

Introduction to Generative AI - Natural Language Processing

Meng Fang
OxML 2024



Outline

Introduction to Generative AI and LLMs

Prompt Engineering

Expanding knowledge for LLMs

Language agents

What is Generative AI?

Artificial intelligence systems that can produce high quality content, specifically text, images, and audio.

Alan Turing: Computer Pioneer

ChatGPT 3.5

You who is Alan Turing?

ChatGPT Alan Turing was a British mathematician, logician, and computer scientist who is widely considered one of the fathers of modern computer science. He was born on June 23, 1912, in Maida Vale, London, and died on June 7, 1954, in Wilmslow, Cheshire, England.

During World War II, Turing played a crucial role in breaking German codes, contributing significantly to the Allied victory. He worked at Bletchley Park, the UK's codebreaking center, where he led a team that developed the Bombe, an

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Gemini

Gemini was just updated. See update

who is Alan Turing?

Alan Turing was a brilliant British mathematician and computer scientist who made significant contributions in various fields. He is widely considered the:

- Father of theoretical computer science: Turing's concept of the Turing machine laid the foundation for modern computers.
- Pioneer of artificial intelligence: His Turing test is a benchmark for a machine's ability to exhibit intelligent behavior equivalent to, or indistinguishable from, that of a human.

Turing's wartime achievements were also crucial. He was a key figure at Bletchley Park, the UK's codebreaking center, where his work on deciphering German Enigma messages is believed to have shortened the war and saved countless lives.

Sadly, Turing's life ended tragically. His homosexuality, then criminalized in Britain, led to his chemical castration and eventual suicide. He has since been posthumously pardoned by the British government.

Enter a prompt here

Microsoft Bing SEARCH CHAT

Moe 258 Mobile Choose a conversation style

More Creative More Balanced More Precise

who is Alan Turing?

Searching for: Alan Turing

Generating answers for you...



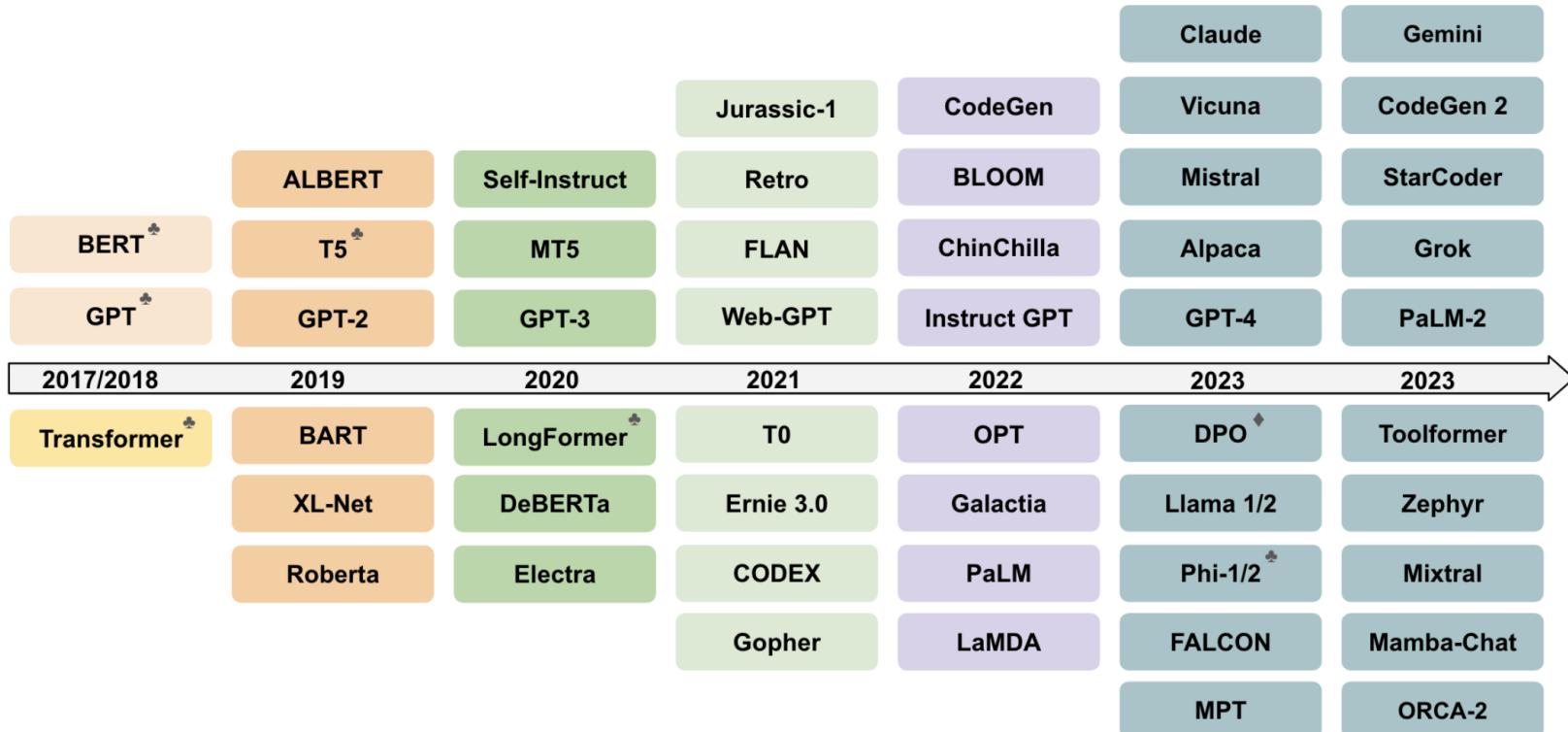
Explore

Alan Turing was an English mathematician, computer scientist, logician, cryptanalyst, philosopher, and theoretical biologist. He is widely considered to be the father of theoretical computer science and artificial intelligence. During World War II, he worked for the

Ask me anything...

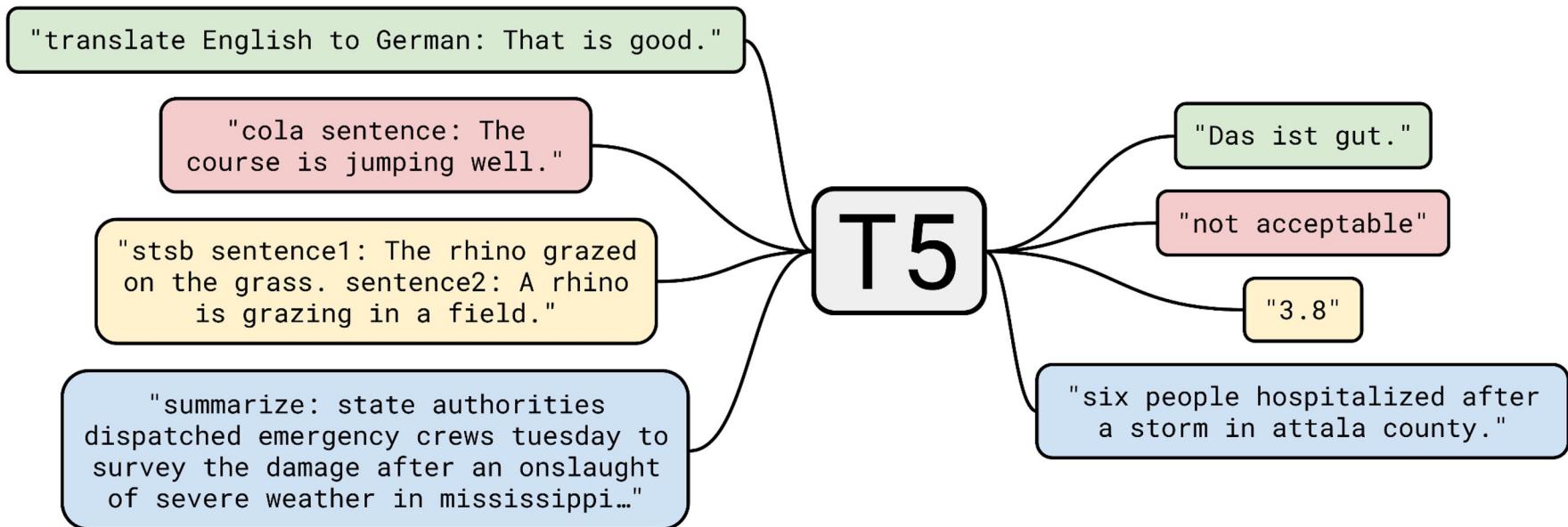
0/2000

The rise of generative AI - Large Language Models



Examples: T5

Text-to-text-transfer-transformer



Examples: GPT-3

Generative Pre-trained Transformer 3

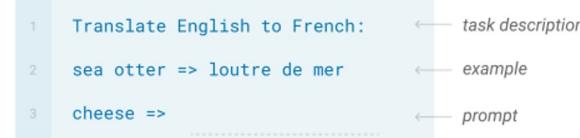
Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.



One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.



Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



Predicting the next token

From Machine Learning to Generative AI

AI history

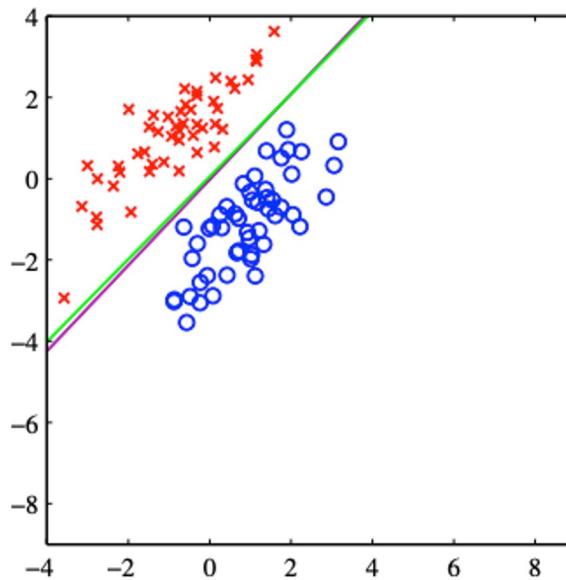
- 1956, Artificial intelligence
- 1997, Machine Learning
- 2017, Deep Learning
- 2021, Generative AI

Supervised learning (labeling things)

Input: a text or an image

Output: a label

Classification



Language modelling

Language Modeling is the task of predicting what word comes next

I like trying new _____
foods
hobbies
products
activities

More formally, given a sequence of words, language modelling can compute the probability distribution of the next word:

- $P(\text{foods} \mid \text{I like trying new}) , P(\text{hobbies} \mid \text{I like trying new}), \dots$

A system that does this is called a Language Model

Generating text using Large Language Models (LLMs)

Text generation process

prompt / input



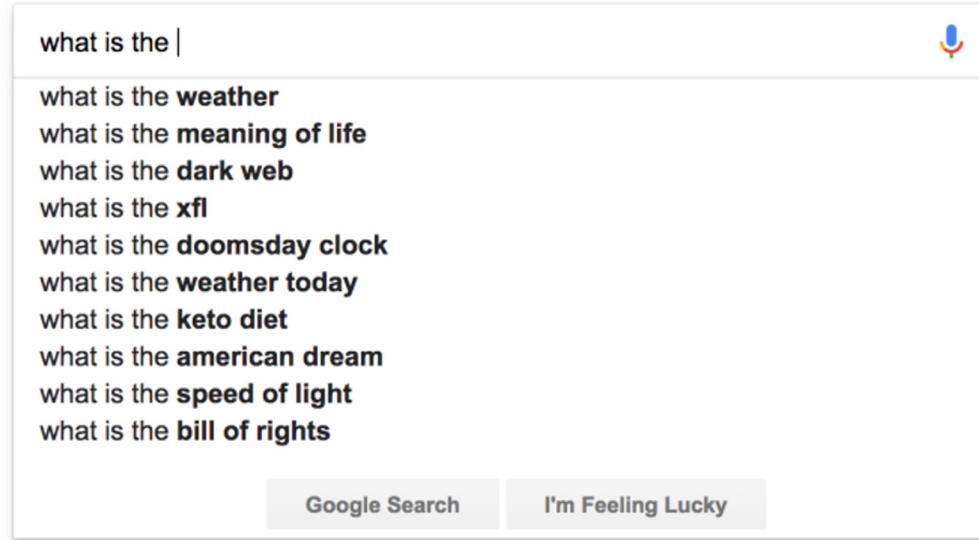
I like trying __

new recipes.
different art forms.
varied workouts.
challenging puzzles.
unique teas.



output

Language Models are everywhere



A screenshot of a Google search interface. The search bar at the top contains the text "what is the |". To the right of the search bar is a microphone icon for voice search. Below the search bar is a list of suggested search queries, each preceded by "what is the":

- weather**
- meaning of life
- dark web
- xfl
- doomsday clock
- weather today
- keto diet
- american dream
- speed of light
- bill of rights

At the bottom of the interface are two buttons: "Google Search" and "I'm Feeling Lucky".

Language Models are everywhere

Default (GPT-3.5)

MO

What is NLP?



NLP stands for Natural Language Processing. It is a field of artificial intelligence (AI) that focuses on the interaction between computers and human language. The primary goal of NLP is to enable computers to understand, interpret, and generate human language in a way that is both meaningful and useful.

NLP involves a wide range of tasks and applications, including:

1. **Text Understanding:** This involves tasks like sentiment analysis (determining the emotional tone of a piece of text), named entity recognition (identifying names of people, places, organizations, etc.), and text classification (categorizing text into predefined categories).
2. **Machine Translation:** NLP is used in machine translation systems like Google Translate to automatically translate text from one language to another.



How do large language models work: Tokenizing

Tokenizer, text to numbers: Large Language Models receive a text as input and generate a text as output.

GPT-3.5 & GPT-4 GPT-3 (Legacy)

I like trying

Clear Show example

Tokens	Characters
3	13

I like trying

[40, 1093, 4560]

How do large language models work: Predicting the next token

- Predicting the next token repeatedly:
 - Given n tokens as input (with max n varying from one model to another), the model is able to predict one token as output.
 - This token is then incorporated into the input of the next iteration, in an expanding window pattern, enabling a better user experience of getting one (or multiple) sentence as an answer.
- Selection process: The output token is chosen by the model according to its probability of occurring after the current text sequence.

I like trying _____
new

I like trying new _____
recipes

I like trying new recipes _____

[40, 1093, 4560, 502 19141, 13]

I like trying new _____
foods 0.6
hobbies 0.15
products 0.15
activities 0.1

How do large language models work

I like trying _____
new

I like trying new _____
recipes

I like trying new recipes _____
.

[40, 1093, 4560, 502 19141, 13]



Input -> Output

- (input, output) pairs
- We can train a very large model on a vast amount of data, such as hundreds of billions of words, to obtain a language model like ChatGPT.

A new way to find information

ChatGPT 3.5 ▾

You
Where is London?

ChatGPT
London is located in the southeastern part of England, United Kingdom. It's situated along the River Thames.

Message ChatGPT 

ChatGPT can make mistakes. Consider checking important information.

ChatGPT 3.5 ▾

You
Plan a trip for London

ChatGPT
Sure! Planning a trip to London can be exciting and there's so much to see and do. Here's a suggested itinerary for a 5-day trip to London:

Day 1: Arrival and Central London Exploration

- **Morning:** Arrive in London and check into your accommodation.
- **Afternoon:** Head to Buckingham Palace to witness the Changing of the Guard ceremony (if scheduled).
- **Evening:** Take a stroll along the South Bank of the Thames River, enjoy views of the London Eye and Big Ben, and have dinner at one of the many restaurants in the area.

Day 2: Historic London

- **Morning:** Start your day with a visit to the Tower of London, explore the Crown Jewels, and take a guided tour by the Yeomen Warders.

A new translator



ChatGPT 3.5 ▾



You

Translate the following from English to Dutch:

London is the capital and largest city of both England and the United Kingdom, with a population of around 8.8 million, and its metropolitan area is the largest in Western Europe, with a population of 14.9 million. It stands on the River Thames in south-east England at the head of a 50-mile (80 km) estuary down to the North Sea and has been a major settlement for nearly two millennia.



ChatGPT

Londen is de hoofdstad en grootste stad van zowel Engeland als het Verenigd Koninkrijk, met een bevolking van ongeveer 8,8 miljoen, en het grootstedelijk gebied is het grootste in West-Europa, met een bevolking van 14,9 miljoen. Het ligt aan de rivier de Theems in het zuidoosten van Engeland aan het hoofd van een estuarium van 50 mijl (80 km) tot aan de Noordzee en is al bijna twee millennia lang een belangrijke nederzetting.



Writing assistant

ChatGPT 3.5 ▾ 



You

Rewrite the following:

London is a city located in England. It is a big city.

 ChatGPT

London, situated in England, stands as a vast metropolis.



Message ChatGPT 

ChatGPT can make mistakes. Consider checking important information.

ChatGPT 3.5 ▾ 



You

Write a short story for children



ChatGPT

Title: The Magical Forest Adventure

Once upon a time, in a small village nestled between rolling hills and whispering streams, there lived a curious little girl named Lily. Lily had big, bright eyes that twinkled like stars, and a heart full of wonder. Every day, she would explore the woods that bordered her village, seeking new adventures and mysteries to uncover.

One sunny morning, as Lily wandered deeper into the forest, she stumbled upon a hidden path she had never seen before. Excited by the prospect of a new adventure, she skipped along the path, her feet kicking up tiny clouds of dust. The trees whispered secrets to her, and the birds sang melodies of ancient tales.

Search or Chat?

Google How to make cappuccino without machine X |   

Sign in

Videos Images Youtube Dolce gusto Foam TASSIMO Freddo News Books

All filters Tools SafeSearch

About 22,500,000 results (0.30 seconds)

Pams Daily Dish
<https://pamsdailydish.com/no-machine-3-step-home...> :

No Machine - 3 Step Homemade Cappuccino

1. Prepare coffee or espresso. ... 3. Pour prepared coffee into a large mug, add sugar.
Stir to combine. Add in frothy milk. That's it!



People also ask :

- Can you make a cappuccino without a coffee machine? 
- How to make a cappuccino at home? 
- Can you make a cappuccino with instant coffee? 
- How do you make coffee with milk without a coffee maker? 

Feedback

Videos :


[How to Make Perfect Cappuccino at Home Without Machine](#)
YouTube · Full Kitchen
16 Jan 2019

Cappuccino Without Ma

ChatGPT 3.5

You
How to make cappuccino without machine

ChatGPT
Making cappuccino without a machine is possible, though it won't be exactly the same as the one prepared with an espresso machine. Here's a simple recipe for making a homemade cappuccino using basic kitchen tools:

Ingredients:

- Strong coffee (brewed)
- Milk
- Sugar (optional)
- Cocoa powder or cinnamon for garnish (optional)

Equipment: 

Message ChatGPT... 

ChatGPT can make mistakes. Consider checking important information.

Applications - Writing

Sam Altman: Tech Influencer

ChatGPT 3.5

You Write a report about Sam Altman

ChatGPT As of my last knowledge update in January 2022, Sam Altman is a prominent figure in the technology and startup ecosystem. However, please note that my information might be outdated, and it's advisable to verify the latest details from reliable sources.

Sam Altman: A Profile

Background:

Sam Altman is an entrepreneur, investor, and influential figure in the tech industry. Born on April 22, 1985, in St. Louis, Missouri, Altman demonstrated an early aptitude for technology and entrepreneurship. He attended Stanford University but dropped out in 2005 to start his first company, Loopt, a location-based social networking app.

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Ilya Sutskever's AI Contributions

ChatGPT 3.5

You Write a report about Ilya Sutskever

ChatGPT As of my last knowledge update in January 2022, Ilya Sutskever is a prominent figure in the field of artificial intelligence and deep learning. Please note that there might have been developments or changes in his career or achievements since then. I recommend checking more recent sources for the latest information.

Ilya Sutskever is a computer scientist and a leading expert in the field of artificial intelligence. He is known for his significant contributions to deep learning, a subfield of machine learning that focuses on neural networks with multiple layers (deep neural networks). Sutskever co-founded OpenAI, an artificial intelligence research laboratory, along with Elon Musk, Sam Altman, Greg Brockman, Wojciech Zaremba, and John Schulman.

Here are some key points about Ilya Sutskever's background and contributions up to my last

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Applications - Translation

Sam Altman: Tech Influencer

ChatGPT 3.5

You 翻译: Sam Altman: A Profile

Background:
Sam Altman is an entrepreneur, investor, and influential figure in the tech industry. Born on April 22, 1985, in St. Louis, Missouri, Altman demonstrated an early aptitude for technology and entrepreneurship. He attended Stanford University but dropped out in 2005 to start his first company, Loopt, a location-based social networking app.

Loopt and Early Entrepreneurial Success:
Loopt gained attention as one of the early location-based social networking platforms. It allowed users to discover and share their location with friends. The company received funding and support, and in 2012, it was acquired by Green Dot Corporation.

Y Combinator:
Sam Altman's influence in the startup world became more pronounced when he joined Y Combinator (YC) in 2014 as its president. Y Combinator is one of the most renowned startup accelerators globally, providing mentorship, funding, and resources to early-stage startups. Under Altman's leadership, Y Combinator continued to play a pivotal role in shaping the trajectory of numerous successful startups.

OpenAI:
In 2019, Altman became the co-chairman of OpenAI, an artificial intelligence research lab focused on ensuring that AI benefits all of humanity. OpenAI has been at the forefront of AI research, and Altman's involvement signifies his interest in the ethical and responsible development of artificial intelligence.

ChatGPT

Sam Altman: 简介

背景:
Sam Altman是一位企业家、投资人，也是科技行业的重要人物。他出生于1985年4月22日，出生在密苏里州圣路易斯，早年展现出对技术和创业的天赋。他曾就读于斯坦福大学，但在2005年辍学，开始创办自己的第一家公司Loopt，这是一款基于地理位置的社交网络应用。

Loopt和早期的创业成功:
Loopt成为早期备受关注的基于地理位置的社交网络平台之一。它允许用户发现并与朋友分享他们的位置。该公司得到了资金和支持，2012年被Green Dot Corporation收购。

Y Combinator:
当Sam Altman于2014年加入Y Combinator (YC) 担任总裁时，他在创业界的影响力变得更为显著。Y Combinator是全球最著名的创业加速器之一，为早期初创企业提供导师指导、资金和资源。在Altman的领导下，Y Combinator继续在塑造众多成功初创企业的发展轨迹中发挥着关键作用。

OpenAI:

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Ilya Sutskever: AI Pioneer

ChatGPT 3.5

You 翻译: Ilya Sutskever is a computer scientist and a leading expert in the field of artificial intelligence. He is known for his significant contributions to deep learning, a subfield of machine learning that focuses on neural networks with multiple layers (deep neural networks). Sutskever co-founded OpenAI, an artificial intelligence research laboratory, along with Elon Musk, Sam Altman, Greg Brockman, Wojciech Zaremba, and John Schulman.

Here are some key points about Ilya Sutskever's background and contributions up to my last knowledge update:

Education:
Ilya Sutskever holds a Ph.D. in Machine Learning from the University of Toronto. His doctoral thesis, titled "Training Recurrent Neural Networks," reflects his early work and interest in the training of deep neural networks.

Research Contributions:
Sutskever has made significant contributions to the development of algorithms for training deep neural networks, particularly recurrent neural networks (RNNs) and generative models. His research has helped advance the capabilities of these models in tasks such as natural language processing and image generation.

OpenAI:
As one of the co-founders of OpenAI, Ilya Sutskever has been instrumental in the organization's mission to advance artificial intelligence in a safe and beneficial manner. OpenAI is known for its commitment to conducting cutting-edge research and sharing the results with the wider scientific community.

ChatGPT

伊利亚·苏茨克弗 (Ilya Sutskever) 是一位计算机科学家，也是人工智能领域的领军专家。他以在深度学习领域的显著贡献而闻名，深度学习是机器学习的一个子领域，专注于具有多层次结构的神经网络（深度神经网络）。苏茨克弗是OpenAI的联合创始人，这是一家人工智能研究实验室，与伊隆·马斯克、萨姆·奥特曼、格雷格·布罗克曼、沃伊切赫·扎伦巴和约翰·舒尔曼一起创立。

以下关于伊利亚·苏茨克弗背景和贡献的一些关键信息，截至我上次更新知识的时候：

教育:
伊利亚·苏茨克弗在多伦多大学获得了机器学习博士学位。他的博士论文题为“循环神经网络的训练”，反映了他早期在深度神经网络训练方面的工作和兴趣。

研究贡献:
苏茨克弗在深度神经网络的训练算法方面做出了重要贡献，特别是在循环神经网络 (RNNs) 和生成模型方面。他的研究帮助推动了这些模型在自然语言处理和图像生成等任务中的能力。

OpenAI:
作为OpenAI的联合创始人之一，伊利亚·苏茨克弗在推动该组织以安全和有益的方式推进人工智能

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Applications - Coding

OpenAI API Python Program

ChatGPT 3.5

You
Write a program for using the OpenAI API

ChatGPT
To use the OpenAI API, you need to follow these general steps:

- Obtain API key: Sign up for access to the OpenAI API and obtain your API key.
- Install the OpenAI Python library: You can install the library using pip:

```
bash
pip install openai
```
- Use the OpenAI API: You can use the OpenAI API by making HTTP requests with the `openai` Python library.
Here's a simple example using Python:

```
python
import openai

# Set your OpenAI API key
api_key = "your-api-key"
openai.api_key = api_key

# Define the prompt for the model
prompt = "Translate the following English text to French:"

# Call the OpenAI API to generate text based on the prompt
response = openai.Completion.create(
    engine="text-davinci-003", # Specify the engine
    prompt=prompt,
    max_tokens=100 # Adjust the maximum number of tokens in the generated text
)

# Print the generated text
generated_text = response.choices[0].text.strip()
print(generated_text)
```

Replace `“your-api-key”` with your actual OpenAI API key. You may also need to adjust the prompt and other parameters based on your specific use case.

Make sure you comply with OpenAI’s use-case policies and guidelines when using the API. You can find the most up-to-date information in the ↓ API documentation:
<https://beta.openai.com/docs/>

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Corrected OpenAI API Code

ChatGPT 3.5

You
Debug the following Python code:

```
import openai

# Set your OpenAI API key
api_key = "your-api-key"
openai.api_key = api_key

# Define the prompt for the model
prompt = "Translate the following English text to French:"

# Call the OpenAI API to generate text based on the prompt
response = openai.Completion.create(
    engine="text-davinci-003", # Specify the engine
    prompt=prompt,
    max_tokens=100 # Adjust the maximum number of tokens in the generated text
)

# Print the generated text
generated_text = response.choices[0].text.strip()
print(generated_text)
```

ChatGPT
It seems there's a small mistake in your code. The correct way to access the generated text from the OpenAI API response is using `response.choices[0].text`, not `response.choices[1].text`. Here's the corrected code:

```
python
import openai

# Set your OpenAI API key
api_key = "your-api-key"
openai.api_key = api_key

# Define the prompt for the model
prompt = "Translate the following English text to French:"

# Call the OpenAI API to generate text based on the prompt
response = openai.Completion.create(
    engine="text-davinci-003", # Specify the engine
    prompt=prompt,
    max_tokens=100 # Adjust the maximum number of tokens in the generated text
)
```

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Applications - Chatting

Car Buying Assistance

ChatGPT 3.5

You
You are a car seller tasked with selling customers a car.

ChatGPT
Certainly! I'd be happy to help you find the perfect car. To get started, could you please provide me with some information about your preferences and requirements? For example:

1. What type of vehicle are you looking for? (Sedan, SUV, truck, etc.)
2. Do you have a specific make or model in mind?
3. What features are important to you? (Fuel efficiency, safety features, technology, etc.)
4. What is your budget range?
5. Do you have any specific color preferences?

Once I have this information, I can recommend some options that meet your criteria and provide you with more details about each car.

You
Looking for SUVs

ChatGPT
Great! SUVs are versatile vehicles that offer a good balance of space, comfort, and often have additional features. To help narrow down the options, could you provide some more details?

1. **Budget Range:** What is your budget for the SUV?
2. **Size Preferences:** Are you looking for a compact SUV, midsize SUV, or a full-size SUV?
3. **Usage:** How do you plan to use the SUV? (e.g., daily commute, family trips, off-road adventures)
4. **Features:** Are there specific features that are important to you? (e.g., safety features, technology, entertainment system, towing capacity)
5. **Fuel Efficiency:** Is fuel efficiency an important factor for you?
6. **Preferred Brands:** Do you have any preferences for certain SUV brands?

Once I have this information, I can provide you with some tailored recommendations that suit your needs and preferences.

You
Could you suggest some brands

ChatGPT
Certainly! There are several reputable SUV brands that offer a variety of models with different features and capabilities. Here are some popular SUV brands:

Message ChatGPT...

ChatGPT can make mistakes. Consider checking important information.

Generative Pre-trained Transformers (GPT)

Generative Pre-trained Transformers (GPT)

Decoder-based transformers

The first GPT model, introduced in 2018 by OpenAI, was just the decoder part of the original transformer.

Input: What is NLP? →



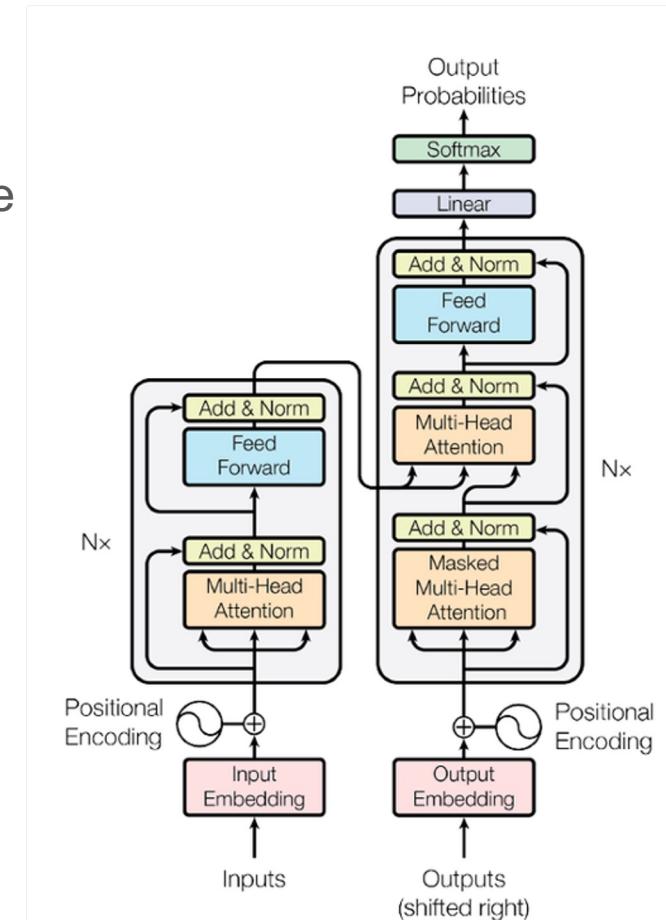
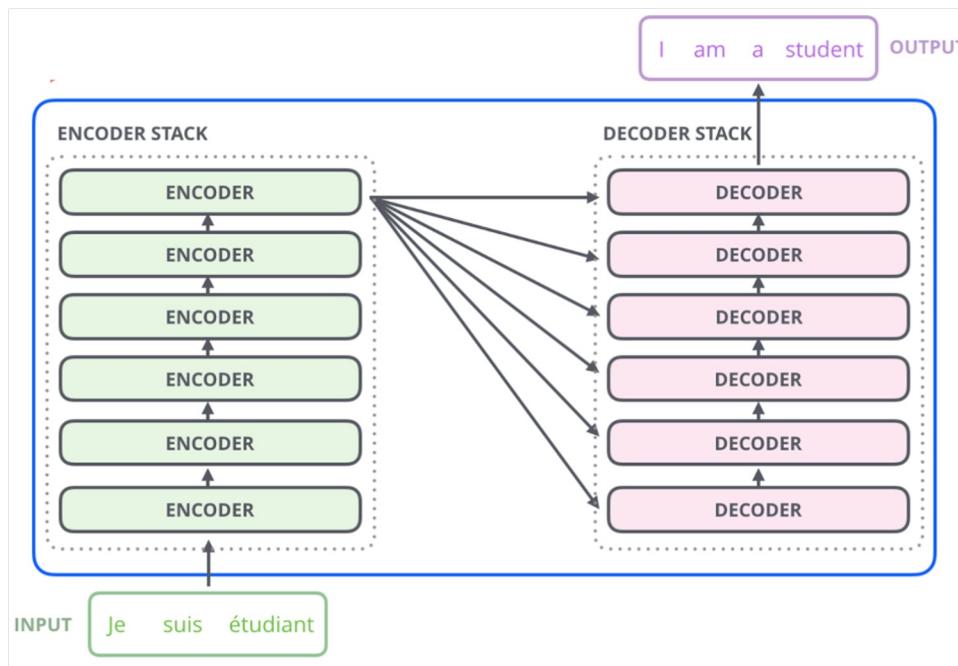
Output:
NLP stands for Natural Language Processing, which is a subfield of artificial intelligence (AI) that focuses on the interaction between computers and

.....

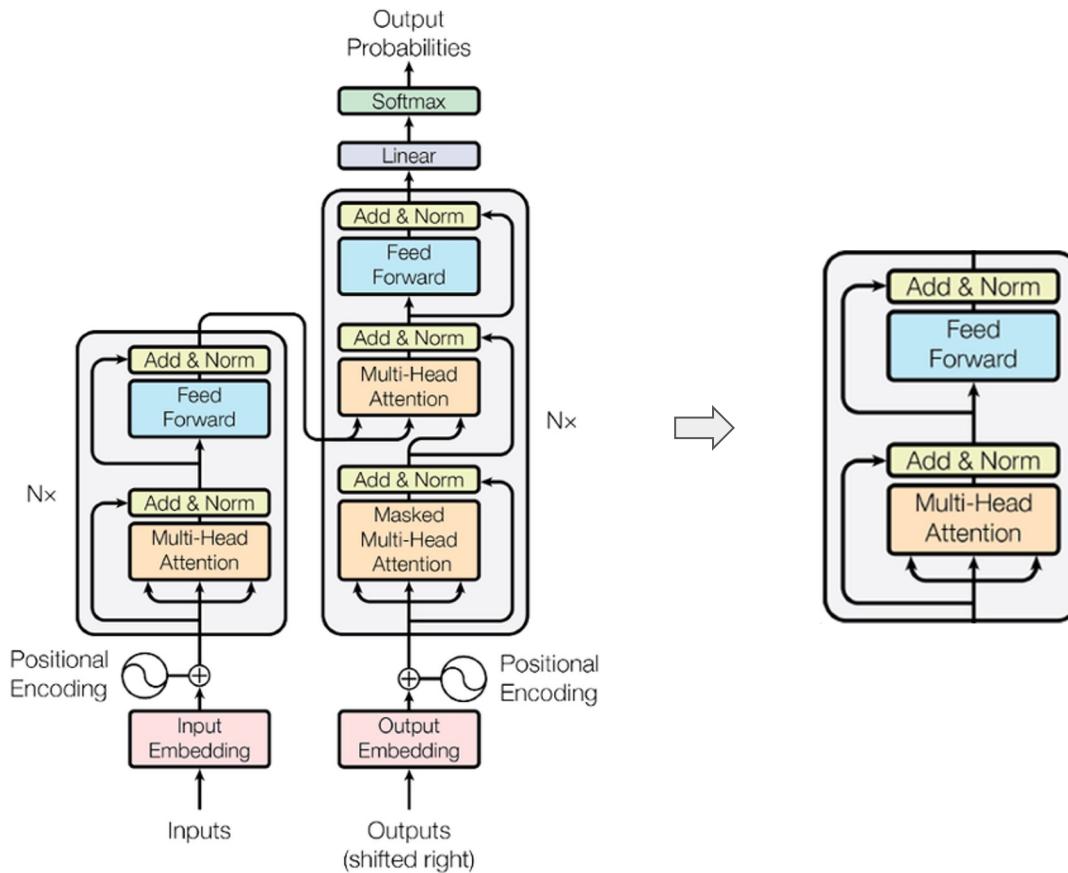
From Internet

Transformers

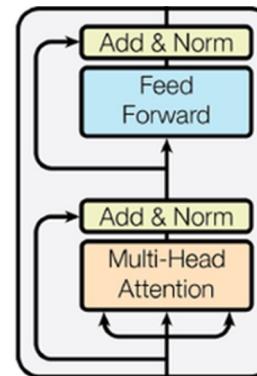
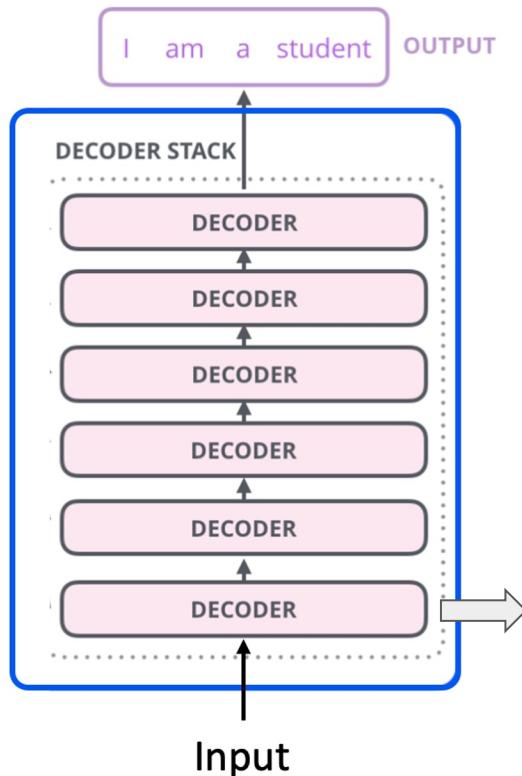
Standard transformers: encoder-decoder architecture



Transformer block

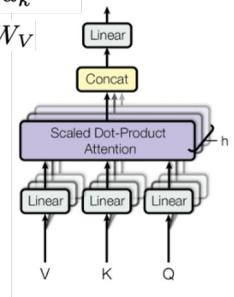


Decoder-based transformers



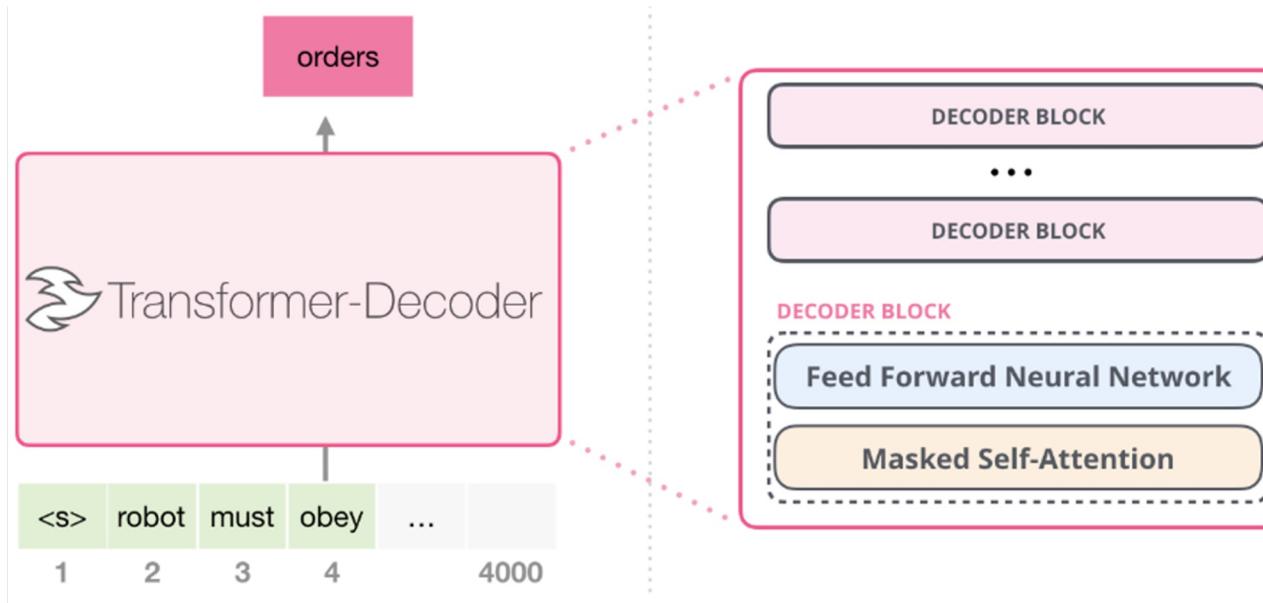
$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$
$$Q = XW_Q, \quad K = XW_K, \quad V = XW_V$$



GPT Architecture

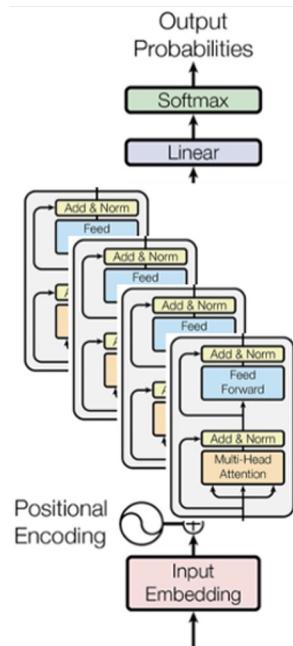
A stack of decoders (decoder blocks)



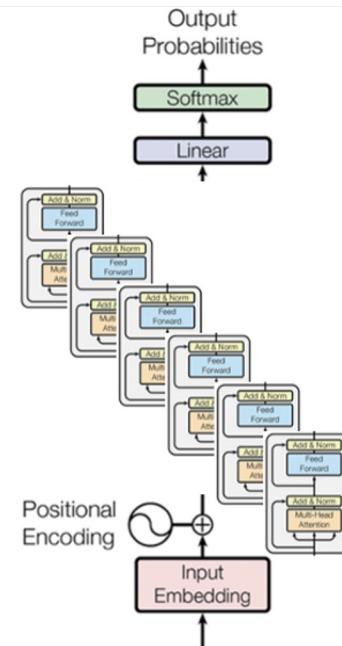
GPT models from OpenAI

GPT-2/-3/-4 have mostly just been larger versions. With the key differences coming from training data and training processes.

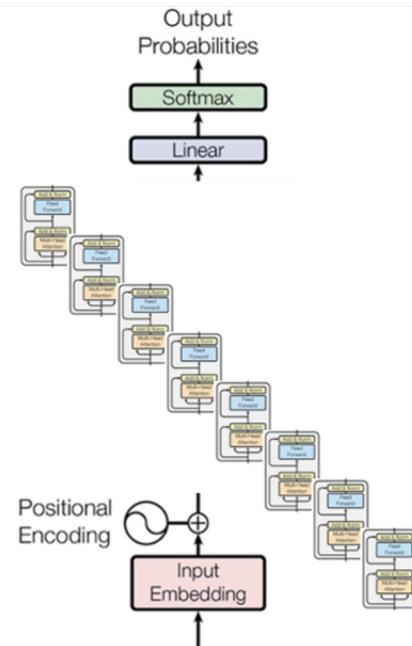
- GPT (2018)
- GPT-2 (2019)
- GPT-3 (2020)



GPT/GPT-1
12x
BooksCorpus



GPT-2
48x
WebText



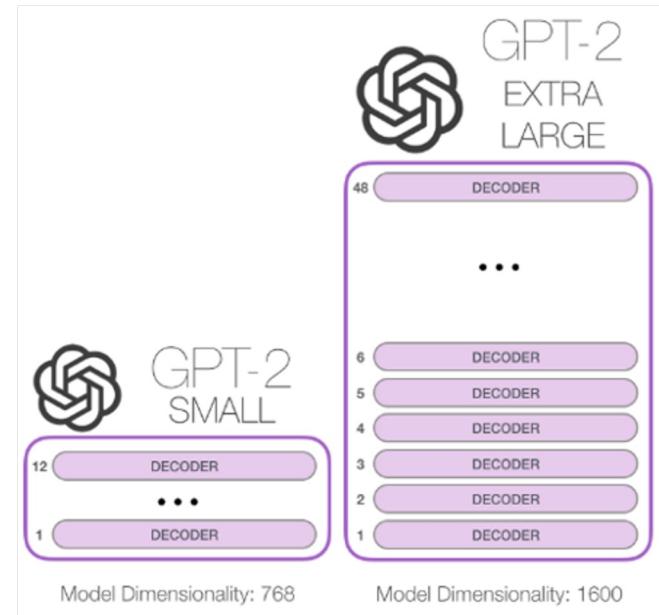
GPT-3
96x
WebText 2

GPT models have many parameters

Model size refers to the number of parameters in the model.

The larger the model, the better it will be in understanding the nuances of human language.

Parameter	GPT-2 Small	GPT-2 Extra Large	Difference Extra Large / Small
Layers	12	48	3x
Model Dimensionality	768	1600	2x
Attention Heads per Layer	12	25	2.5x
Total	117 Million	1.5 Billion	12.8x



Prompt Engineering

Meng Fang
OxML 2024



What is Prompt Engineering?

Prompt Engineering is the process of designing and optimizing text inputs (prompts) to deliver consistent and high-quality responses (completions) for a specific application objective and model.

prompt / input



I like trying __

- new recipes.
- different art forms.
- varied workouts.
- challenging puzzles.
- unique teas.



output

Prompt Engineering

Improve the intuition for it is to practice more and adopt a trial-and-error approach that combines application domain expertise with recommended techniques and model-specific optimizations.

To execute the exercises you will need:

- An OpenAI API key - the service endpoint for a deployed LLM.
- A Python Runtime - in which the Notebook can be executed.

OPENAI_API_KEY=st-xxxxxxxx

Prompt Engineering

We can think of this as a 2-step process:

- designing the initial prompt for a given model and objective
- refining the prompt iteratively to improve the quality of the response -> Trial-and-error process

Prompt Engineering

We first need to understand three concepts:

- Tokenization = how the model "sees" the prompt
- Base LLMs = how the foundation model "processes" a prompt
- Instruction-Tuned LLMs = how the model can now see "tasks"

Tokenization

<https://platform.openai.com/tokenizer>

GPT-3.5 & GPT-4 GPT-3 (Legacy)

OpenAI's large language models (sometimes referred to as GPT's) process text using tokens, which are common sequences of characters found in a set of text. The models learn to understand the statistical relationships between these tokens, and excel at producing the next token in a sequence of tokens.

Clear Show example

Tokens	Characters
58	301

OpenAI's large language models (sometimes referred to as GPT's) process text using tokens, which are common sequences of characters found in a set of text. The models learn to understand the statistical relationships between these tokens, and excel at producing the next token in a sequence of tokens.

TEXT TOKEN IDS

Tokens	Characters
58	301

[5109, 15836, 596, 3544, 4221, 4211, 320, 57753, 14183, 311, 439, 480, 2898, 596, 8, 1920, 1495, 1701, 11460, 11, 902, 527, 4279, 24630, 315, 5885, 1766, 304, 264, 743, 315, 1495, 13, 578, 4211, 4048, 311, 3619, 279, 29564, 12135, 1990, 1521, 11460, 11, 323, 25555, 520, 17843, 279, 1828, 4037, 304, 264, 8668, 315, 11460, 13]

TEXT TOKEN IDS

LLMs - Foundation Models

Once a prompt is tokenized, the primary function of the "Base LLM" (or Foundation model) is to predict the token in that sequence.

- LLMs are trained on massive text datasets.
- They can continue predicting the sequence till terminated by user intervention or some pre-established condition.

We can chat. But what if the user wanted to see something specific that met some criteria or task objective?

Instruction Tuned LLMs

An Instruction Tuned LLM starts with the foundation model and fine-tunes it with examples or input/output pairs that can contain **clear instructions** - and the response from the AI attempt to follow that instruction.

This uses techniques like Reinforcement Learning with Human Feedback (RLHF) that can train the model to follow instructions and learn from feedback so that it produces responses that are better-suited to practical applications and more-relevant to user objectives.

Instruction examples

Prompt:

Classify the text into neutral, negative or positive.

Text: I like walking.

Sentiment:

Output:

Positive

Instruction examples

Prompt:

Summarize the text delimited by triple backticks into a single sentence.

```
```You should express what you want a model to do by providing instructions that are as clear and specific as you can possibly make them. This will guide the model towards the desired output, and reduce the chances of receiving irrelevant or incorrect responses. Don't confuse writing a clear prompt with writing a short prompt. In many cases, longer prompts provide more clarity and context for the model, which can lead to more detailed and relevant outputs.
```

...

Output:

Clear and specific instructions are crucial when guiding a model's behavior, emphasizing the importance of expressing desired outcomes to reduce the likelihood of irrelevant or incorrect responses; the length of the prompt should not be sacrificed for clarity and context, as longer prompts often lead to more detailed and relevant model outputs.

# What LLMs can and can not do

Knowledge cutoffs: An LLM's knowledge of the world is frozen at the time of its training.

Hallucinations

The input (and output) length is limited.

Bias and Toxicity: An LLM can reflect the biases that exist in the text it learned.

# Advanced Technologies

Zero-shot Prompting

Few-shot Prompting

Chain-of-Thought Prompting

Self-Consistency

# Zero-Shot Prompting

Prompt:

Classify the sentiment of the following statement: "{text}"

Options:

1. Very Positive
2. Positive
3. Neutral
4. Negative
5. Very Negative

\*Replace {text} with the actual text you want to classify for sentiment.

=

Positive Walking Pref



ChatGPT 3.5 ▾



MO You

Classify the sentiment of the following statement: "I like walking"

Options:

1. Very Positive
2. Positive
3. Neutral
4. Negative
5. Very Negative



ChatGPT

The sentiment of the statement "I like walking" is Positive.



Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.

# Code example: Zero-Shot Prompting

```
import openai

def classify_sentiment(api_key, text):
 openai.api_key = api_key

 # Generate a prompt that asks the model to classify the sentiment of the text
 prompt = f"Classify the sentiment of the following statement: \'{text}\'\n\nOptions:\n1. Very Positive\n2. Positive\n3. Neutral\n4. Negative\n5. Very Negative"

 # Use the ChatGPT model to generate a response
 try:
 response = openai.Completion.create(
 engine="text-davinci-003", # Choose the appropriate model version
 prompt=prompt,
 max_tokens=60,
 temperature=0 # Use a lower temperature for more deterministic outputs
)
 return response.choices[0].text.strip()
 except Exception as e:
 return str(e)

Replace 'your_api_key_here' with your actual OpenAI API key
api_key = 'your_api_key_here'
text = "This new smartphone model delivers an exceptional user experience."

result = classify_sentiment(api_key, text)
print("Sentiment Classification:", result)
```

# Few-Shot Prompting

1-shot

Classify the sentiment of the following sentences as 'Positive', 'Negative', or 'Neutral'.

3-shot

Sentence: "I love sunny days."  
Sentiment: Positive

Sentence: "Their performance exceeded all my expectations!"  
Sentiment:

Classify the sentiment of the following sentences as 'Positive', 'Negative', or 'Neutral'.

Sentence: "I love sunny days."  
Sentiment: Positive

Sentence: "This soup tastes terrible."  
Sentiment: Negative

Sentence: "He is always so kind and helpful."  
Sentiment: Positive

Sentence: "Their performance exceeded all my expectations!"  
Sentiment:



ChatGPT 3.5 ▾

# Few-Shot Prompting

5-shot

Classify the sentiment of the following sentences as 'Positive', 'Negative', or 'Neutral'.

Sentence: "I love sunny days."  
Sentiment: Positive

Sentence: "This soup tastes terrible."  
Sentiment: Negative

Sentence: "He is always so kind and helpful."  
Sentiment: Positive

Sentence: "I'm feeling very sick today."  
Sentiment: Negative

Sentence: "The movie was okay, nothing special."  
Sentiment: Neutral

Sentence: "Their performance exceeded all my expectations!"  
Sentiment:

You

Classify the sentiment of the following sentences as 'Positive', 'Negative', or 'Neutral'.

Sentence: "I love sunny days."  
Sentiment: Positive

Sentence: "This soup tastes terrible."  
Sentiment: Negative

Sentence: "He is always so kind and helpful."  
Sentiment: Positive

Sentence: "I'm feeling very sick today."  
Sentiment: Negative

Sentence: "The movie was okay, nothing special."  
Sentiment: Neutral

Sentence: "Their performance exceeded all my expectations!"  
Sentiment:



ChatGPT

Sentence: "Their performance exceeded all my expectations!"  
Sentiment: Positive



Message ChatGPT...





# Some Tips for Few-Shot Prompting

- The label space and the distribution of the input text specified by the demonstrations are both important (regardless of whether the labels are correct for individual inputs).
- The format you use also plays a key role in performance, even if you just use random labels, this is much better than no labels at all.
- Additional results show that selecting random labels from a true distribution of labels (instead of a uniform distribution) also helps.

\*random labels

ChatGPT 3.5 ▾



You

Classify the sentiment of the following sentences as 'Positive', 'Negative', or 'Neutral'.

Sentence: "I love sunny days."

Sentiment: Positive

Sentence: "This soup tastes terrible."

Sentiment: Negative

Sentence: "He is always so kind and helpful."

Sentiment: Negative

Sentence: "I'm feeling very sick today."

Sentiment: Positive

Sentence: "The movie was okay, nothing special."

Sentiment: Neutral

Sentence: "Their performance exceeded all my expectations!"

Sentiment:



ChatGPT

Sentiment: Positive



Is this conversation helpful so far?

Message ChatGPT...



# Chain-of-Thought Prompting

## Standard Prompting

### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

### Model Output

A: The answer is 27. X

## Chain-of-Thought Prompting

### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls.  $5 + 6 = 11$ . The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

### Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had  $23 - 20 = 3$ . They bought 6 more apples, so they have  $3 + 6 = 9$ . The answer is 9. ✓

# Chain-of-Thought Prompting

Prompt:

The odd numbers in this group add up to an even number: 4, 8, 9, 15, 12, 2, 1.

A: Adding all the odd numbers (9, 15, 1) gives 25. The answer is False.

The odd numbers in this group add up to an even number: 17, 10, 19, 4, 8, 12, 24.

A: Adding all the odd numbers (17, 19) gives 36. The answer is True.

The odd numbers in this group add up to an even number: 16, 11, 14, 4, 8, 13, 24.

A: Adding all the odd numbers (11, 13) gives 24. The answer is True.

The odd numbers in this group add up to an even number: 17, 9, 10, 12, 13, 4, 2.

A: Adding all the odd numbers (17, 9, 13) gives 39. The answer is False.

The odd numbers in this group add up to an even number: 15, 32, 5, 13, 82, 7, 1.

A:

Output

Adding all the odd numbers (15, 5, 13, 7, 1) gives 41. The answer is False.

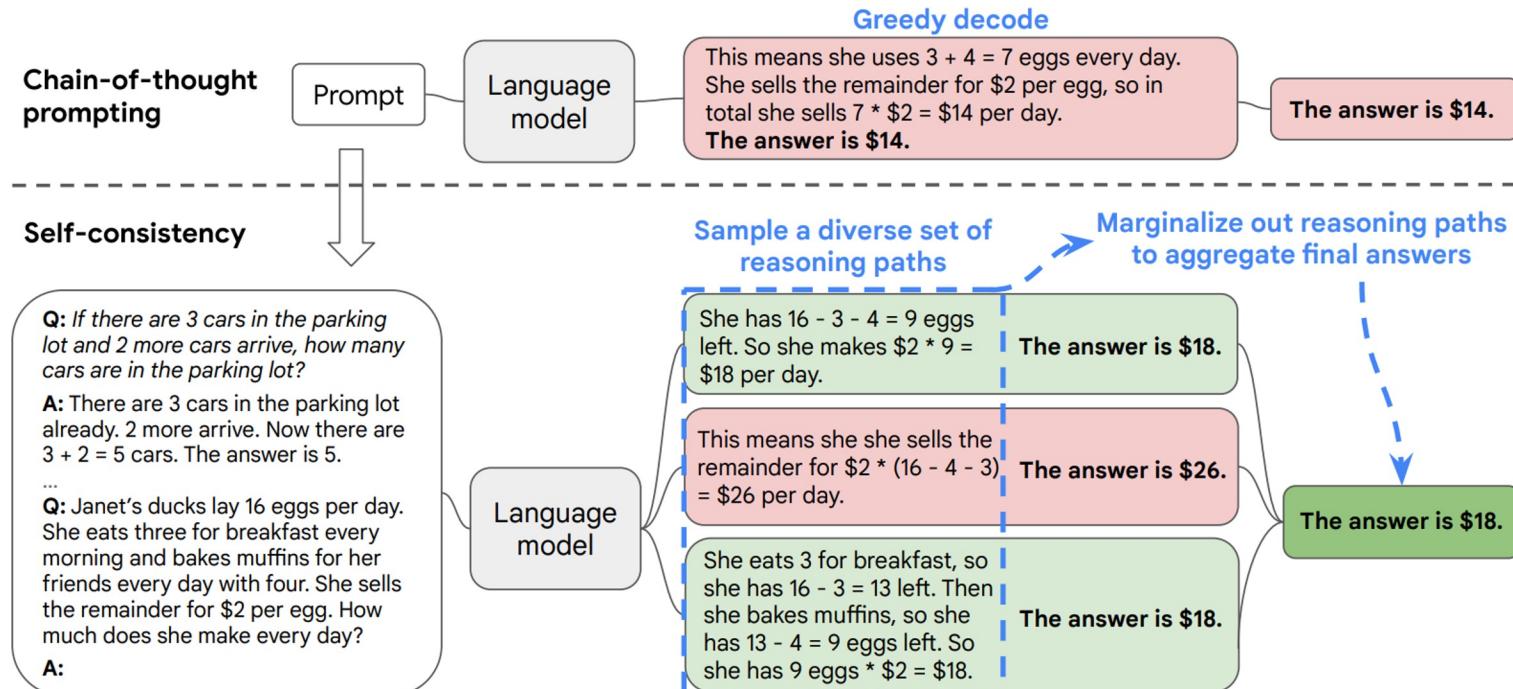
# Self-Consistency

Self-consistency aims to replace the naive greedy decoding used in chain-of-thought prompting.

The idea is to sample multiple, diverse reasoning paths through few-shot CoT, and use the generations to select the most consistent answer.

This helps to boost the performance of CoT prompting on tasks involving arithmetic and commonsense reasoning.

# Chain-of-thought vs. Self-Consistency



# Expanding Knowledge for LLMs

Meng Fang  
OxML 2024



# Retrieval-Augmented Generation

Generation: Large Language Models (LLMs), such as ChatGPT/GPT-3.5/4 and others

LLMs can generate texts in response to a user input, referred to as a prompt.

Input/Prompt



LLM

U

Who won the world cup 2018?



The 2018 FIFA World Cup was won by the French national football (soccer) team. France defeated Croatia 4-2 in the final match to win their second FIFA World Cup. The tournament took place in Russia from June 14 to July 15, 2018.

U

Who won the world cup 2022?

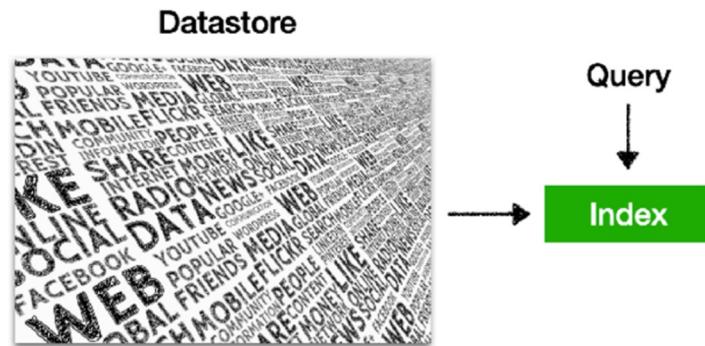


I'm sorry, but as of my last knowledge update in September 2021, I do not have information on the winner of the 2022 FIFA World Cup, as it had not yet taken place. The 2022 FIFA World Cup was scheduled to be held in Qatar from November 21 to December 18, 2022. To find out the winner of the tournament, I recommend checking the latest news or conducting an online search for up-to-date information.

\*A use case in 2021

# Retrieval

## Retrieval from a datastore or a collection of documents

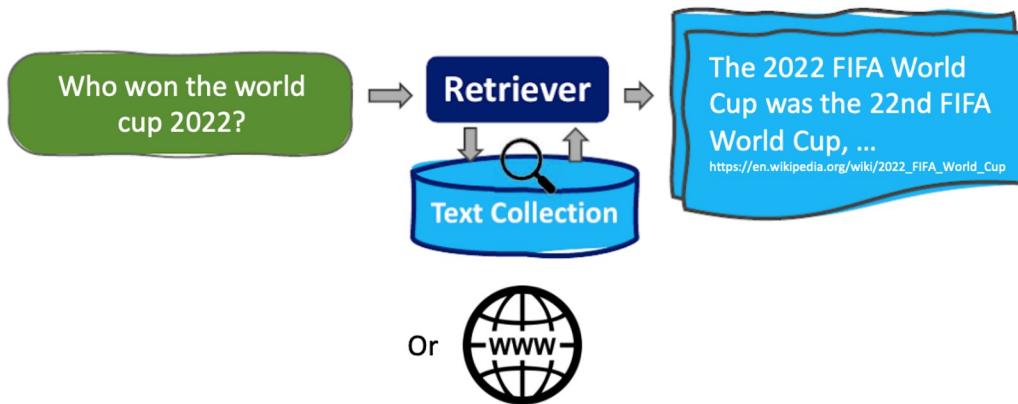


Internet,  
or a collection of documents

- Representative tasks: open-domain QA, fact checking, entity linking, ...

# Retrieval

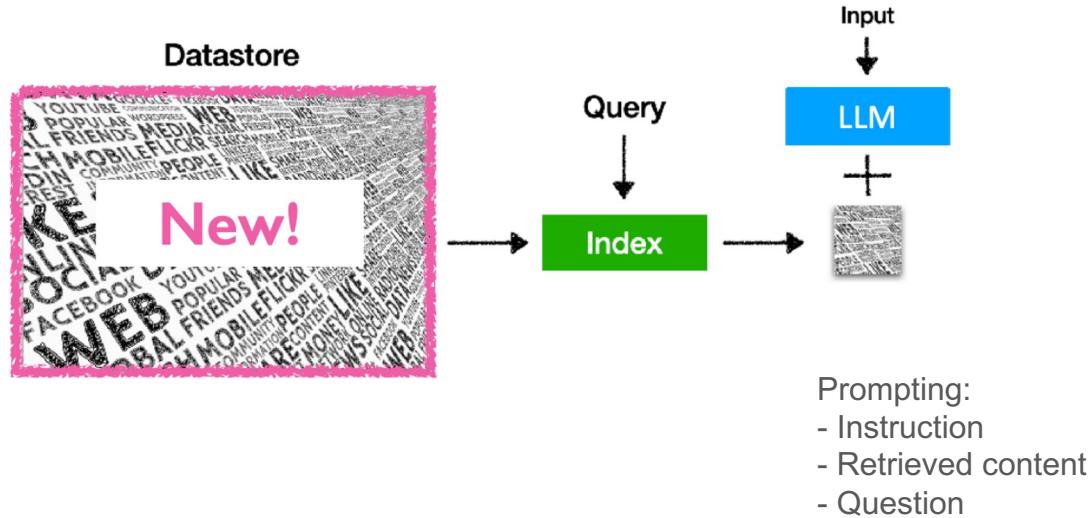
Retrieval from a datastore or a collection of documents



- Representative tasks: open-domain QA, fact checking, entity linking, ..

# Retrieval-Augmented Generation

## Retrieval + LLM



# Retrieval-Augmented Generation

## Retrieval + LLM

 Who won the world cup 2022?  


  Finished browsing

Argentina won the 2022 FIFA World Cup, defeating France in the final. This victory marked Argentina's third World Cup title. The final match was especially notable as Lionel Messi scored twice, leading Argentina to win on penalties after a dramatic game [1](#) [2](#) [3](#) [4](#) [5](#).

# Why retrieval → LLMs?

LLMs can't memorize all (long-tail) knowledge in their parameters

LLMs' knowledge is easily outdated and hard to update

LLMs' output is challenging to interpret and verify

LLMs are shown to easily leak private training data

LLMs are \*large\* and expensive to train and run

# LLMs can't memorize all knowledge in their parameters



List 5 important papers authored by Geoffrey Hinton



Geoffrey Hinton is a renowned computer scientist ...  
Here are five important papers authored by him:

1. **"Learning Internal Representations by Error Propagation"** (with D. E. Rumelhart and R. J. Williams) - This paper, published in **1986**, .. 
2. **"Deep Boltzmann Machines"** (with R. Salakhutdinov) - Published in **2009**, .. 
3. **"Deep Learning"** (with Y. Bengio and A. Courville) - Published as a book in **2016**, ... 
4. **"Attention Is All You Need"** (with V. Vaswani, N. Shazeer, et al.) - Published in **2017**, this paper introduced the Transformer model, ... 



Database: <https://dblp.org/>

[+] Geoffrey E. Hinton [-] ↗ ↘ 🔍

> Home > Persons

[...] Person information

- affiliation: Google DeepMind, London, UK
- affiliation: University of Toronto, Department of Computer Science, ON, Canada
- award (2018): Turing Award
- award (2016): BBVA Foundation Frontiers of Knowledge Award

[...] 2020 - today ⓘ

2023

- [j70] ⓘ ⓘ ⓘ ⓘ ⓘ Geoffrey E. Hinton:  
How to Represent Part-Whole Hierarchies in a Neural Network. *Neural Comput.* 35(3): 413-452 (2023)
- [c162] ⓘ ⓘ ⓘ ⓘ ⓘ Ting Chen, Ruixiang Zhang, Geoffrey E. Hinton:  
Analog Bits: Generating Discrete Data using Diffusion Models with Self-Conditioning. *ICLR* 2023
- [c161] ⓘ ⓘ ⓘ ⓘ ⓘ Mengye Ren, Simon Kornblith, Renjie Liao, Geoffrey E. Hinton:  
Scaling Forward Gradient With Local Losses. *ICLR* 2023

2022

- [c160] ⓘ ⓘ ⓘ ⓘ ⓘ Kevin Clark, Kelvin Guu, Ming-Wei Chang, Panupong Pasupat, Geoffrey E. Hinton, Mohammad Norouzi:  
Meta-Learning Fast Weight Language Models. *EMNLP* 2022: 9751-9757
- [c159] ⓘ ⓘ ⓘ ⓘ ⓘ Ting Chen, Saurabh Saxena, Lala Li, David J. Fleet, Geoffrey E. Hinton:  
Pix2seq: A Language Modeling Framework for Object Detection. *ICLR* 2022
- [c158] ⓘ ⓘ ⓘ ⓘ ⓘ Ting Chen, Saurabh Saxena, Lala Li, Tsung-Yi Lin, David J. Fleet, Geoffrey E. Hinton:  
A Unified Sequence Interface for Vision Tasks. *NeurIPS* 2022

# LLMs' knowledge is easily outdated and hard to update



Who is the CEO of Twitter?



ChatGPT

As of my **knowledge cutoff in September 2021**, the CEO of Twitter is **Jack Dorsey**....

Google search results for "Who is the CEO of Twitter?". The search bar shows the query. Below it are various filters: 2023, Images, News, Right now, Latest, Former, Videos, First, Books. The main result is about Linda Yaccarino, with a snippet: "About 1,140,000,000 results (0.50 seconds)". It shows her title as "Twitter / CEO" and her name "Linda Yaccarino". A small image of Linda Yaccarino is shown. Below this, a news snippet from the New York Times discusses her hiring by Elon Musk.

Who is the CEO of Twitter?

2023 Images News Right now Latest Former Videos First Books

About 1,140,000,000 results (0.50 seconds)

Twitter / CEO

Linda Yaccarino

5 Jun 2023–

When Elon Musk announced last month that he had hired Linda Yaccarino as Twitter's chief executive, he said he was “excited” to bring on someone who could “focus primarily on business operations.”

29 Jun 2023

The datastore can be easily updated and expanded - even without retraining!

# LLMs' output is challenging to interpret and verify

Provide references

The screenshot shows the Microsoft Bing Chat interface. At the top, there are 'SEARCH' and 'CHAT' buttons, with 'CHAT' being active. A blue button at the top right says 'create a plan for exploring London for one day'. Below this, two green checkmarks indicate the process: 'Searching for: London itinerary 1 day' and 'Generating answers for you...'. A row of social media sharing icons follows. The main content area starts with a message: 'Sure, here's a one-day itinerary for exploring London:'. Under the heading 'Day 1', there is a bulleted list of five attractions: Buckingham Palace, Westminster Abbey, Big Ben and Houses of Parliament, London Eye, and Tower Bridge. The text concludes with 'I hope this helps! Let me know if you have any other questions or if there is anything else I can help you with.' At the bottom, there is a 'Learn more:' section with five links: 1. londondreaming.com, 2. onedayitinerary.com, 3. onestep4ward.com, 4. visitlondon.com, 5. fullsuitcase.com, and a '— see less' link.

✓ Searching for: London itinerary 1 day

✓ Generating answers for you...

Sure, here's a one-day itinerary for exploring London:

**Day 1**

- **Buckingham Palace:** Start your day at Buckingham Palace to witness the Changing of the Guard ceremony.
- **Westminster Abbey:** Next, visit Westminster Abbey, a UNESCO World Heritage Site and one of the most famous religious buildings in the world.
- **Big Ben and Houses of Parliament:** From Westminster Abbey, walk to Big Ben and Houses of Parliament. You can take a tour of the Houses of Parliament if you book in advance.
- **London Eye:** After Big Ben, take a ride on the London Eye to get a bird's eye view of London.
- **Tower Bridge:** From the London Eye, walk to Tower Bridge and take in the views of the River Thames.

I hope this helps! Let me know if you have any other questions or if there is anything else I can help you with.

Learn more:

- 1. londondreaming.com
- 2. onedayitinerary.com
- 3. onestep4ward.com
- 4. visitlondon.com
- 5. fullsuitcase.com

— see less

LLMs are shown to easily leak private training data

GA Can you write the first page of "True History of the Kelly Gang"

Certainly! Here is the first page of Peter Carey's "True History of the Kelly Gang":

I, NED KELLY, of the Fenian Brotherhood, am the adopted son of a man called Red Kelly and I am the state's most wanted bushranger. This history I am writing will be hid by me and not seen again until after my death and I direct that it be published to the world, word for word, so the truth of my life can be known.

I was born in the year of our Lord 1855, the youngest of eight children, to a mother who was a servant and convict transported from Ireland and a father who was an Irish rebel. I was raised in poverty and ignorance in the colony of Victoria. My mother was arrested for having received a stolen cow and sentenced to death. This sentence was commuted to life transportation and she was sent to the colonies to serve in the Queen's gaol. My father was a wild colonial and in his blood there was a fighting fire. He was shot dead by the police when I was twelve years old and I can still see him falling.

## Private datastore / data access control



Danger! Copyrighted text.

**LLMs are \*large\* and expensive to train and run**

Can we train LLMs every week?

Can we possibly reduce the training and inference costs, and scale down the size of LLMs?

- Use datastore and retrieval

# Fine-tuning

To carry out a task that isn't easy to define in a prompt.

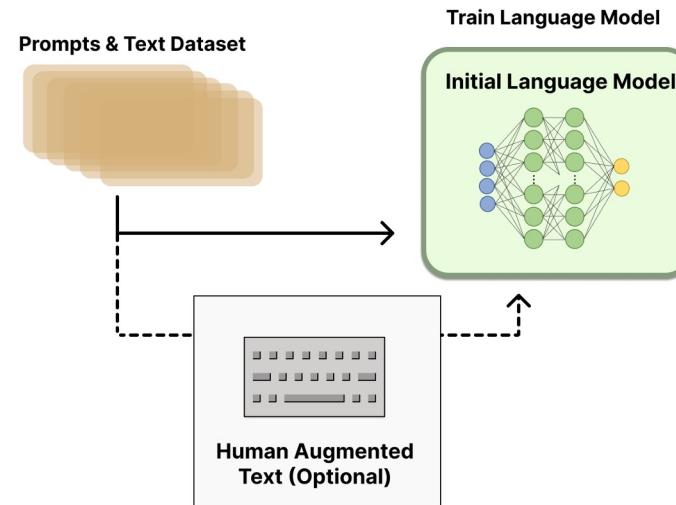
To help LLM gain specific knowledge.

To get a smaller model to perform a task.

# Pretraining an LLM

Pretrain general-purpose LLMs by learning from internet text.

For building a specific application: option of last resort; could help if have a highly specialized domain



# Choosing a model

Model size

1B parameters: GPT-Neo (1.3B), Pythia-1B, OPT-1.3B

10B parameters: LLama-13B, Mistra-13B, ...

100B parameter: GPT-3.5, GPT-4

# Closed or open source?

## Closed-source models

- Easy to use in applications
- More large/powerful models
- Relatively inexpensive
- Some risk of vendor lock-in
- Example: GPT-4

## Open-source models

- Full control over model
- Can run on your own device
- Full control over data  
privacy/access
- Example: Llama-2/3

# How do LLMs learn to follow instructions?

## Fine tuning

Help me brainstorm some fun museums in London.

Here are some suggestions ...

## Input -> Output

Help me brainstorm some fun museums in London. -> **Here**

Help me brainstorm some fun museums in London. Here -> **are some suggestions**

Help me brainstorm some fun museums in London. Here are -> **some suggestions**

Help me brainstorm some fun museums in London. Here are some -> **suggestions**

... -> ...

# Reinforcement learning from human feedback

Step 1: Train a reward (quality) model

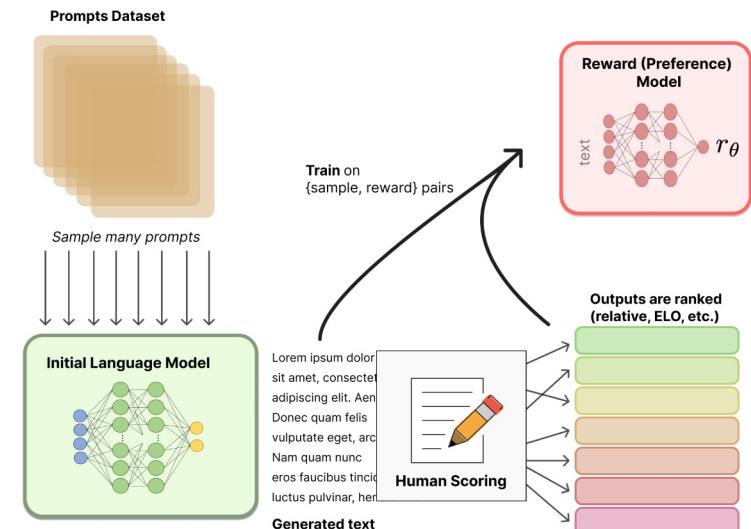
Input -> Output

(Response) | (Score/Reward)

I'm happy to help. Here are some steps ... | 5

Just try your best! | 3

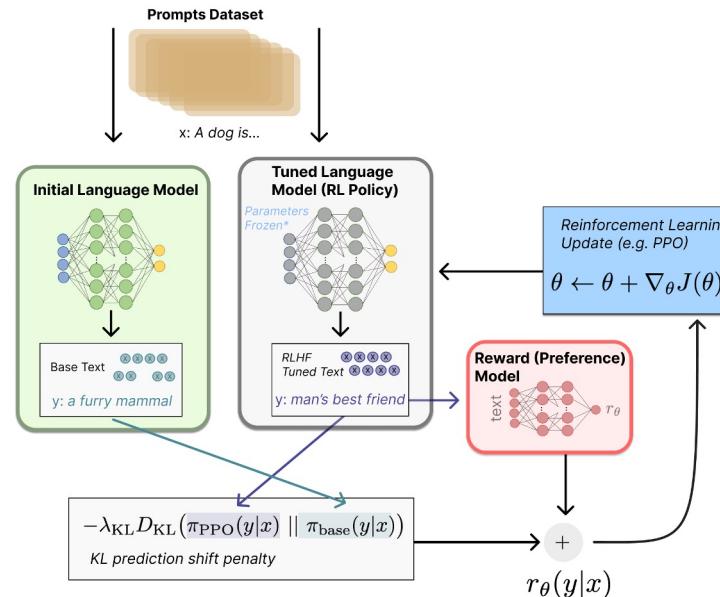
It is hopeless! | 1



# Reinforcement learning from human feedback

Step 1: Train a reward (quality) model

Step 2: Have LLM generate a lot of answers. Further train it to generate more responses that get high scores



From Internet

# Language Agents

Meng Fang  
OxML 2024



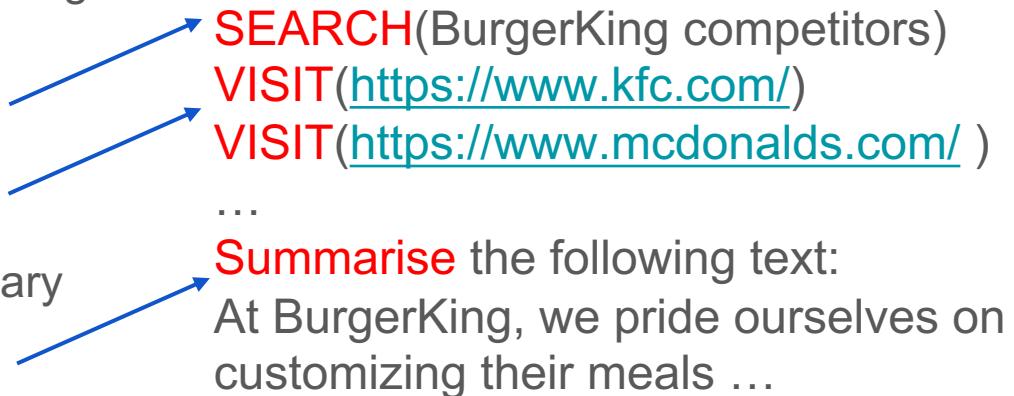
# Agents

Use LLM to choose and carry out complex sequences of actions

Cutting edge area of AI research

Example: Help me research BurgerKing to competitors:

1. Search top competitors
2. Visit website of each competitor
3. For each competitor, write summary based on homepage content



# Agent System

Planning

- Decomposition

Reasoning

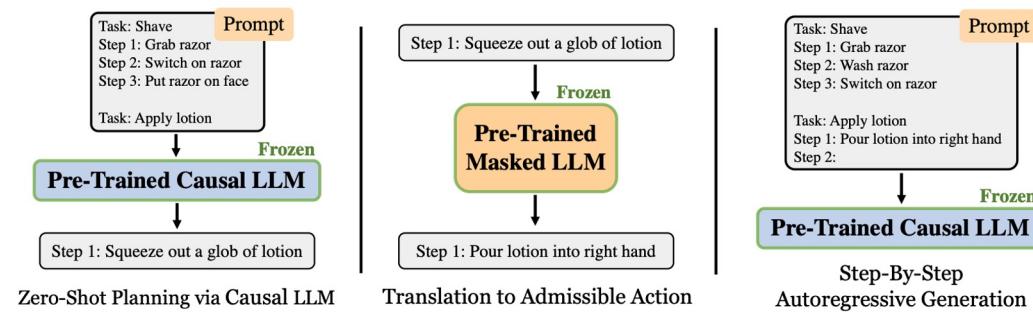
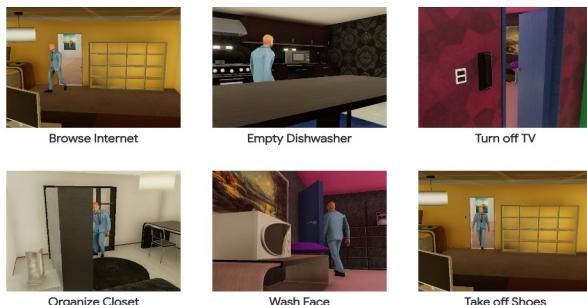
Tool use

- The agent learns to call external APIs for extra information that is missing from the model weights (often hard to change after pre-training)

# Planning

Decompose high-level tasks into sensible mid-level action plans

Each step is admissible action



Large Language Models (LLMs) such as GPT-3 and Codex can plan actions for embodied agents, even without any additional training.

# Planning

## Decomposition

### Stage 1: Decompose Question into Subquestions

Q: It takes Amy 4 minutes to climb to the top of a slide. It takes her 1 minute to slide down. The water slide closes in 15 minutes. How many times can she slide before it closes?

Language Model

A: To solve "How many times can she slide before it closes?", we need to first solve: "How long does each trip take?"

### Stage 2: Sequentially Solve Subquestions

Subquestion 1

It takes Amy 4 minutes to climb to the top of a slide. It takes her 1 minute to slide down. The slide closes in 15 minutes.

Q: How long does each trip take?

Language Model

A: It takes Amy 4 minutes to climb and 1 minute to slide down.  $4 + 1 = 5$ . So each trip takes 5 minutes.

Append model answer to Subquestion 1

Subquestion 2

It takes Amy 4 minutes to climb to the top of a slide. It takes her 1 minute to slide down. The slide closes in 15 minutes.

Q: How long does each trip take?

A: It takes Amy 4 minutes to climb and 1 minute to slide down.  $4 + 1 = 5$ . So each trip takes 5 minutes.

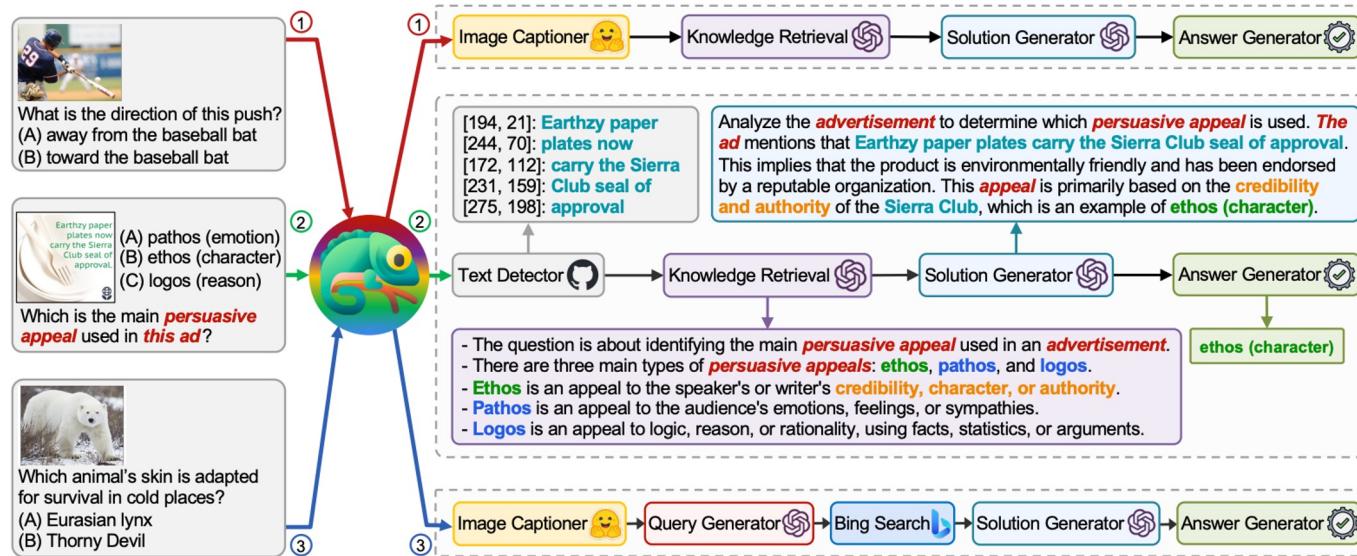
Q: How many times can she slide before it closes?

Language Model

A: The water slide closes in 15 minutes. Each trip takes 5 minutes. So Amy can slide  $15 \div 5 = 3$  times before it closes.

# Planning and Reasoning

Synthesize programs to compose various tools and executing them sequentially to get final answers



# Planning, Reasoning, Tool use

## Examples

**Miles hiked**

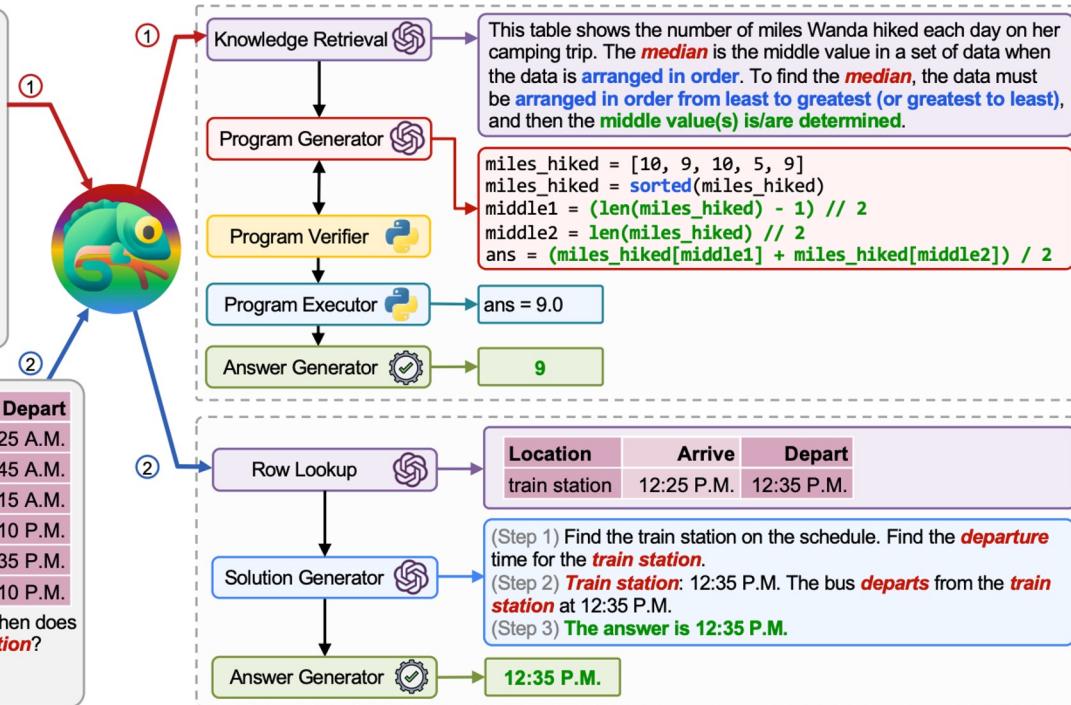
Day	Miles
Sunday	10
Monday	9
Tuesday	10
Wednesday	5
Thursday	9

Wanda went on a camping trip and logged the number of miles she hiked each day. What is the **median** of the numbers?

**Location**      **Arrive**      **Depart**

stadium	10:20 A.M.	10:25 A.M.
park	10:35 A.M.	10:45 A.M.
hotel	11:10 A.M.	11:15 A.M.
airport	12:05 P.M.	12:10 P.M.
train station	12:25 P.M.	12:35 P.M.
bus station	1:10 P.M.	1:10 P.M.

Look at the following schedule. When does the bus **depart** from the **train station**?  
(A) 12:35 P.M. (B) 1:10 P.M.  
(C) 1:10 P.M. (D) 10:45 A.M.



# Summary

Introduction to Generative AI and LLMs

Prompt Engineering

Expanding knowledge for LLMs

Language agents

Thank you.

Q & A