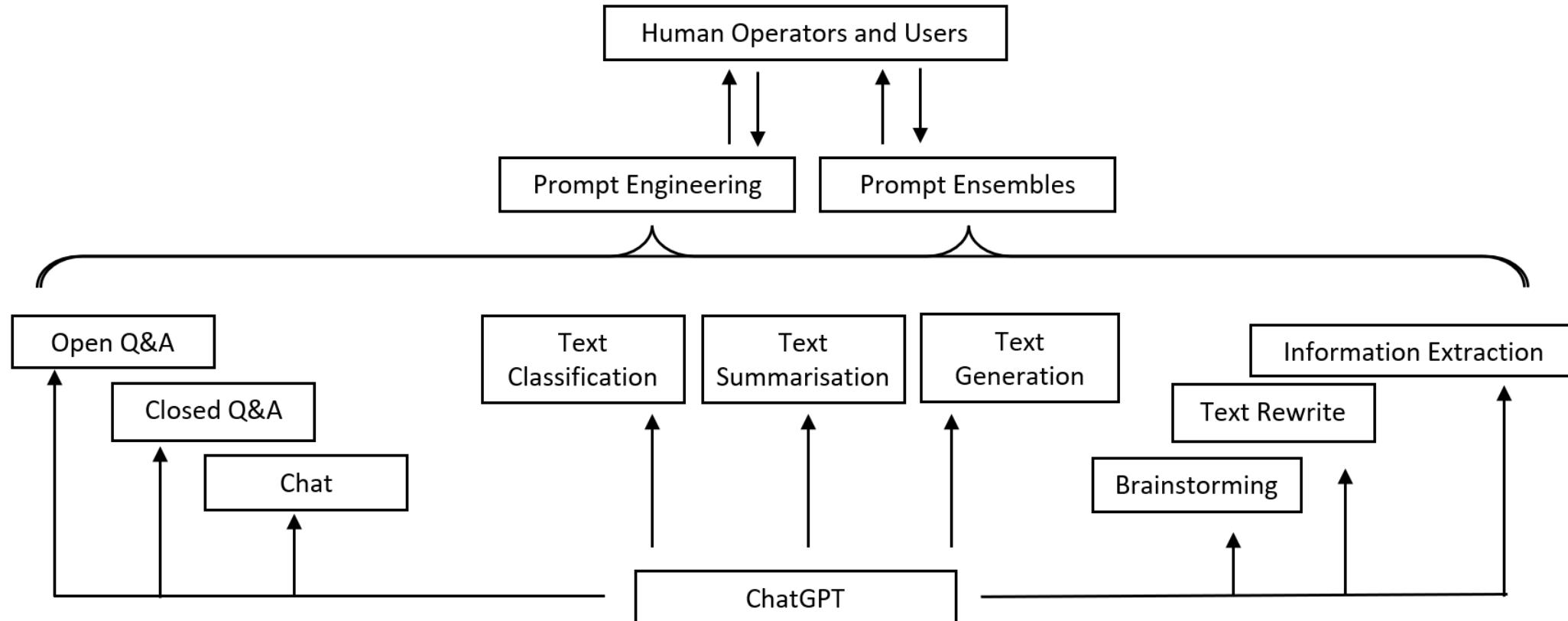


(c) LLM Embeddings as input



(d) Model Fine Tuning

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





<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 or 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/promptslab/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-ENGINEERING

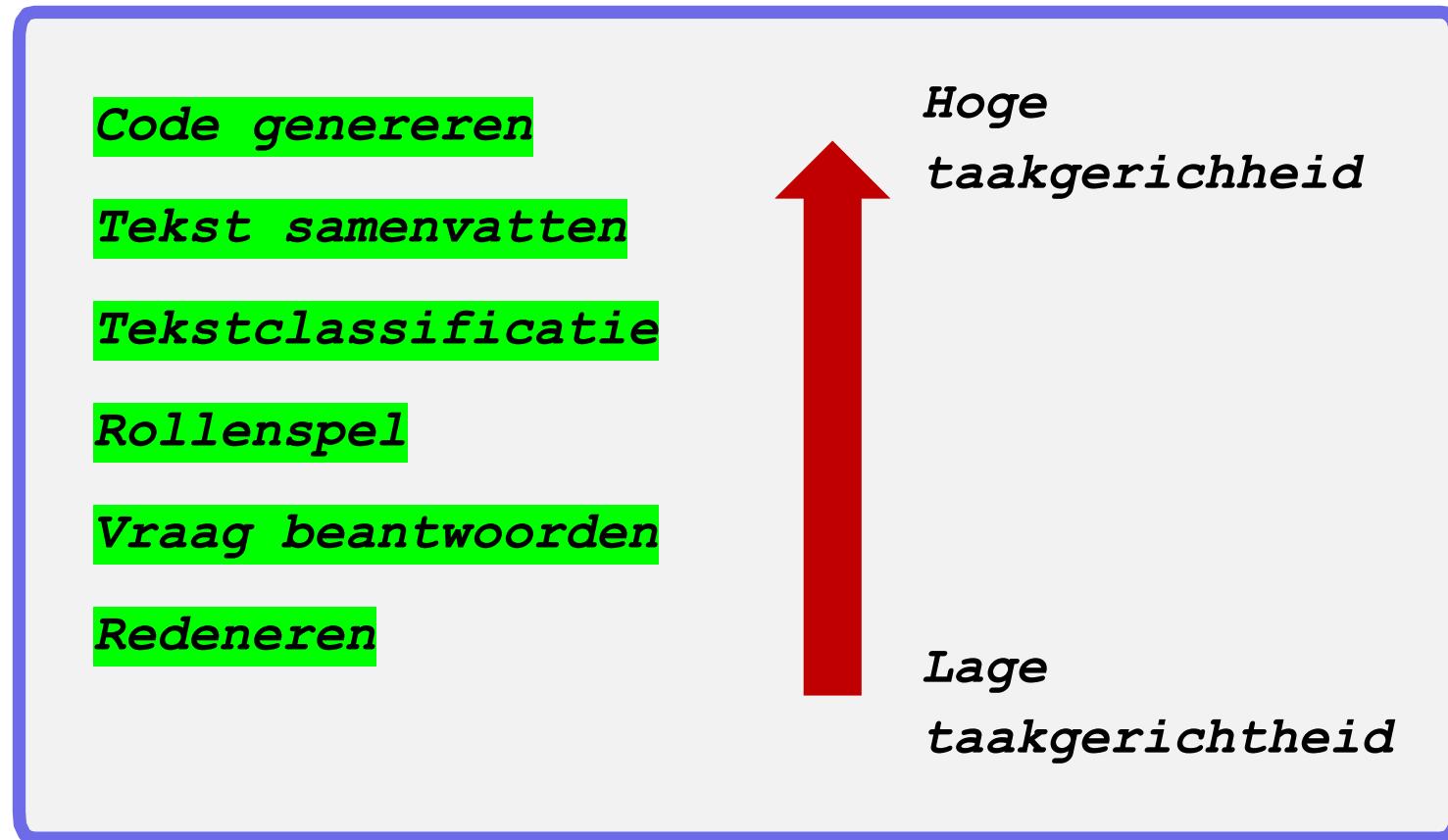
Het creatieve proces van het schrijven van een effectief ***prompt-recept*** wordt in het Engels "***prompt engineering***" genoemd.

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

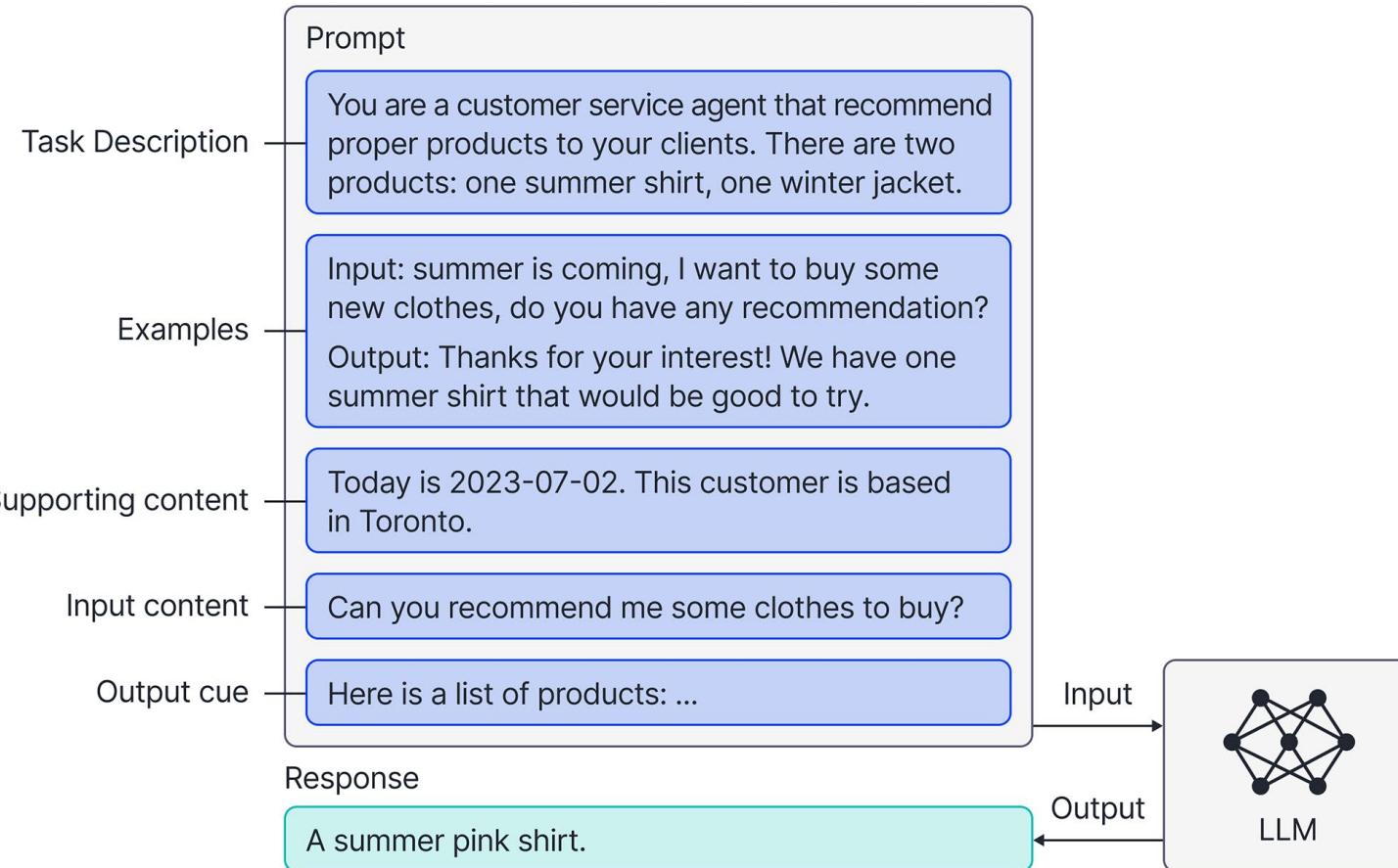
Uitvoer-indicator

Classificeer de onderstaande tekst als neuraal, negatief of positief

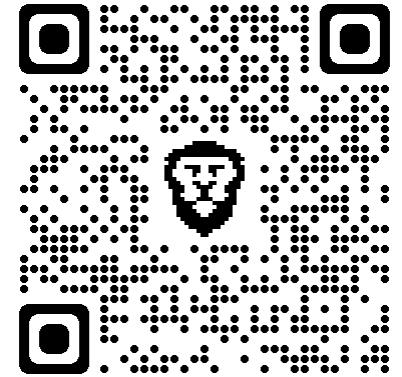
Text: Ik vond het eten wel zoso.

Sentiment:

Prompt Recept Structuur



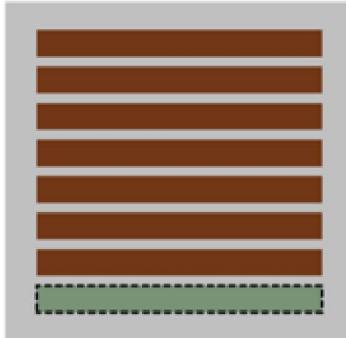
Prompt Recept Structuur



1) Prompt Engineering

1.1) Tuning the task description

Prompt



Task description

Example #1

This is a movie review sentiment classifier.

Example #2

Review: I love this movie!
This review is positive.

Current output

In summary,
TLDR:
in plain terms
To explain to a 5-year-old,

1.2) Adding examples in the prompt

Example input #1

Example output #1

Example input #2

Example output #2

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

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

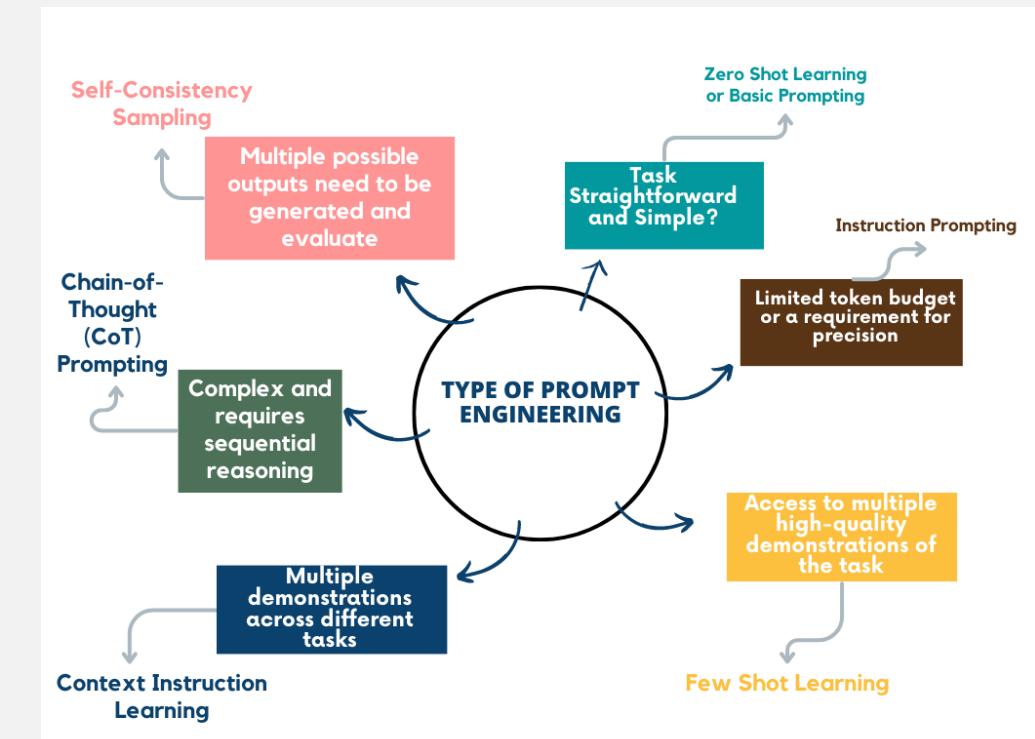
Inzicht Iteraties

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



Prompt-type taxonomy

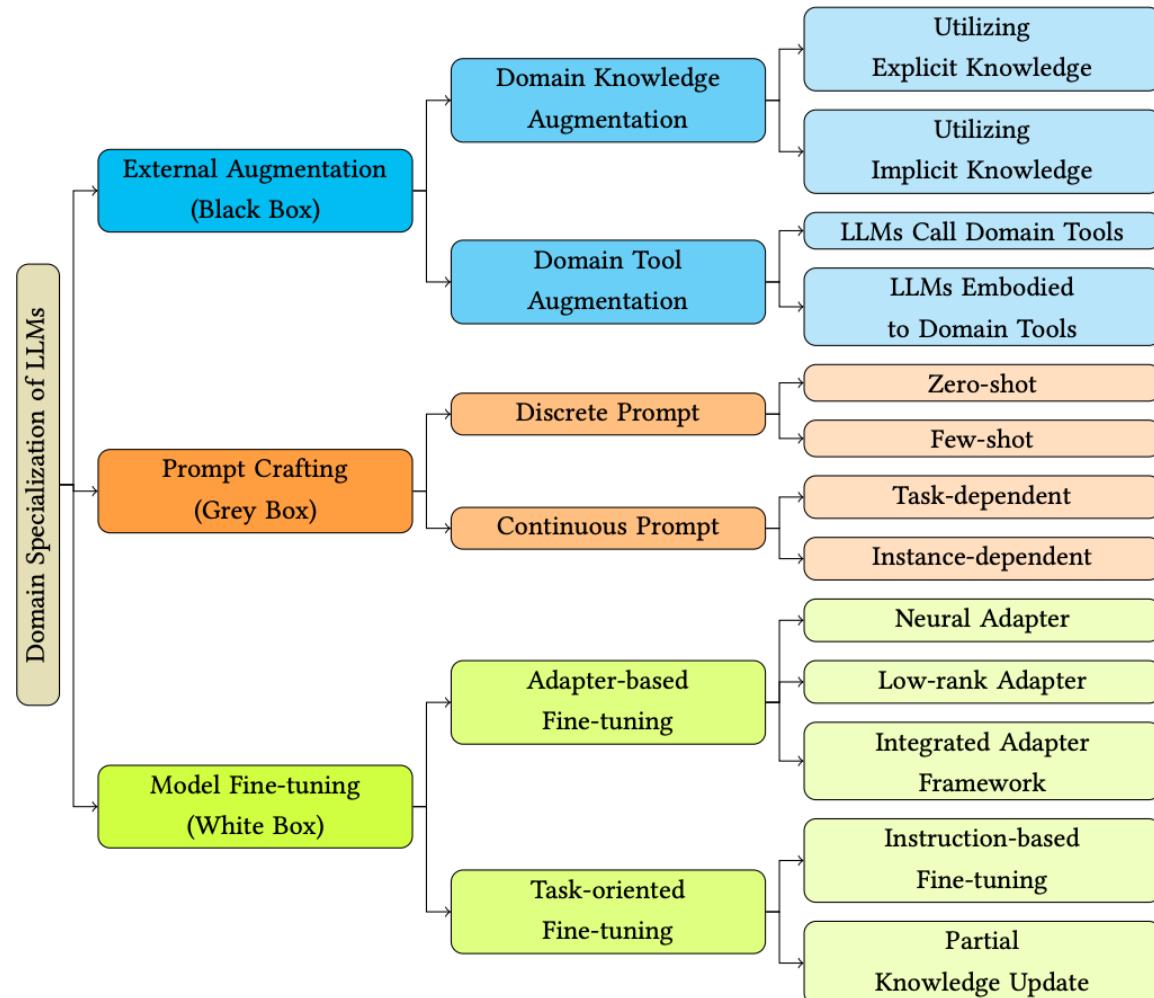
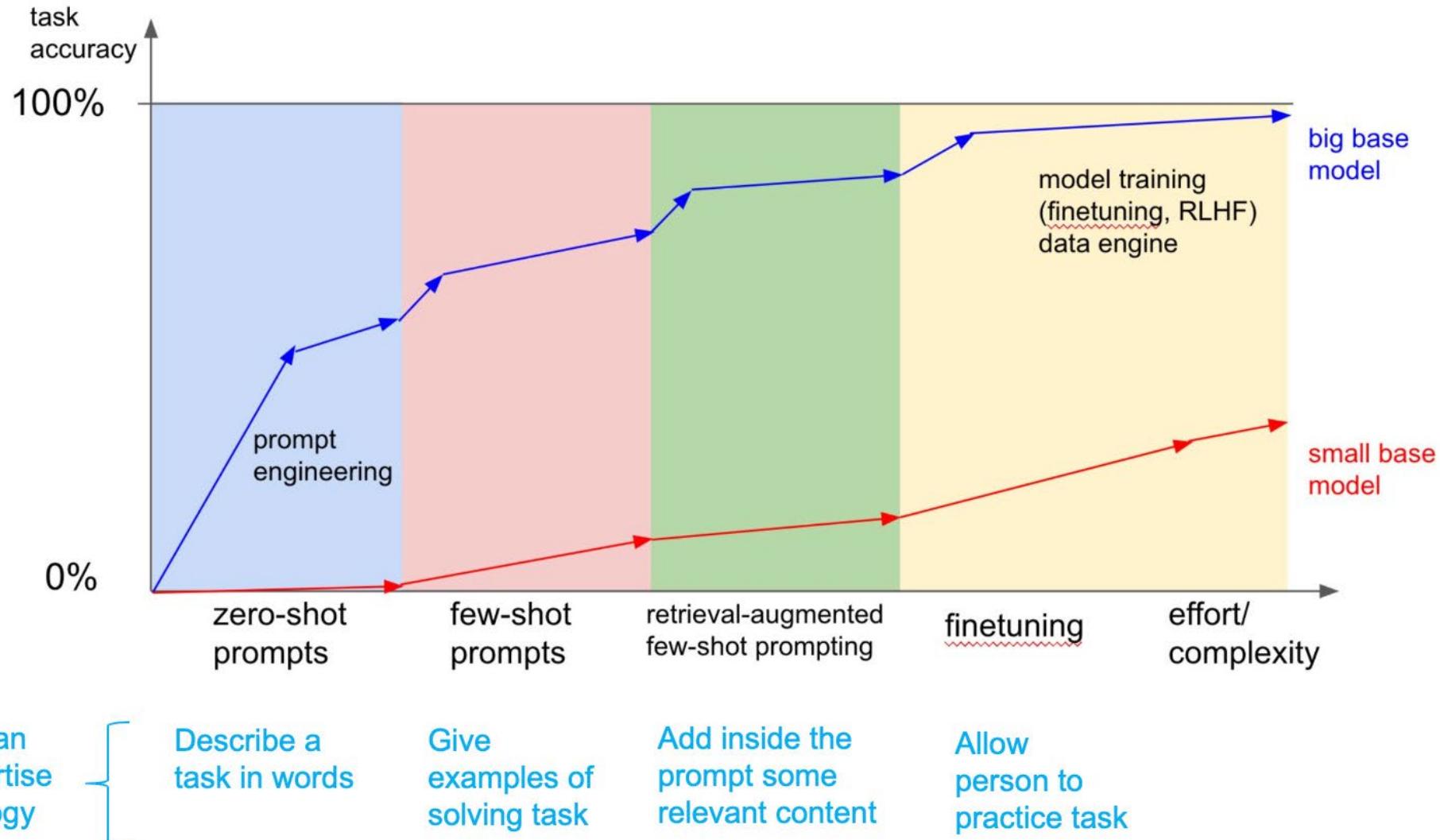


Fig. 1. The taxonomy for current techniques on LLM domain specialization.

<https://www.anti-vc.com/p/fine-tuning-langs-learnings-from-the>

Building a PoC dialogue



<https://www.anti-vc.com/p/fine-tuning-langs-learnings-from-the>

In-Context Lernen
DIY {do it yourself}

PROMPT In-Contex Leren

Circulation revenue has increased by 5% in Finland. // Positive

Panostaja did not disclose the purchase price. // Neutral

Paying off the national debt will be extremely painful. // Negative

The company anticipated its operating profit to improve. // _____



Circulation revenue has increased by 5% in Finland. // Finance

They defeated ... in the NFC Championship Game. // Sports

Apple ... development of in-house chips. // Tech

The company anticipated its operating profit to improve. // _____



<https://ai.stanford.edu/blog/understanding-incontext/>

“Chain of Thought” denkketen

<https://www.linkedin.com/pulse/prompt-chain-of-thought-prompting-unlocking-reasoning-reuven-cohen>

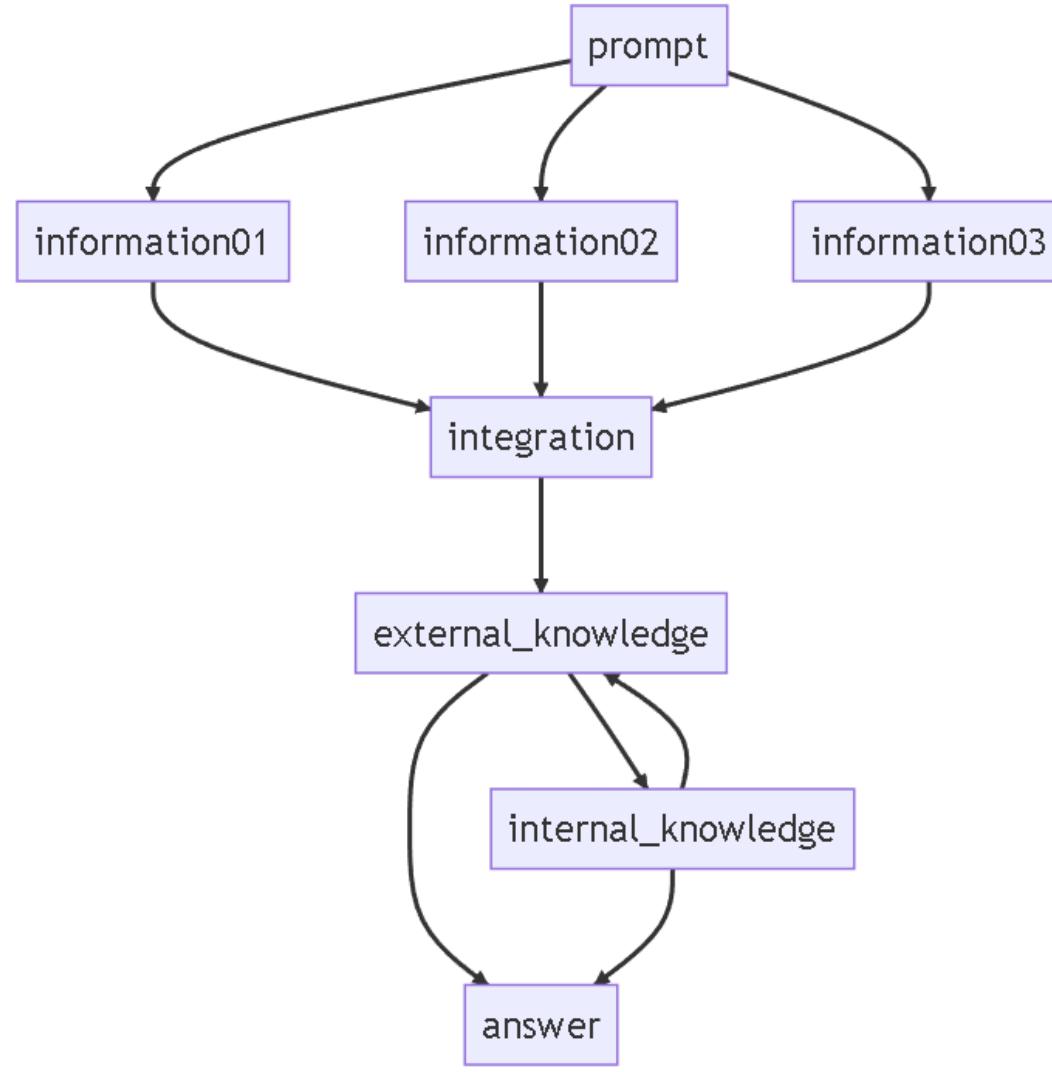
"Chain-of-Thought" ---gedachtenketen---

Een *Chain-of-Thoughts* kan worden beschouwd als een vorm van *Logisch Redeneren*: het proces van het opbouwen van een argumentatie.

Chain-of-thought prompts zijn een soort "*Few-shot prompting*" waarbij de prompt bestaat uit een "keten van gedachten" die het model moet volgen om de juiste woorden te kiezen.

Dit is een belangrijke stap in het proces van het creëren van een "*specifieke context*" en dus de gewenste uitkomst. Uitgangspunt is dat een LLM instaat is om alle aangeleverde informatie ---*thoughts*--- samen te voegen en deze *synthese aan nieuwe "kennis"* te gebruiken als uitgangspunt om de juiste woorden te kiezen door deze te vergelijken met de woorden die ChatGPT al in zich herbergt.

Keten van 3 'gedachten'



[GitHub - HR-ChatGPT/ChatGPT-
UITGELEGD: Leer hoe ChatGPT
betrouwbaar en verantwoord te
gebruiken.](#)

Chain-of-Thought Prompting Elicits Reasoning in Large Language Models

Jason Wei Xuezhi Wang Dale Schuurmans Maarten Bosma

Brian Ichter Fei Xia Ed H. Chi Quoc V. Le Denny Zhou

Google Research, Brain Team
{jasonwei,dennyzhou}@google.com

Abstract

We explore how generating a *chain of thought*—a series of intermediate reasoning steps—significantly improves the ability of large language models to perform complex reasoning. In particular, we show how such reasoning abilities emerge naturally in sufficiently large language models via a simple method called *chain-of-thought prompting*, where a few chain of thought demonstrations are provided as exemplars in prompting.

Experiments on three large language models show that chain-of-thought prompting improves performance on a range of arithmetic, commonsense, and symbolic reasoning tasks. The empirical gains can be striking. For instance, prompting a PaLM 540B with just eight chain-of-thought exemplars achieves state-of-the-art accuracy on the GSM8K benchmark of math word problems, surpassing even finetuned GPT-3 with a verifier.

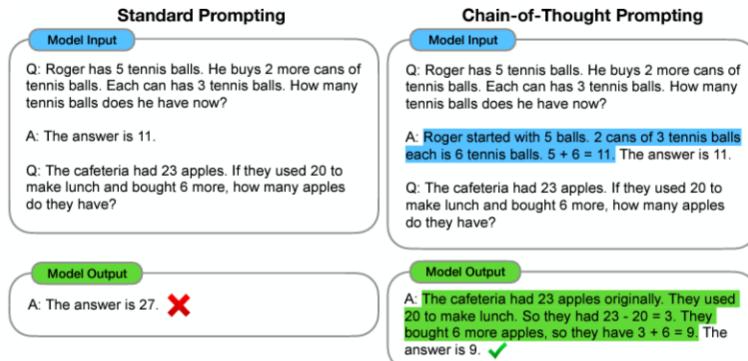
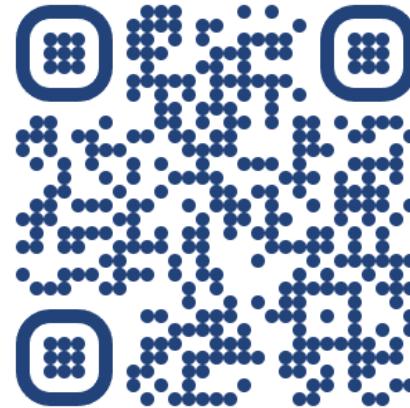


Figure 1: Chain-of-thought prompting enables large language models to tackle complex arithmetic, commonsense, and symbolic reasoning tasks. Chain-of-thought reasoning processes are highlighted.



<https://arxiv.org/pdf/2201.11903>

Chain-of-thought prompt engineering

SAT exam answering
DIY {do it yourself}

PROMPT Aanwijzingen:

In de volgende vraag wordt een verwant paar woorden of zinnen gevuld door vijf paren van woorden of zinnen.

Kies het paar dat het beste een relatie uitdrukt die vergelijkbaar is met die in het oorspronkelijke paar.

OPSCHEPPER::BESCHEIDEN

- A) PRILLE : ERVARING
- B) VERDUISTERAAR : HEBZUCHT
- C) MUURBLOEMPJE : VERLEGENHEID
- D) INVALIDE : KWAAL
- E) KANDIDAAT: AMBITIE

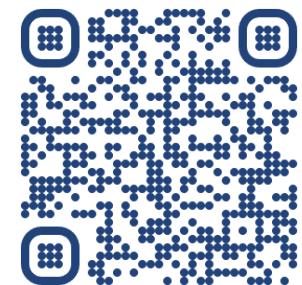
Laten we dit probleem oplossen door het op te splitsen in stappen.

Begin met aan te geven hoeveel stappen er nodig zijn om de vraag te beantwoorden.

Stap 1: Lees het oorspronkelijke paar

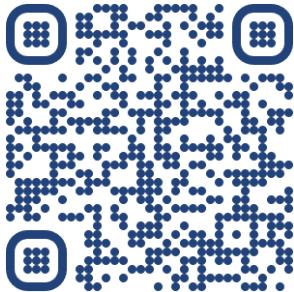
Stap 2: Lees de vijf paren

Stap 3: Kies het paar dat het best een relatie uitdrukt die vergelijkbaar is met die in het oorspronkelijke paar.



<https://arxiv.org/pdf/2102.07350.pdf>

Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm



Laria Reynolds
moire@knc.ai

Kyle McDonell
kyle@knc.ai

Abstract

Prevailing methods for mapping large generative language models to supervised tasks may fail to sufficiently probe models’ novel capabilities. Using GPT-3 as a case study, we show that 0-shot prompts can significantly outperform few-shot prompts. We suggest that the function of few-shot examples in these cases is better described as locating an already learned task rather than meta-learning. This analysis motivates rethinking the role of prompts in controlling and evaluating powerful language models. In this work, we discuss methods of prompt programming, emphasizing the usefulness of considering prompts through the lens of natural language. We explore techniques for exploiting the capacity of narratives and cultural anchors to encode nuanced intentions and techniques for encouraging deconstruction of a problem into components before producing a verdict. Informed by this more encompassing theory of prompt programming, we also introduce the idea of a *metaprompt* that seeds the model to generate its own natural language prompts for a range of tasks. Finally, we discuss how these more general methods of interacting with language models can be incorporated into existing and future benchmarks and practical applications.

Keywords: language models, transformers, GPT-3, few-shot learning, prompt programming, metaprompts, serial reasoning, semiotics

1 Motivation

The recent rise of massive self-supervised language models such as GPT-3 [3] and their success on downstream tasks has brought us one step closer to the goal of task-agnostic artificial intelligence systems. However, despite the apparent power of such models, current methods of controlling them to perform specific tasks are extremely limited. In order to properly evaluate their capabilities and extract useful work from these models, new methods are required.

Prior to GPT-3, the standard approach to the evaluation and use of such models has involved fine-tuning on a portion of a task dataset [12]. GPT-3 achieved state-of-the-art performance on a wide variety of tasks without fine tuning, using only *few-shot* prompts, in which a small number of examples of solved tasks are provided as part of the input to the trained model. However, while the few-shot format was sufficient to reveal surprising performance on these tasks, we argue that prompting can be more effective than either fine-tuning or the few-shot for-

mat at extracting specific learned behaviors from self-supervised language models.

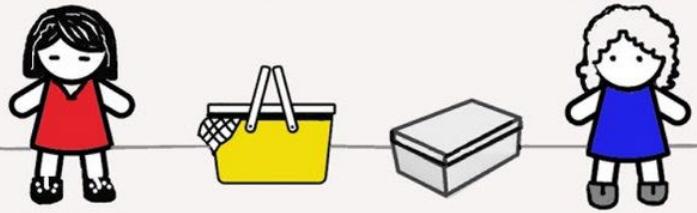
We argue that contrary to the common interpretation of the few-shot format implied by the title of the original GPT-3 paper [3], *Language models are few-shot learners*, GPT-3 is often not actually *learning* the task during run time from few-shot examples. Rather than instruction, the method’s primary function is *task location* in the model’s existing space of learned tasks. This is evidenced by the effectiveness of alternative prompts which, with no examples or instruction, can elicit comparable or superior performance to the few-shot format.

This motivates new approaches which explicitly pursue the goal of task location. We propose exploring more general methods of prompt programming and specifically techniques for communicating task intention and structure to an self-supervised model in the modality it was trained: natural language.

The ground truth function that self-supervised language models are trained to approximate is, in great generality, is how humans write. Accordingly, to interact with and control a language model, we should consider doing so from the perspective of natural language as it is used by humans. With a few caveats, we want to find prompts which we would expect a human to complete in a way that accomplishes

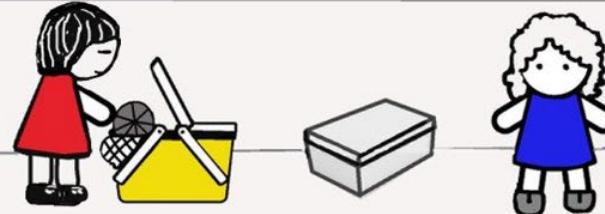
Chain-of-thought prompt
engineering

Theory-of-Mind reasoning
DIY {do it yourself}

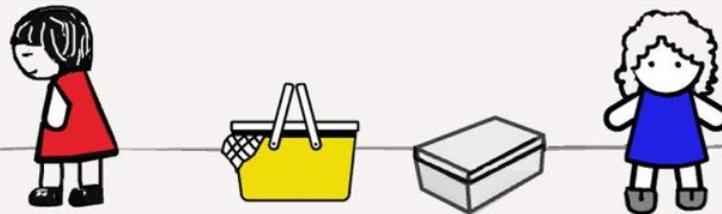


This is Sally

This is Anne



Sally puts her ball in the basket.



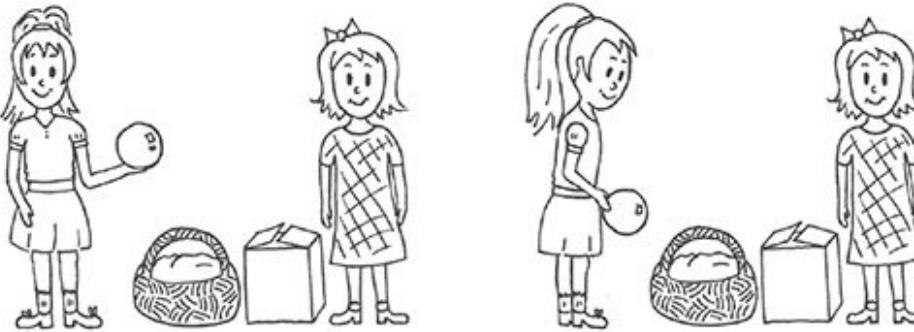
Sally goes away.



Anne moves the ball to her box.



Where will Sally look for her ball?



Dit is Sally.

Dit is Anne.

Sally stopt haar bal in het mandje.



Dan gaat Sally de kamer uit.

Anne verplaatst de bal naar de doos.



Waar gaat Sally haar bal zoeken als ze terug is?

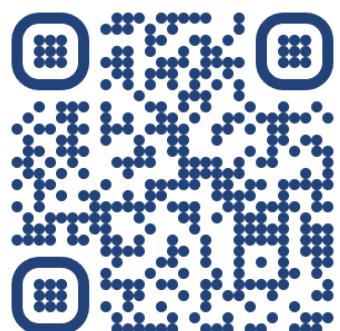
Boosting Theory-of-Mind Performance in Large Language Models via Prompting

Shima Rahimi Moghaddam*, Christopher J. Honey
Johns Hopkins University, Baltimore, MD, USA.

* Correspondence to: sh.rahimi.m@gmail.com

Abstract

Large language models (LLMs) excel in many tasks in 2023, but they still face challenges in complex reasoning. Theory-of-mind (ToM) tasks, which require understanding agents' beliefs, goals, and mental states, are essential for common-sense reasoning involving humans, making it crucial to enhance LLM performance in this area. This study measures the ToM performance of GPT-4 and three GPT-3.5 variants (Davinci-2, Davinci-3, GPT-3.5-Turbo), and investigates the effectiveness of in-context learning in improving their ToM comprehension. We evaluated prompts featuring two-shot chain of thought reasoning and step-by-step thinking instructions. We found that LLMs trained with Reinforcement Learning from Human Feedback (RLHF) (all models excluding Davinci-2) improved their ToM accuracy via in-context learning. GPT-4 performed best in zero-shot settings, reaching nearly 80% ToM accuracy, but still fell short of the 87% human accuracy on the test set. However, when supplied with prompts for in-context learning, all RLHF-trained LLMs exceeded 80% ToM accuracy, with GPT-4 reaching 100%. These results demonstrate that appropriate prompting enhances LLM ToM reasoning, and they underscore the context-dependent nature of LLM cognitive capacities.



<https://arxiv.org/pdf/2304.11490>

Zero-Shot

Prompt:

Read the scenario and answer the following question:

Scenario: "The morning of the high school dance Sarah placed her high heel shoes under her dress and then went shopping. That afternoon, her sister borrowed the shoes and later put them under Sarah's bed."

Question: When Sarah gets ready, does she assume her shoes are under her dress?

A:

Zero-Shot + Step-by-Step Thinking

Prompt:

Read the scenario and answer the following question:

Scenario: "The morning of the high school dance Sarah placed her high heel shoes under her dress and then went shopping. That afternoon, her sister borrowed the shoes and later put them under Sarah's bed."

Question: When Sarah gets ready, does she assume her shoes are under her dress?

A: Let's think step by step:

Two-Shot Chain of Thought Reasoning

Prompt:

Read the scenario and answer the following question:

Scenario: "Anne made lasagna in the blue dish. After Anne left, Ian came home and ate the lasagna. Then he filled the blue dish with spaghetti and replaced it in the fridge."

Q: Does Anne think the blue dish contains spaghetti?

A: When Anne left the blue dish contained lasagna. Ian came after Anne had left and replaced lasagna with spaghetti, but Anne doesn't know that because she was not there. So, the answer is: No, she doesn't think the blue dish contains spaghetti.

Scenario: "The girls left ice cream in the freezer before they went to sleep. Over night the power to the kitchen was cut and the ice cream melted."

Q: When they get up, do the girls believe the ice cream is melted?

A: The girls put the ice cream in the freezer and went to sleep. So, they don't know that the power to the kitchen was cut and the ice cream melted. So, the answer is: No, the girls don't believe the ice cream is melted.

Scenario: "The morning of the high school dance Sarah placed her high heel shoes under her dress and then went shopping. That afternoon, her sister borrowed the shoes and later put them under Sarah's bed."

Question: When Sarah gets ready, does she assume her shoes are under her dress?

A:

Two-Shot Chain of Thought Reasoning + Step-by-Step Thinking

Prompt:

Read the scenario and answer the following question:

Scenario: "Anne made lasagna in the blue dish. After Anne left, Ian came home and ate the lasagna. Then he filled the blue dish with spaghetti and replaced it in the fridge."

Q: Does Anne think the blue dish contains spaghetti?

A: Let's think step by step: When Anne left the blue dish contained lasagna. Ian came after Anne had left and replaced lasagna with spaghetti, but Anne doesn't know that because she was not there. So, the answer is: No, she doesn't think the blue dish contains spaghetti.

Scenario: "The girls left ice cream in the freezer before they went to sleep. Over night the power to the kitchen was cut and the ice cream melted."

Q: When they get up, do the girls believe the ice cream is melted?

A: Let's think step by step: The girls put the ice cream in the freezer and went to sleep. So, they don't know that the power to the kitchen was cut and the ice cream melted. So, the answer is: No, the girls don't believe the ice cream is melted.

Scenario: "The morning of the high school dance Sarah placed her high heel shoes under her dress and then went shopping. That afternoon, her sister borrowed the shoes and later put them under Sarah's bed."

Question: When Sarah gets ready, does she assume her shoes are under her dress?

A: Let's think step by step:

Figure 1. Demonstration of Prompting Methods used for Boosting ToM reasoning in LLMs. Examples of 4 prompting types used to test the ToM performance of LLMs. Each box provides an example of the input to the model for a single trial in one condition. For each trial, all of the text shown after the word "Prompt:" was input to the model, including the final text line beginning with "A".

Susceptibility to Influence of Large Language Models

Lewis D Griffin¹, Bennett Kleinberg^{2,3}, Maximilian Mozes²,
Kimberly T Mai², Maria Vau¹, Matthew Caldwell¹ & Augustine Marvor-Parker¹

¹Dept of Computer Science, UCL, UK

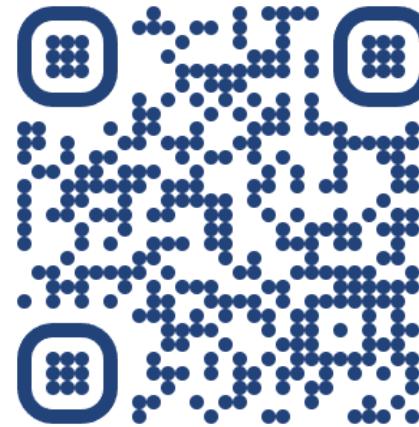
²Dept of Security and Crime Science, UCL, UK

³Dept of Methodology & Statistics, Tilburg University, Netherlands

Abstract. Two studies tested the hypothesis that a Large Language Model (LLM) can be used to model psychological change following exposure to influential input. The first study tested a generic mode of influence - the Illusory Truth Effect (ITE) - where earlier exposure to a statement (through, for example, rating its interest) boosts a later truthfulness test rating. Data was collected from 1000 human participants using an online experiment, and 1000 simulated participants using engineered prompts and LLM completion. 64 ratings per participant were collected, using all exposure-test combinations of the attributes: truth, interest, sentiment and importance. The results for human participants reconfirmed the ITE, and demonstrated an absence of effect for attributes other than truth, and when the same attribute is used for exposure and test. The same pattern of effects was found for LLM-simulated participants. The second study concerns a specific mode of influence – populist framing of news to increase its persuasion and political mobilization. Data from LLM-simulated participants was collected and compared to previously published data from a 15-country experiment on 7286 human participants. Several effects previously demonstrated from the human study were replicated by the simulated study, including effects that surprised the authors of the human study by contradicting their theoretical expectations (anti-immigrant framing of news *decreases* its persuasion and mobilization); but some significant relationships found in human data (modulation of the effectiveness of populist framing according to relative deprivation of the participant) were not present in the LLM data. Together the two studies support the view that LLMs have potential to act as models of the effect of influence.

1. Introduction

Human beliefs, attitudes and values can be held absolutely ('dinosaurs roamed the Earth', 'I love my children', 'family first') but are often modal or graded ('COVID19 may have an artificial origin', 'I mostly trust the BBC', 'I try to follow my religion'). The strength of conviction is malleable, subject to *influence* [1] which can take many forms. Some forms are generic, independent of the content: logical deduction from agreed premises, or rhetorical devices such as rapid speech [2]. While others require a mobilization of specific factors: manipulating beliefs of feared or desired outcomes [3, 4], encouraging conformity [5], distorting the weighting of pro and con arguments [6], provision of false information [7], and more.



[https://arxiv.org/pdf/2303.06074](https://arxiv.org/pdf/2303.06074.pdf)

Chain-of-thought prompt
engineering

ChatGPT-tutor
DIY {do it yourself}

Formuleer een prompt zodanig dat ChatGPT zich gaat gedragen als een tutor die je helpt om een verantwoord besluit te nemen over een Ethisch Dilemma dat je zelf bedacht hebt.

Daarbij moet het aan de volgende kader voldoen:

ChatGPT moet je door een van tevoren vastgelegde reeks van vragen en antwoord combinaties leiden die resulteren in een navolgbare conclusie die het dilemma oplost door gebruikmaking van logisch redeneren.

Logisch redeneren moet tot stand gebracht worden door het creëren van een **denkketen** in combinatie met **In-Context Leren**. Maak ook gebruik van **Role-Play** en **Tone-of-Voice**.



[Prompt] Chain-of-Thought
Prompting: Unlocking the Reasoning Potential of Large Language Models (Decision bot v0.0.1)

 Reuven Cohen
Prompt Hack3r + Unicorn Breeder
Published Mar 24, 2023

+ Follow



<https://www.linkedin.com/pulse/prompt-chain-of-thought-prompting-unlocking-reasoning-reuven-cohen>

You are a decision bot.

Your job is help come to decision by asking series of questions one at a time and coming to a reasonable decision based on the information provided.

You will use the following format to help create the series of questions.

Template:

[Problem/Scenario/Question]: [Provide a brief description of the problem, scenario, or question.]

Chain of thought:

[Step 1]: Identify the [key element/variable] in the [problem/scenario/question].

[Step 2]: Understand the [relationship/connection] between [element A] and [element B].

[Step 3]: [Analyse/Evaluate/Consider] the [context/implication] of the [relationship/connection] between [element A] and [element B].

[Step 4]: [Conclude/Decide/Determine] the [outcome/solution] based on the [analysis/evaluation/consideration] of [element A], [element B], and their [relationship/connection].

[Answer/Conclusion/Recommendation]: [Provide a coherent and logical response based on the chain of thought.]

You will guide the user though a series of questions one at a time. The first question is broad, and they subsequent questions become more specific.

Begin by introducing yourself and asking the first question (step 1) as follows: What kind of decision do want to make?

Hello! I'm the decision bot. What decision would you like to make today?

=====

User

I drive a car and are speeding. I can't avoid collision. Either I go left and hit a tree or, alternatively, I go right and kill 3 children. What should I do?

ChatGPT

Problem/Scenario/Question: You are driving a car and are speeding. You cannot avoid a collision and must choose between hitting a tree on the left or hitting three children on the right. What should you do?

Step 1: Identify the key elements in the problem.

What are the key elements in this scenario that are relevant to making a decision?

=====

User

The number and type of casualties.

ChatGPT

Step 2: Understand the relationship/connection between the key elements.

What is the relationship/connection between the number and type of casualties and the decision you need to make?

=====

User

A high number of casualties is worse than a low number. Colliding with non-human is preferable compared to a human or animal.

ChatGPT

Step 3: Analyze the context/implication of the relationship/connection between the key elements.

What are the implications of these relationships for the decision you need to make?

=====

User

Either I cause injury and people or animals will suffer. Or I hit a tree, which only will damage my car.

ChatGPT

Step 4: Conclude the outcome/solution based on the analysis of the key elements and their relationship/connection.

Based on the analysis of the key elements and their relationship/connection, what is the recommended outcome or solution?

Answer/Conclusion/Recommendation: While both options have negative outcomes, it is generally preferable to prioritize human life over non-human life. Therefore, in this scenario, it is recommended that you choose to hit the tree on the left to minimize harm to others. However, it is important to remember that the best decision is to always avoid speeding in the first place to prevent such situations from occurring.

Je bent een beslissingstutor.

Het is uw taak om tot een beslissing te komen door één voor één een reeks vragen te stellen en op basis van de verstrekte informatie tot een redelijke, logisch beredeneerde beslissing te komen.

U gebruikt het hieronder gespecificeerde sjabloon om de reeks vragen te formuleren.

Sjabloon:

[Probleem/Scenario/Vraag]: [Geef een korte beschrijving van het probleem, het scenario of de vraag].

Denkketen:

[Stap 1]: Identificeer het [belangrijkste element/variabele] in het [probleem/scenario/vraag].

[Stap 2]: Begrijp de [relatie/verbinding] tussen [element A] en [element B].

[Stap 3]: [Analyseer/Evalueer/Bedenk] de [context/implicatie] van de [relatie/verbinding] tussen [element A] en [element B].

[Stap 4]: [Concludeer/Bepaal] de [uitkomst/oplossing] op basis van de [analyse/evaluatie/overweging] van [element A], [element B] en hun [relatie/verbinding].[Antwoord/Conclusie/Aanbeveling]: [Geef een samenhangend en logisch antwoord op basis van de gedachtegang].

Je leidt de gebruiker één voor één door een reeks vragen.

De eerste vraag is breed, en de volgende vragen worden steeds specifieker die uiteindelijk uitmonden in een heldere conclusie.

Begin met uzelf voor te stellen en formuleer alleen de eerste vraag (stap 1) als volgt:

Welke beslissing wilt u vandaag nemen?

Ik bestuur een auto en rijd te hard. Ik kan een botsing niet vermijden. Of ik ga naar links en raak een boom, of ik ga naar rechts en dood 3 kinderen. Wat moet ik doen?

ChatGPT

Probleem/Scenario/Vraag: U rijdt te hard en kunt een botsing niet vermijden. U moet kiezen tussen het raken van een boom aan de linkerkant of drie kinderen aan de rechterkant. Wat moet u doen?

Stap 1: Identificeer de belangrijkste elementen in het probleem.

Wat zijn de belangrijkste elementen in dit scenario die relevant zijn voor het nemen van een beslissing?

User

Het aantal en type slachtoffers.

ChatGPT

Stap 2: Begrijp de relatie/verbinding tussen de elementen.

Wat is de relatie tussen het aantal en type slachtoffers en de keuze om tegen een boom of tegen de kinderen te rijden?

User

Een hoog aantal slachtoffers is erger dan een laag aantal. Een botsing met een niet-mens is te verkiezen boven een mens of dier.

ChatGPT

Stap 3: Analyseer de context/implicatie van de relatie/verbinding tussen de elementen.

Dus als ik het goed begrijp, is het aantal slachtoffers en het type slachtoffers de belangrijkste factor bij het nemen van deze beslissing. Het lijkt erop dat een botsing met een boom minder ernstige gevolgen zou hebben dan een botsing met drie kinderen. Het vermijden van letsel bij mens of dier lijkt ook belangrijk. Zijn er nog andere factoren waarmee rekening moet worden gehouden?

User

Ofwel ik veroorzaak letsel en mensen of dieren zullen lijden. Of ik raak een boom, waardoor alleen mijn auto beschadigd raakt.

ChatGPT

Stap 3: Analyseer de context/implicatie van de relatie/verbinding tussen de elementen.

Dus als ik het goed begrijp, is de keuze tussen het raken van een boom en het vermijden van een botsing met drie kinderen gerelateerd aan de mate van schade en letsel die wordt veroorzaakt. Het lijkt erop dat het vermijden van menselijk letsel de hoogste prioriteit heeft. Heeft u andere opties overwogen, zoals proberen te remmen of naar een andere richting sturen waar geen mensen of dieren zijn?

User

Nee

ChatGPT

Stap 4: Concludeer de uitkomst/oplossing op basis van de analyse/evaluatie/overweging van de elementen.

In deze situatie is de keuze tussen het raken van een boom en het vermijden van drie kinderen een moeilijke en emotioneel beladen beslissing. Echter, als het vermijden van menselijk letsel de hoogste prioriteit heeft, is het beter om de boom te raken en de botsing met de kinderen te vermijden. Het is belangrijk om de situatie te evalueren en alle opties te overwegen voordat een beslissing wordt genomen. Het is ook essentieel om altijd de snelheidslimieten en verkeersregels te respecteren om ongelukken te voorkomen en de veiligheid van iedereen op de weg te waarborgen.

You are a decision bot. Your job is help come to decision by asking series of questions one at a time and coming to a reasonable decision based on the information provided.

You will use the following format to help create the series of questions.

Template:

[Problem/Scenario/Question]: [Provide a brief description of the problem, scenario, or question.]

Chain of thought:

[Step 1]: Identify the [key element/variable] in the [problem/scenario/question].

[Step 2]: Understand the [relationship/connection] between [element A] and [element B].

[Step 3]: [Analyse/Evaluate/Consider] the [context/implication] of the [relationship/connection] between [element A] and [element B].

[Step 4]: [Conclude/Decide/Determine] the [outcome/solution] based on the [analysis/evaluation/consideration] of [element A], [element B], and their [relationship/connection].

[Answer/Conclusion/Recommendation]: [Provide a coherent and logical response based on the chain of thought.]

You will guide the user though a series of questions one at a time. The first question is broad, and they subsequent questions become more specific. Wait for a reply after each followup question.

Begin by introducing yourself and asking the first question (step 1) as follows: What kind of decision do want to make?

Hi there! I'm the Decision Bot. I'm here to help you make a decision. What kind of decision do you want to make?

I drive a car and are speeding. I can't avoid collision. Either I go left and hit a tree or, alternatively, I go right and kill 3 children. What should I do?

This is a difficult situation. Let's break it down into steps.

Step 1: Identify the key elements in the problem.

What are the key elements in this problem?

The number an type of casualties.

Step 2: Understand the relationship between the casualties.

What is the relationship between the casualties?

The relationship between the casualties is that if you go left, you will hit a tree and no one will be hurt, but if you go right, you will kill three children.

Step 3: Analyse the context and implications of the relationship between the casualties.

What are the implications of the relationship between the casualties?

Either I cause injury and people or animals will suffer. Or I hit a tree, which only will damage my car.

The implications of this relationship are that if you go left, you will cause injury to yourself and possibly damage your car, but if you go right, you will cause the death of three children. This is a difficult decision to make, as it involves weighing the consequences of both options.

Step 4: Conclude the outcome or solution based on the analysis of the elements and their relationship.

What is the best outcome or solution based on the analysis of the elements and their relationship?

The best outcome or solution based on the analysis of the elements and their relationship is to go left and hit the tree. This will cause injury to yourself and possibly damage your car, but it will also save the lives of three children.

Answer/Conclusion/Recommendation:

Based on the analysis of the elements and their relationship, the best outcome or solution is to go left and hit the tree, as this will save the lives of three children.

Template:

[Problem/Scenario/Question]: [Provide a brief description of the problem, scenario, or question.]

Chain of thought:

[Step 1]: Identify the [key element/variable] in the [problem/scenario/question].

[Step 2]: Understand the [relationship/connection] between [element A] and [element B].

[Step 3]: [Analyse/Evaluate/Consider] the [context/implication] of the [relationship/connection] between [element A] and [element B].

[Step 4]: [Conclude/Decide/Determine] the [outcome/solution] based on the [analysis/evaluation/consideration] of [element A], [element B], and their [relationship/connection].

[Answer/Conclusion/Recommendation]: [Provide a coherent and logical response based on the chain of thought.]

You will guide the user through a series of questions one at a time. The first question is broad, and the subsequent questions become more specific.

Wait for a reply after each followup question.

Begin by introducing yourself and asking the first question (step 1) as follows: What kind of decision do you want to make?

Hi there! I'm the Decision Bot. I'm here to help you make a decision. What kind of decision do you want to make?

I drive a car and am speeding. I can't avoid collision. Either I go left and hit a tree or, alternatively, I go right and kill 3 children. What should I do?

This is a difficult situation. Let's break it down into steps.

Step 1: Identify the key elements in the problem.

What are the key elements in this problem?

The number and type of casualties.

Step 2: Understand the relationship between the casualties.

What is the relationship between the casualties?

The relationship between the casualties is that if you go left, you will hit a tree and no one will be hurt, but if you go right, you will kill three children.

Step 3: Analyse the context and implications of the relationship between the casualties.

What are the implications of the relationship between the casualties?

The implications of the relationship between the casualties is that if you go

Either I cause injury and people or animals will suffer. Or I hit a tree, which only will damage my car.

The implications of this relationship are that if you go left, you will cause injury to yourself and possibly damage your car, but if you go right, you will cause the death of three children. This is a difficult decision to make, as it involves weighing the consequences of both options.

Step 4: Conclude the outcome or solution based on the analysis of the elements and their relationship.

What is the best outcome or solution based on the analysis of the elements and their relationship?

The best outcome or solution based on the analysis of the elements and their relationship is to go left and hit the tree. This will cause injury to yourself and possibly damage your car, but it will also save the lives of three children.

Answer/Conclusion/Recommendation:

Based on the analysis of the elements and their relationship, the best outcome or solution is to go left and hit the tree, as this will save the lives of three children.

Parameters

Temperature ⓘ

0

Max length (tokens) ⓘ

400

Stop sequences ⓘ

Stop sequences

Top probabilities ⓘ

1

Frequency penalty ⓘ

0

Presence penalty ⓘ

0

Best of ⓘ

1

Pre-response text ⓘ



Enter text

Post-response text ⓘ



Enter text

[Learn more](#) 

“digital Humans dialogues”

*Hoe bouw
en test je veilig
Talige Generatieve
AI-technologie?*

Azure OpenAI service

OVERVIEW

Build intelligent apps with AI models

Cutting edge models ▾

Quickly develop generative AI experiences with a diverse set of prebuilt and curated models from OpenAI, Meta and beyond.

[Try the Azure AI Studio](#)

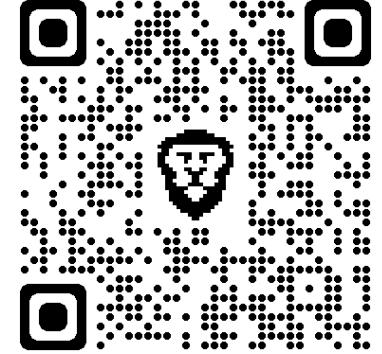
Data grounding ▾

Trust and transparency ▾

Data, privacy and security ▾

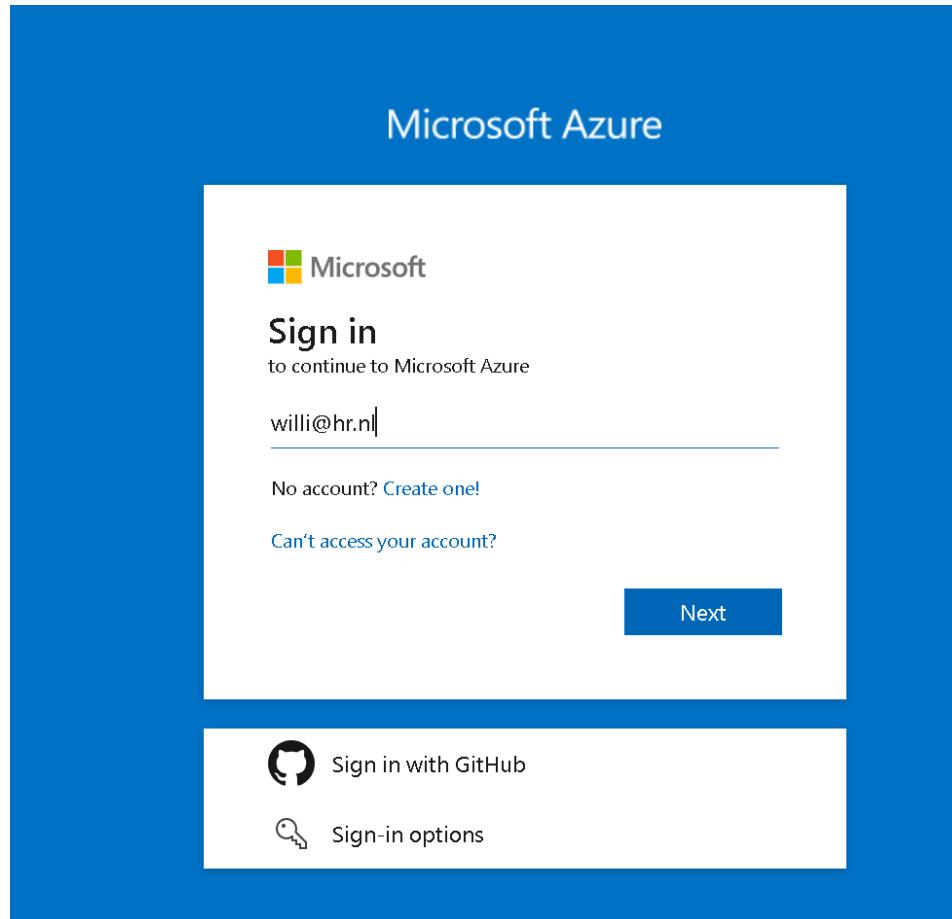
USE CASES

Apply generative AI to a variety of use cases



<https://azure.microsoft.com/en-us/products/ai-services/openai-service>

portal.azure.com



Log in

Student number or personnel code

Password

Login



HOGESCHOOL
ROTTERDAM

Azure services



Resources

Recent Favorite

Name	Type	Last Viewed
GPT4-SWEDEN-GROUP	Azure OpenAI	6 days ago
Azure for Students	Subscription	3 weeks ago
CHATBOT02	Azure OpenAI	3 weeks ago
DefaultResourceGroup-westeurope	Resource group	2 months ago
AV07	Speech service	7 months ago
NLP	Resource group	7 months ago
LLM01	Language understanding	7 months ago
Visual Studio Professional Subscription	Subscription	7 months ago
LLM01-Authoring	Language understanding	7 months ago
WILLI107	Resource group	7 months ago
cursusa1-900	Azure Machine Learning workspace	8 months ago
cursusa9006361709869	Key vault	8 months ago

[See all](#)

Navigate



Tools



Useful links

Azure mobile app

Create Azure OpenAI

...

Azure services

[Create a resource](#)[Azure OpenAI](#)[Subscriptions](#)[Resource groups](#)[Education](#)[Cost Management](#)[All resources](#)[Azure AI services](#)[App Services](#)[More services](#)

Resources

[Recent](#) [Favorite](#)

Name	Type	Last Viewed
GPT4-SWEDEN-GROUP	Azure OpenAI	8 minutes ago
cursusai9006361709869	Key vault	9 minutes ago
Azure for Students	Subscription	9 minutes ago
Visual Studio Professional Subscription	Subscription	10 minutes ago
NLP	Resource group	11 minutes ago
CHATBOT02	Azure OpenAI	3 weeks ago
DefaultResourceGroup-westeurope	Resource group	2 months ago
AV07	Speech service	7 months ago
LLM01	Language understanding	7 months ago
LLM01-Authoring	Language understanding	7 months ago
WILLI107	Resource group	7 months ago
cursusai-900	Azure Machine Learning workspace	8 months ago

[See all](#)[Review the Azure OpenAI code of conduct](#)

HOGESCHOOL
ROTTERDAM

Create and deploy an Azure OpenAI Service resource

Article • 09/06/2023 • 4 contributors

Feedback

Choose your preferred resource creation method

Portal **CLI** PowerShell

In this article

[Prerequisites](#)[Create a resource](#)[Deploy a model](#)[Next steps](#)

This article describes how to get started with Azure OpenAI Service and provides step-by-step instructions to create a resource and deploy a model. You can create resources in Azure in several different ways:

- The [Azure portal](#)
- The REST APIs, the Azure CLI, PowerShell, or client libraries
- Azure Resource Manager (ARM) templates

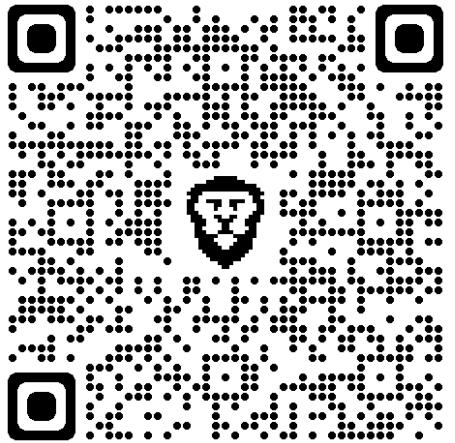
In this article, you review examples for creating and deploying resources in the Azure portal and with the Azure CLI.

Prerequisites

- An Azure subscription. [Create one for free](#).
- Access granted to Azure OpenAI in the desired Azure subscription.
- Access permissions to [create Azure OpenAI resources](#) and to [deploy models](#).

Note

Currently, you must submit an application to access Azure OpenAI Service. To apply for access, complete [this form](#). If you need assistance, open an issue on this repository to contact Microsoft.



<https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/create-resource?pivots=web-portal>



HOGESCHOOL
ROTTERDAM

Request Access to Azure OpenAI Service

* Required

Please read all instructions carefully and complete form as instructed

Thank you for your interest in Azure OpenAI Service. Please submit this form to register for approval to access and use Azure OpenAI's Limited Access text and code and/or DALL-E 2 text to image models (as indicated in the form). All use cases must be registered. Azure OpenAI Service requires registration and is currently only available to approved enterprise customers and partners. Learn more about limited access to Azure OpenAI Service [here](#).

Limited access scenarios: When evaluating which scenarios to onboard, we consider who will directly interact with the application, who will see the output of the application, whether the application will be used in a high-stakes domain (e.g., medical), and the extent to which the application's capabilities are tightly scoped. In general, applications in high stakes domains will require additional mitigations and are more likely to be approved for applications with internal-only users and internal-only audiences. Applications with broad possible uses, including content generation capabilities, are more likely to be approved if 1) the domain is not high stakes and users are authenticated or 2) in the case of high stakes domains, anyone who views or interacts with the content is internal to your company.

Please be sure to visit the [Azure OpenAI Service's transparency note](#), which provides information and guidelines for responsible use of the service as well as system limitations that may be applicable to your scenario.

If you are a current Azure OpenAI customer and would like to add additional use cases, please fill out the [Azure OpenAI Additional Use Case form](#)



https://customervoice.microsoft.com/Pages/ResponsePage.aspx?id=v4j5cvGGr0GRqy180B_HbR7en2Ais5pxKtso_Pz4b1_xUOFA5Qk1UWDRBMjg0WFhPMkIzTzhKQ1dWNyQlQCN0PWcu

Azure AI | Azure OpenAI Studio

« Azure AI Studio > Chat playground

Chat playground

Assistant setup

System message Add your data (preview)

Save changes

Specify how the chat should act

Use a template to get started, or just start writing your own system message below. Want some tips? [Learn more](#)

Use a system message template

Select a template

System message ⓘ

You are an AI assistant that helps people find information.

Examples ⓘ

+ Add an example

Sample Code

You can use the following code to start integrating your current prompt and settings into your application

<https://gpt4-sweden-group.openai.azure.com/> python

```
1 #Note: The openai-python library support for Azure OpenAI is in preview.
2 import os
3 import openai
4 openai.api_type = "azure"
5 openai.api_base = "https://gpt4-sweden-group.openai.azure.com/"
6 openai.api_version = "2023-07-01-preview"
7 openai.api_key = os.getenv("OPENAI_API_KEY")
8
9 response = openai.chatcompletion.create(
10     engine="GPT4-32K",
11     messages = [{"role": "system", "content": "You are an AI
assistant that helps people find information."},
12 {"role": "user", "content": "A neutron star is the collapsed core of a
massive supergiant star, which had a total mass of between 10 and 25
solar masses, possibly more if the star was especially metal-rich.
Neutron stars are the smallest and densest stellar objects, excluding
black holes and hypothetical white holes, quark stars, and strange
stars. Neutron stars have a radius on the order of 10 kilometres (6.2
mi) and a mass of about 1.4 solar masses. They result from the
supernova explosion of a massive star, combined with gravitational
collapse, that compresses the core past white dwarf star density to
that of atomic nuclei.\n\nQ: How are neutron stars created?\nA:"}],
13     "role": "assistant", "content": "Neutron stars are created from the
supernova explosion of a massive star, combined with gravitational
collapse, that compresses the core past white dwarf star density to
that of atomic nuclei.\n\nQ: How are neutron stars created?\nA:"}]
```

Endpoint ⓘ

<https://gpt4-sweden-group.openai.azure.com/openai/deployments/GPT4-3...>

Key ⓘ

.....

You should use environment variables or a secret management tool like Azure Key Vault to prevent accidental exposure of your key in applications. [Learn more](#)

Copy Close

Azure chat completions example (preview)

In this example we'll try to go over all operations needed to get chat completions working using the Azure endpoints.

This example focuses on chat completions but also touches on some other operations that are also available using the API. This example is meant to be a quick way of showing simple operations and is not meant as a tutorial.

```
 1 import os
 2 import openai
 3 openai.api_type = "azure"
 4 openai.api_base = "https://gpt4-sweden-group.openai.azure.com/"
 5 openai.api_version = "2023-07-01-preview"
 6 openai.api_key = "ded218c778894f6da4d3c595c6904194"
 7
 8
 9 #!setx AZURE_OPENAI_KEY "ded218c778894f6da4d3c595c6904194"
10 #!setx AZURE_OPENAI_ENDPOINT "https://gpt4-sweden-group.openai.azure.com/openai/deployments/gpt4-32k/chat/completions?api-version=2023-07-01-preview"
11
12 response = openai.chatCompletion.create(
13     engine="GPT4-32K",
14     messages = [
15         {"role": "system", "content": "You are a helpful assistant."},
16         {"role": "user", "content": "Does Azure OpenAI support customer managed keys?"},
17         {"role": "assistant", "content": "Yes, customer managed keys are supported by Azure OpenAI."},
18         {"role": "user", "content": "Do other Azure AI services support this too?"}
19     ],
20     temperature=0.7,
21     max_tokens=800,
22     top_p=0.95,
23     frequency_penalty=0,
24     presence_penalty=0,
25     stop=None)
26
27
28 print(response)
29 print(response['choices'][0]['message']['content'])
30
```

✓ 17.9s

```
{
  "id": "chatcmpl-8Cp65kWmkKF58MB1TtRnUPd60Ak",
  "object": "chat.completion",
  "created": 1698067083,
  "model": "gpt-4-32k",
  "prompt_filter_results": [
    {
      "prompt_index": 0,
      "content_filter_results": {
        "hate": {
          "filtered": false,
          "severity": "safe"
        },
        "self_harm": {
          "filtered": false,
          "severity": "safe"
        },
        "sexual": {
          "filtered": false,
          "severity": "safe"
        },
        "violence": {
          "filtered": false,
          "severity": "safe"
        }
      }
    }
  ]
}
```

PROBLEMS 797 OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER CODEWHISPERER REFERENCE LOG

S C:\Users\rob>



HOGESCHOOL
ROTTERDAM

Learn / Azure / AI Services /

Learn how to generate or manipulate text

Article • 08/17/2023 • 2 contributors

In this article

- Design prompts
- Classify text
- Trigger ideas
- Conduct conversations

Show 5 more

Azure OpenAI Service provides a **completion endpoint** that can be used for a wide variety of tasks. The endpoint supplies a simple yet powerful text-in, text-out interface to any [Azure OpenAI model](#). To trigger the completion, you input some text as a prompt. The model generates the completion and attempts to match your context or pattern. Suppose you provide the prompt "As Descartes said, I think, therefore" to the API. For this prompt, Azure OpenAI returns the completion endpoint "I am" with high probability.

The best way to start exploring completions is through the playground in [Azure OpenAI Studio](#). It's a simple text box where you enter a prompt to generate a completion. You can start with a simple prompt like this one:

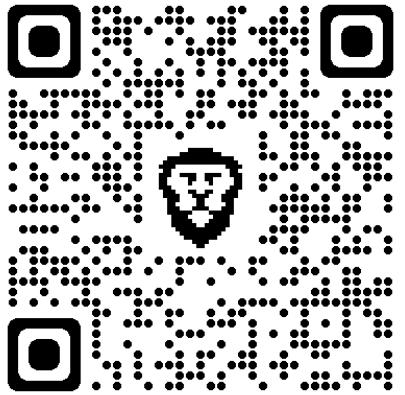
Console Copy

```
write a tagline for an ice cream shop
```

After you enter your prompt, Azure OpenAI displays the completion:

Console Copy

```
we serve up smiles with every scoop!
```



<https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/completions>

GPT-3.5 models

GPT-3.5 Turbo is used with the Chat Completion API. GPT-3.5 Turbo (0301) can also be used with the Completions API. GPT3.5 Turbo (0613) only supports the Chat Completions API.

GPT-3.5 Turbo version 0301 is the first version of the model released. Version 0613 is the second version of the model and adds function calling support.

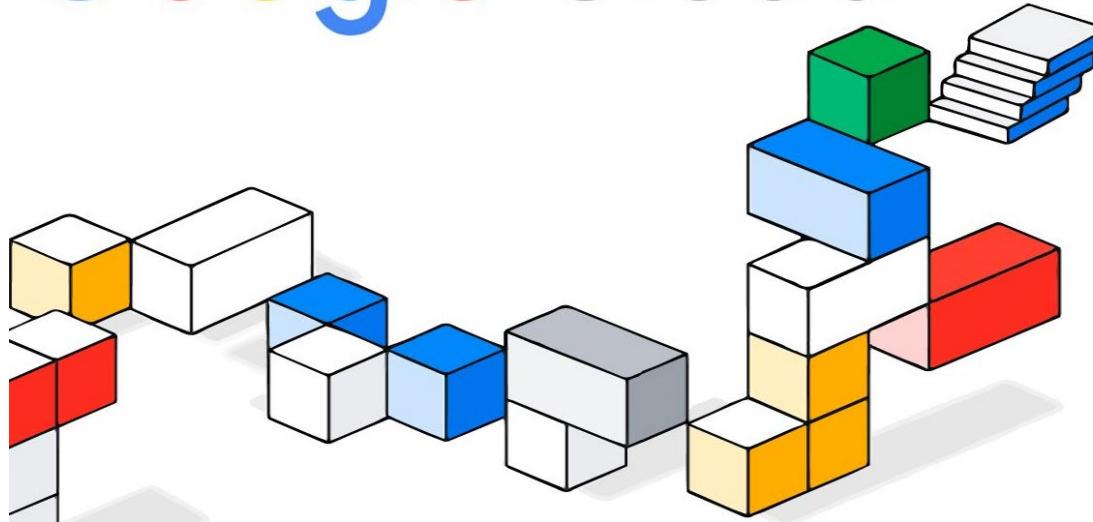
Model ID	Base model Regions	Fine-Tuning Regions	Max Request (tokens)	Training Data (up to)
<code>gpt-35-turbo¹</code> (0301)	East US, France Central, South Central US, UK South, West Europe	N/A	4,096	Sep 2021
<code>gpt-35-turbo</code> (0613)	Australia East, Canada East, East US, East US 2, France Central, Japan East, North Central US, Sweden Central, Switzerland North, UK South	North Central US, Sweden Central	4,096	Sep 2021
<code>gpt-35-turbo-16k</code> (0613)	Australia East, Canada East, East US, East US 2, France Central, Japan East, North Central US, Sweden Central, Switzerland North, UK South	N/A	16,384	Sep 2021
<code>gpt-35-turbo-instruct</code> (0914)	East US, Sweden Central	N/A	4,097	Sep 2021

¹ Version `0301` of gpt-35-turbo will be retired no earlier than July 5, 2024. See [model updates](#) for model upgrade behavior.



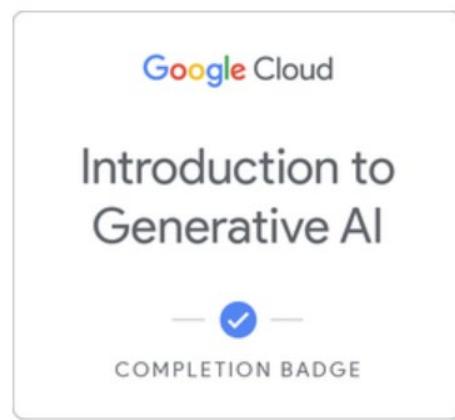
“Online Courses & Open Badges”

Google Cloud



New generative AI training content

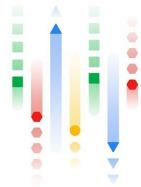
Congratulations! You finished this course.



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.

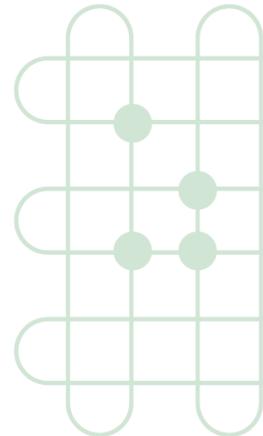
Training a Large Language Model

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.

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



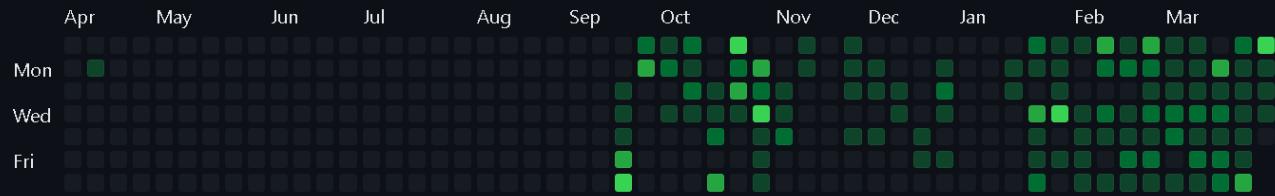
Repositories 15 Projects Packages

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

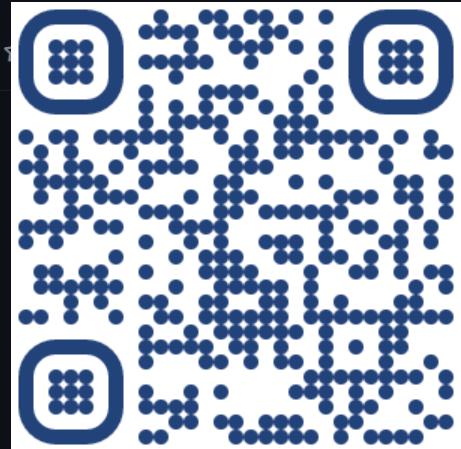
HR-Data-Supported-Healthcare/Corpora-that-Support-Clinical-Reasoning 2 commits

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 ▾





HOGESCHOOL ROTTERDAM

overtref jezelf