



香港中文大學(深圳)  
The Chinese University of Hong Kong, Shenzhen

# DDA6307/CSC6052/MDS6002: Natural Language Processing

## Lecture 6: Prompt Engineering

Spring 2024  
Benyou Wang  
School of Data Science

# Recap

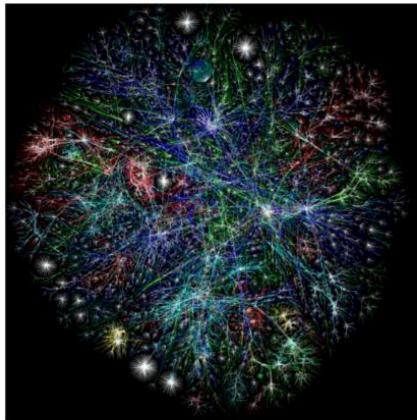
1. **What** are Large Language Models (LLMs)?
2. **Why** LLMs?
3. **How** do we use LLMs?

# What is an LLM

Large, general-purpose language  
models can be pre-trained and then  
fine-tuned for specific purposes

# LLMs (Intelligence) as compression

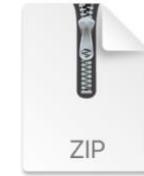
Think of it like compressing the internet.



Chunk of the internet,  
~10TB of text



6,000 GPUs for 12 days, ~\$2M  
~ $1e24$  FLOPS

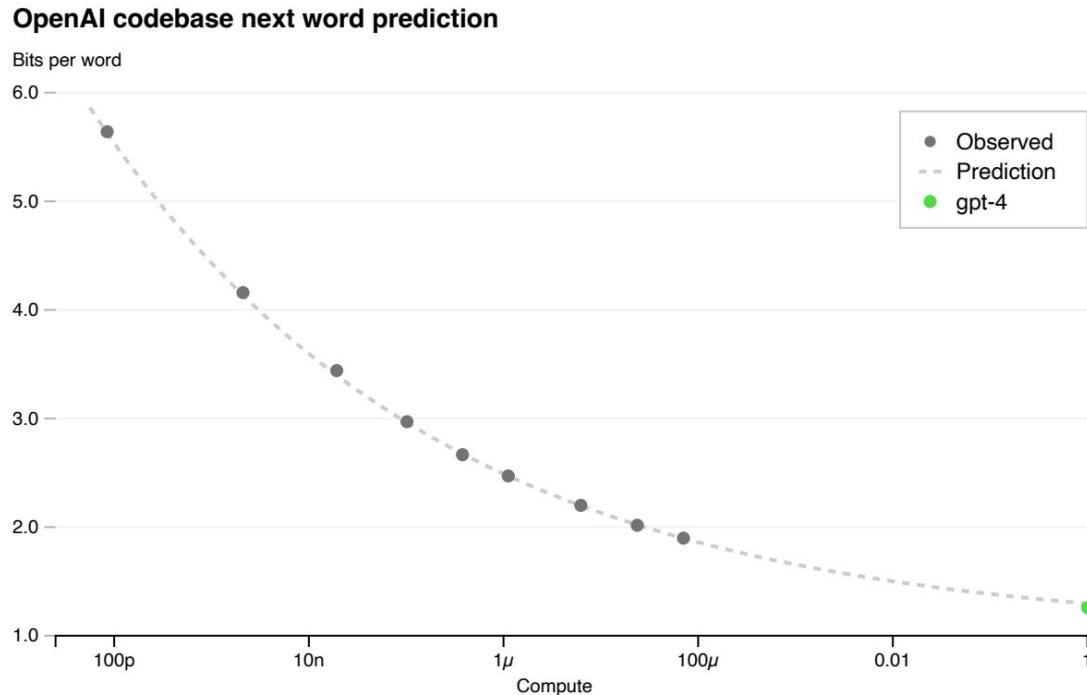


parameters.zip

~140GB file

\*numbers for Llama 2 70B

# The Scaling law of GPT-4 in terms of Compression



# Why Larger language models

- More world knowledge (LAMA)
  - Language models as knowledge base?
- Larger capacity to learn problem-solving Abilities
  - Coding, revising articles, reasoning etc.
- Better generalization to unseen tasks

- Emergent ability (涌现能力)

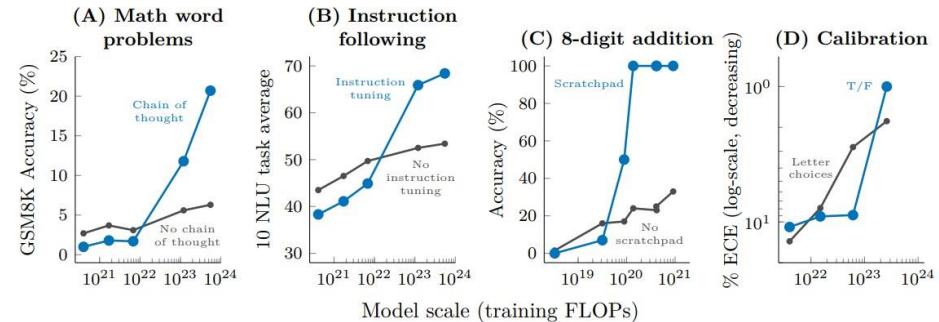


Figure 3: Specialized prompting or finetuning methods can be emergent in that they do not have a positive effect until a certain model scale. A: Wei et al. (2022b). B: Wei et al. (2022a). C: Nye et al. (2021). D: Kadavath et al. (2022). An analogous figure with number of parameters on the x-axis instead of training FLOPs is given in Figure 12. The model shown in A-C is LaMDA (Thoppilan et al., 2022), and the model shown in D is from Anthropic.

# Why Small language models (SLM)

TinyLLaMA: 1.1B

MobileVLM: 1.4B and 2.7B

MobiLlama 0.5B

MobileLLM: 0.1B and 0.3B

- Cheaper but with comparable performance at least in a narrow domain
- Real-time inference in the edges devices

[1] MobileLLM: Optimizing Sub-billion Parameter Language Models for On-Device Use Cases. <https://arxiv.org/pdf/2402.14905.pdf>

[2] MobiLlama: Towards Accurate and Lightweight Fully Transparent GPT. <https://arxiv.org/abs/2402.16840>

[3] MobileVLM : A Fast, Strong and Open Vision Language Assistant for Mobile Devices. <https://arxiv.org/abs/2312.16886>.

[4] TinyLlama: An Open-Source Small Language Model. <https://arxiv.org/abs/2401.02385>

# Level of Using LLMs

**Level 1:** Prompt Engineering

Today's lecture

**Level 2:** Model Fine-tuning

Lecture in next week

**Level 3:** Build Your Own



For a given problem, you could

## Prompt engineering

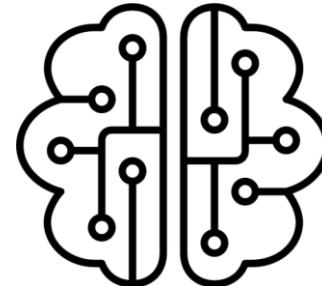
- fancy prompt (e.g. COT)
- Retrieval-augmented generation
- *usually does not inject too much info*

## Finetuning your model

- collect data
- training it



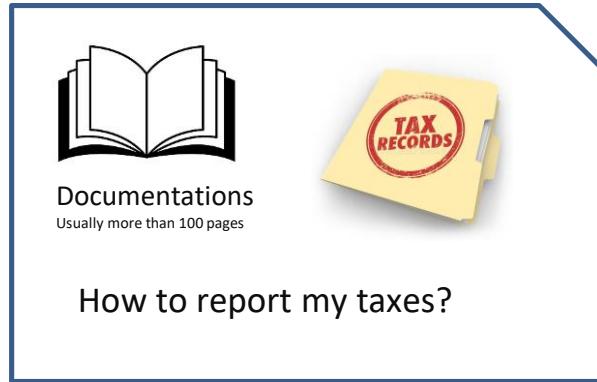
Change how we express  
the information need



Change the model  
who deals with the need

# A new way to balance it: RAG and long context

Super long context to describe the background and inject some knowledges



Today's lecture

# Today's Lecture

- **Knowledge** in LLMs
  - LLMs as knowledge bases
  - Facts updating for LLMs
- **Reasoning** in LLMs
  - Why reasoning is special in LLMs
- **Prompt** Techniques for better reasoning

# Today's Lecture

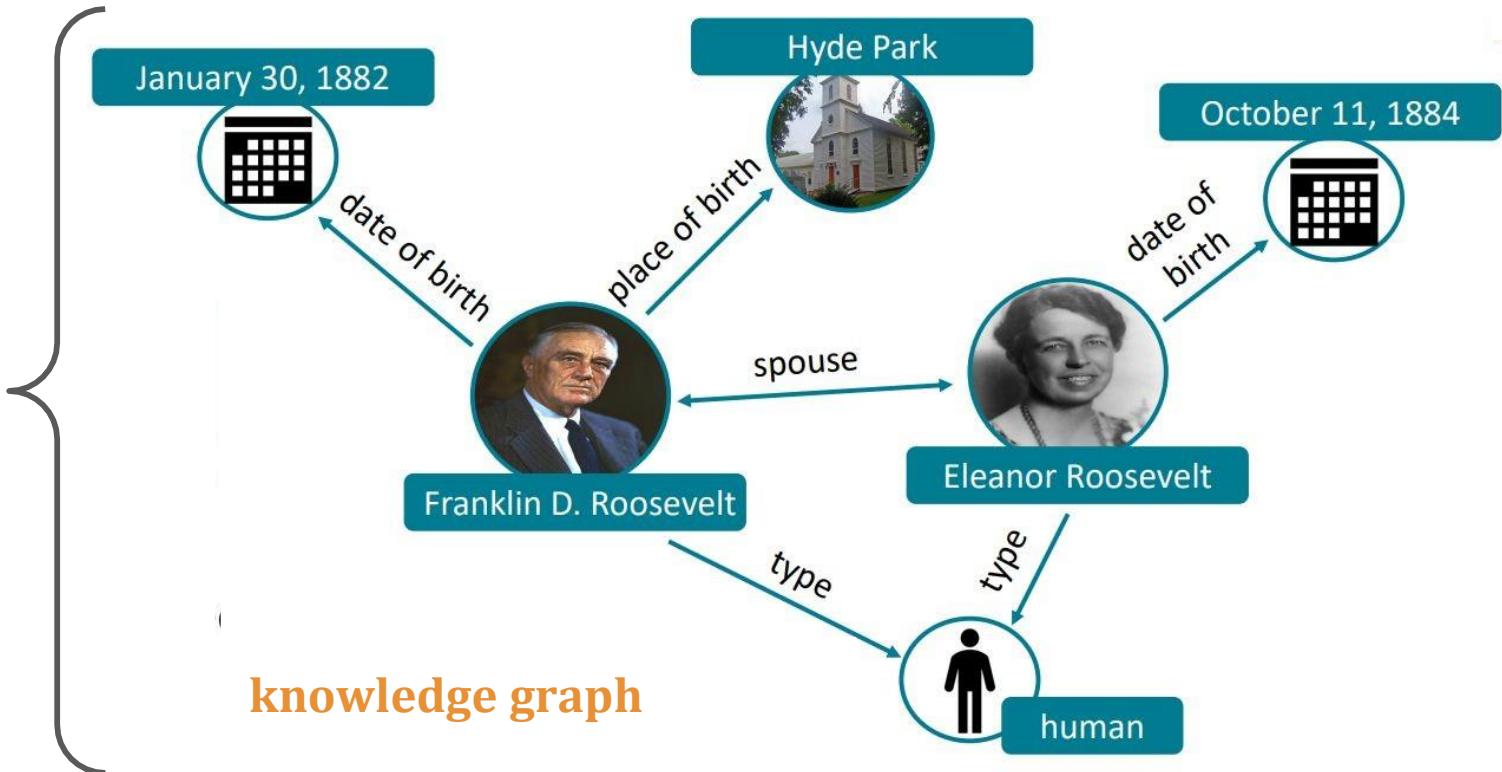
- **Knowledge** in LLMs
  - LLMs as knowledge bases
  - Facts updating for LLMs
- **Reasoning** in LLMs
  - Why reasoning is special in LLMs
- **Prompt** Techniques for better reasoning

What is a knowledge base?

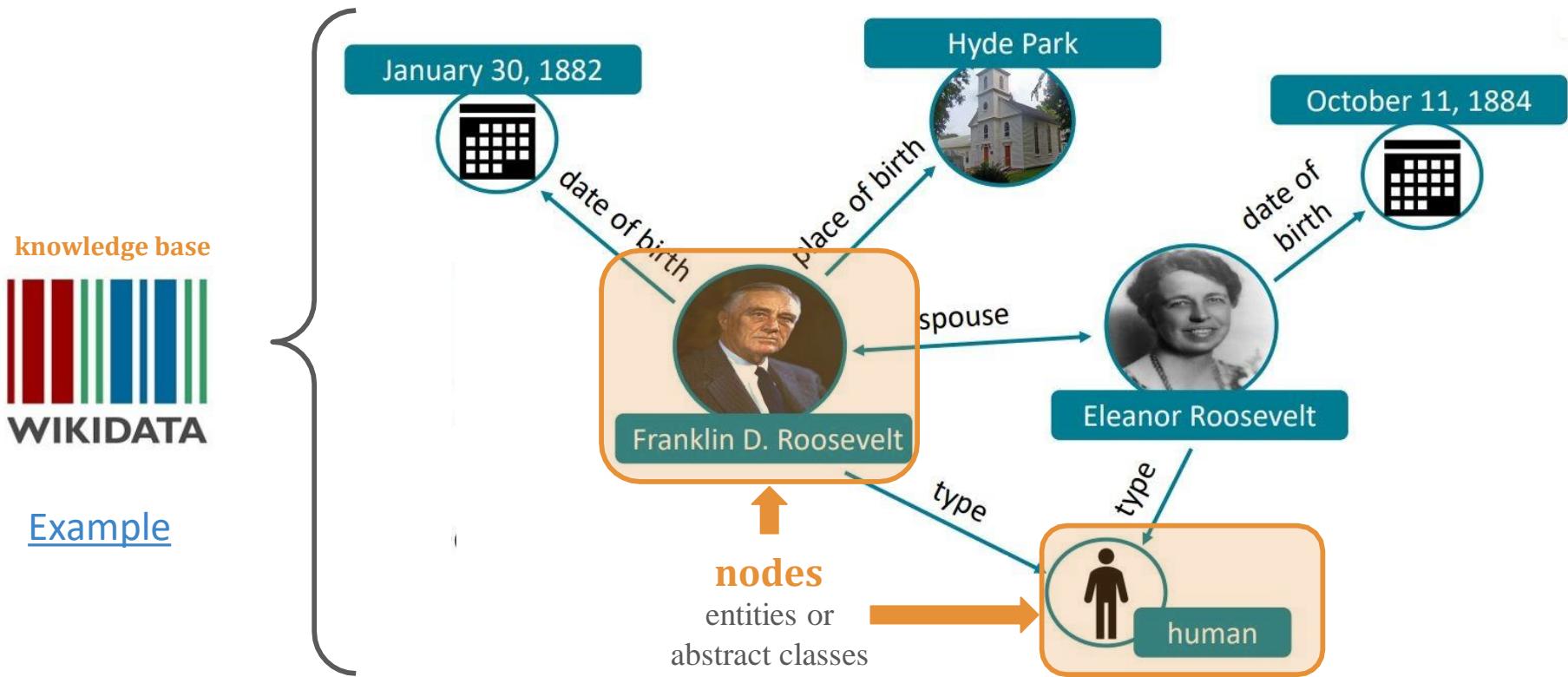
# What is a knowledge base?

knowledge base  
  
WIKIDATA

Example



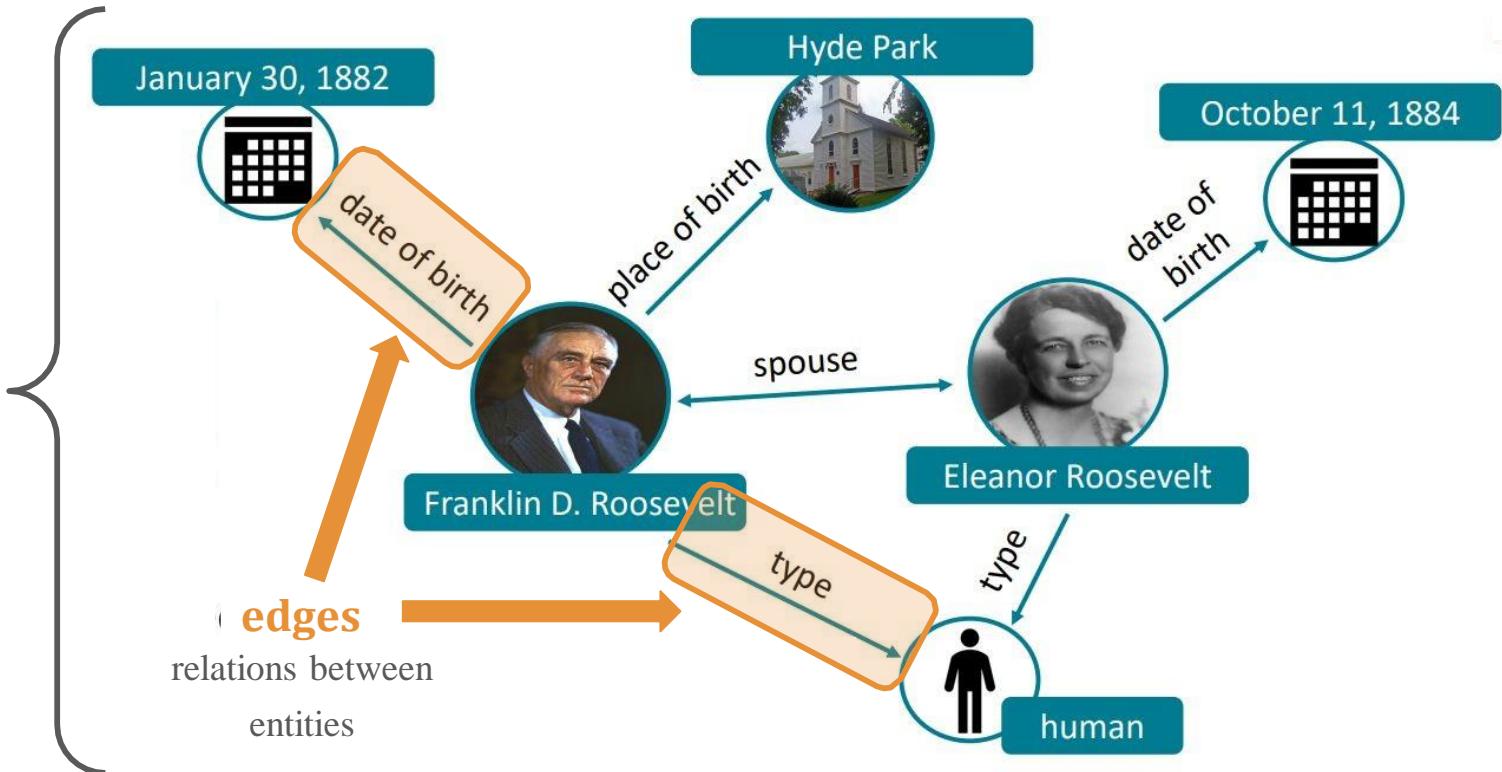
# What is a knowledge base?



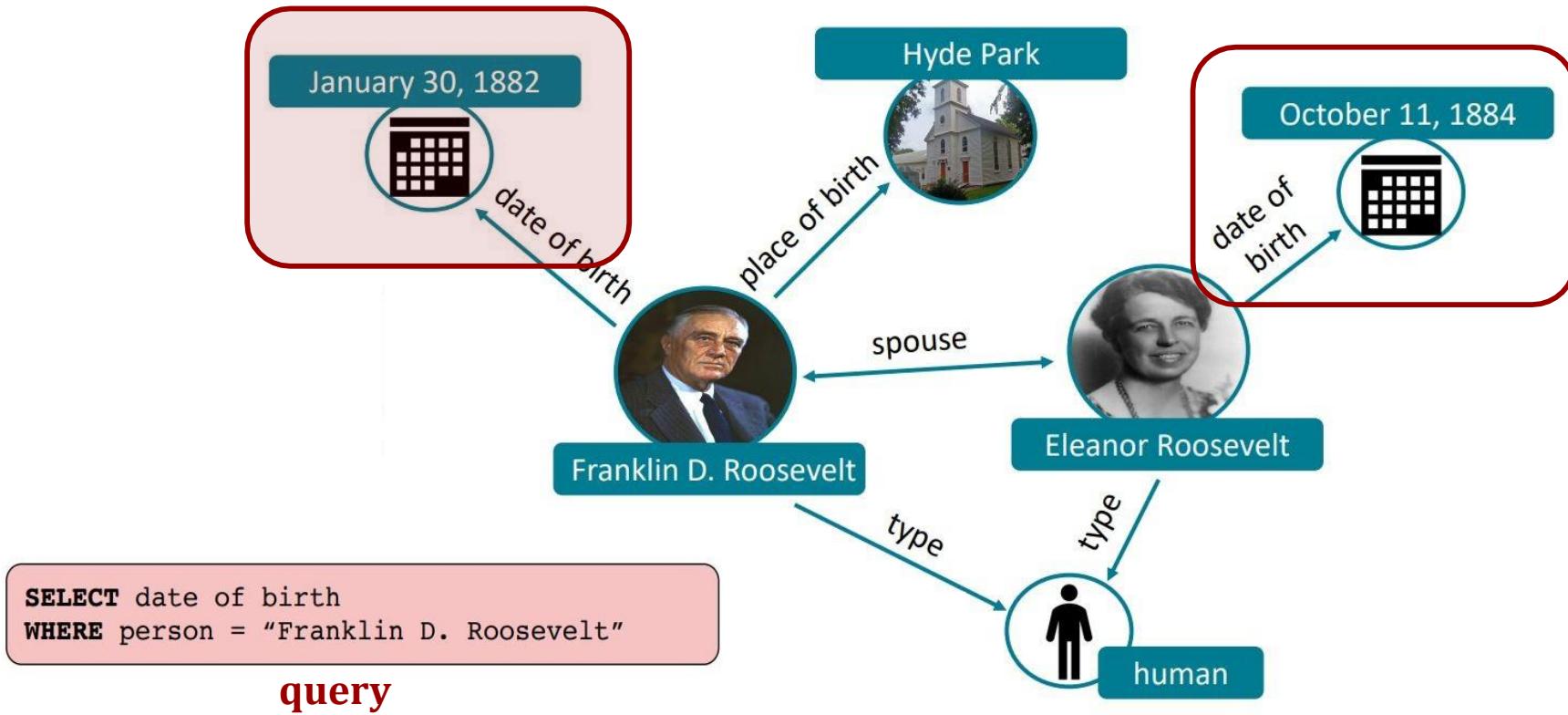
# What is a knowledge base?



Example



# How to query?



# How were knowledge bases formed?

**Valorant**

From Wikipedia, the free encyclopedia

**Valorant** (stylized as VALORANT) is a first-person shooter game developed by Riot Games and published by them under the company's拳头游戏 brand. The game began development in 2014, and was released on April 7, 2022.

**Brie**

From Wikipedia, the free encyclopedia

(Redirected from [Brie \(red\)](#))

**Brie** (brie, brie cheese) is a soft cheese named after the town of Brie in France. It originated in the department of Seine-et-Marne with a sight that would be depending on the manufacturer. It is a central figure in English-language Modernist poetry.

**Thomas Stearns Eliot OM** (26 September 1888 – 4 January 1965) was a poet, essayist, publisher, playwright, literary critic and editor.<sup>[2]</sup> Considered one of the 20th century's major poets, he is a central figure in English-language Modernist poetry.

Born in St. Louis, Missouri, to a prominent Brahmin family, he moved to England in 1914 at the age of 25 and went on to settle, work, and marry there.<sup>[3]</sup> He became a British citizen in 1927 at the age of 39, subsequently renouncing his American citizenship.<sup>[4]</sup>

Eliot first attracted widespread attention for his poem "The Love Song of J. Alfred Prufrock" in 1915, which, at the time of its publication, was considered outstanding.<sup>[5]</sup> It was followed by "The Waste Land" (1922), "The Hollow Men" (1925), "Ash Wednesday" (1930), and *Four Quartets* (1943).<sup>[6]</sup> He was also known for seven plays, particularly *Murder in the Cathedral* (1935) and *The Cocktail Party* (1949). He was awarded the 1948 Nobel Prize in Literature, "for his outstanding, pioneer contribution to present-day poetry".<sup>[7][8]</sup>

**Cont.**

- 1 Gangster
- 11 Uri
- 12 Spy
- 13 Co
- 14 De
- 15 Es
- 16 Re
- 17 Sh

**Agents**

- 21 Du
- 22 Se
- 23 Inf
- 24 Co
- 3 Store
- 4 Devoted

**Skills**

- 1 Product
- 2 Nutton
- 3 Varieties
- 31 Br
- 32 Br
- 33 Br

**Bio**

Thomas Stearns Eliot  
26 September 1888  
St. Louis, Missouri, US

**Died**

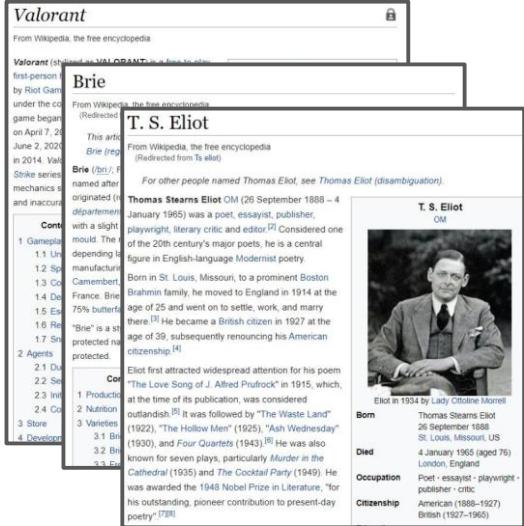
4 January 1965 (aged 76)  
London, England

**Occupation**

Poet · essayist · playwright · publisher · critic

**Citizenship**

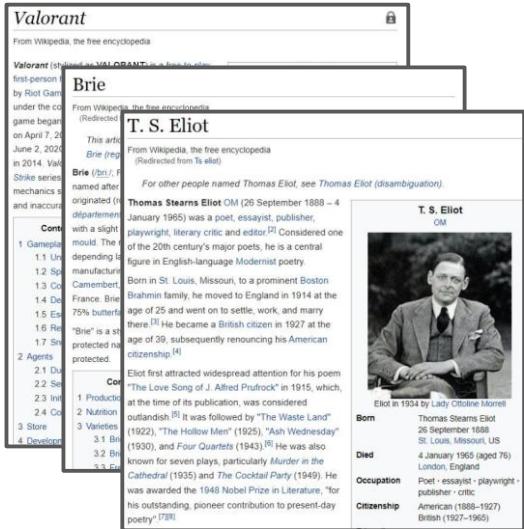
American (1888–1927)  
British (1927–1965)



unstructured text

Structured knowledge base

# How were knowledge bases formed?

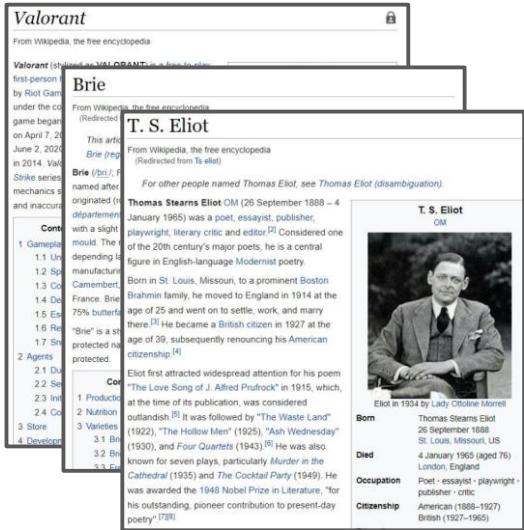


**Knowledge Extraction  
Pipeline**

unstructured text

Structured knowledge base

# How were knowledge bases formed?



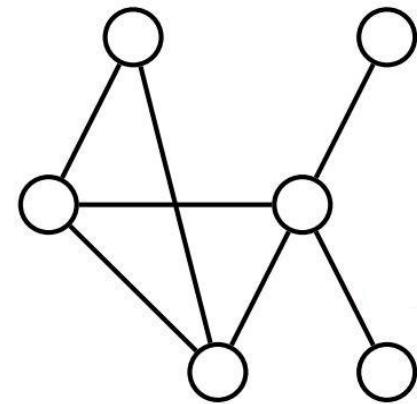
unstructured text



Knowledge Extraction  
Pipeline



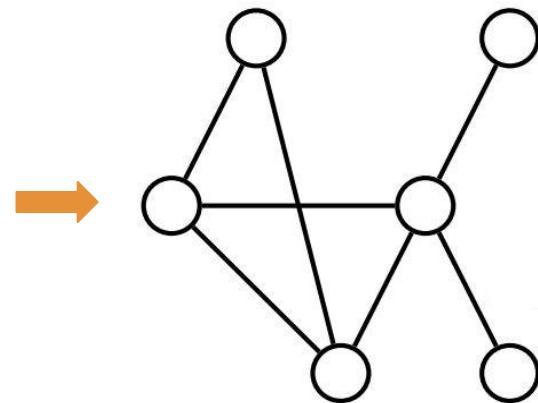
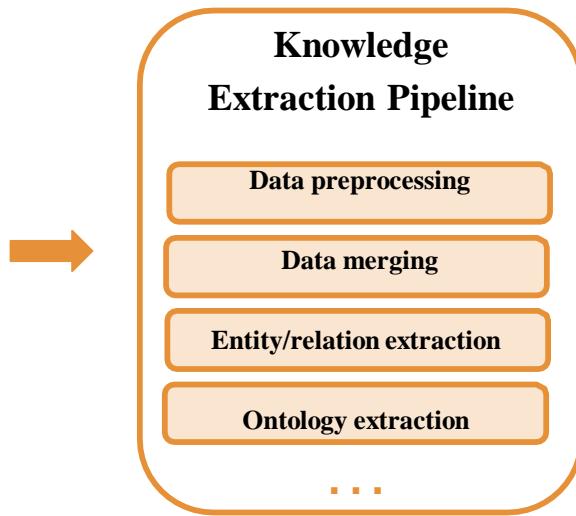
Structured knowledge base



# Downsides of using knowledge bases

# Downsides of using knowledge bases

The image shows two side-by-side screenshots of Wikipedia pages. The top screenshot is for the page 'Valorant' (Valorant (video game)). It displays a warning message: 'This article is flagged as having serious problems. You can help improve it by fixing the problems listed on its talk page.' Below this, the page content is mostly empty, with only a few lines of text about the game's history and mechanics. The bottom screenshot is for the page 'Brie'. It also has a warning message: 'This article is flagged as having serious problems. You can help improve it by fixing the problems listed on its talk page.' The page content is more substantial, featuring a biography of T. S. Eliot, a portrait photo, and several sections of text describing his life and work.



Populating the knowledge base often involves **complicated, multi-step NLP pipelines**

# Downsides of using knowledge bases



unstructured text

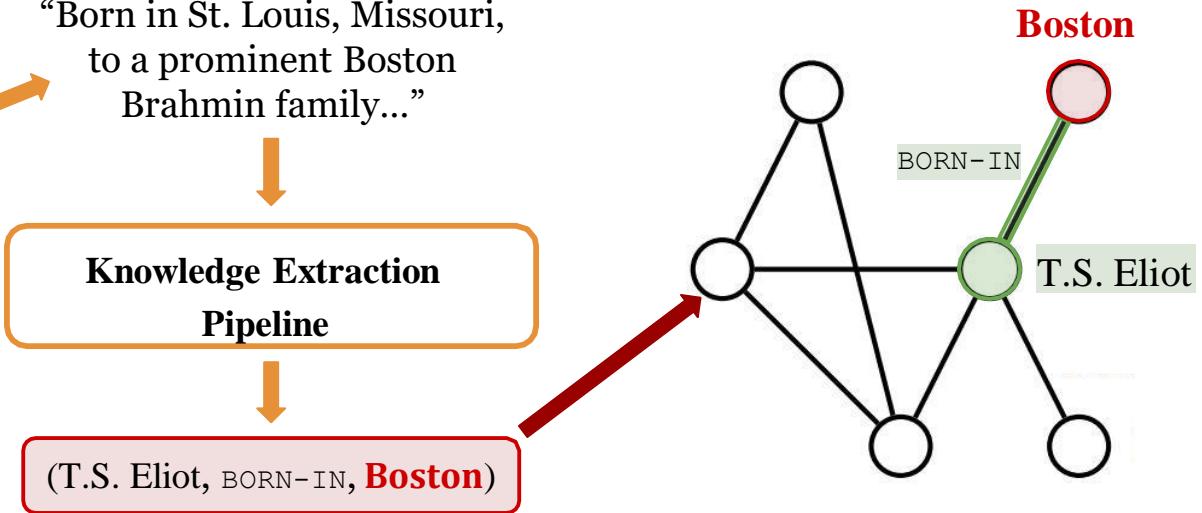
“Born in St. Louis, Missouri,  
to a prominent Boston  
Brahmin family...”

Knowledge Extraction  
Pipeline

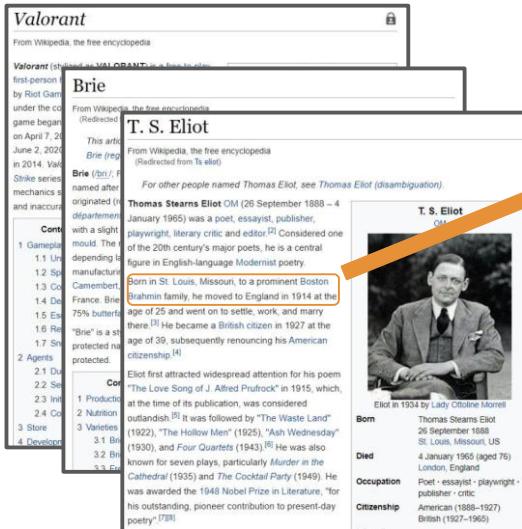
(T.S. Eliot, BORN-IN, Boston)

incorrect extraction

Prone to **error propagation** (from human annotations or knowledge extraction)



# Downsides of using knowledge bases



unstructured text

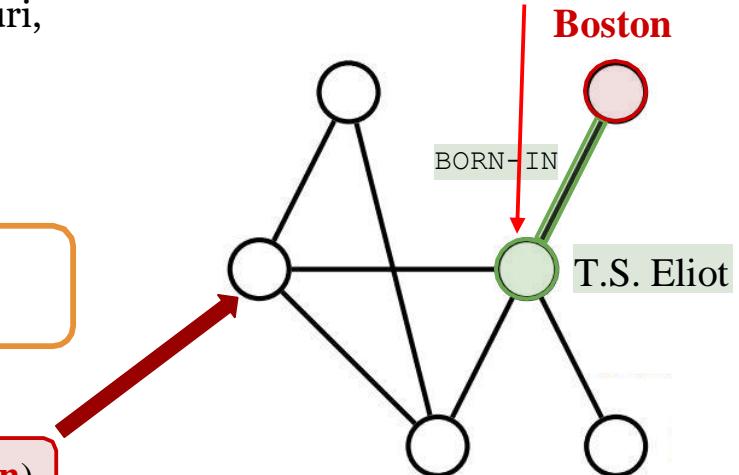
“Born in St. Louis, Missouri,  
to a prominent Boston  
Brahmin family...”

Knowledge Extraction  
Pipeline

(T.S. Eliot, BORN-IN, Boston)

incorrect extraction

Q: Describe Eliot's family's related information.



Triples lead to **information loss**: hard to include all possible information we may be interested in.

## Are there better alternatives?

Traditional knowledge bases are **inflexible**  
and require **significant manual effort**.

# Language Models as Knowledge Bases? (Petroni et al., 2019)

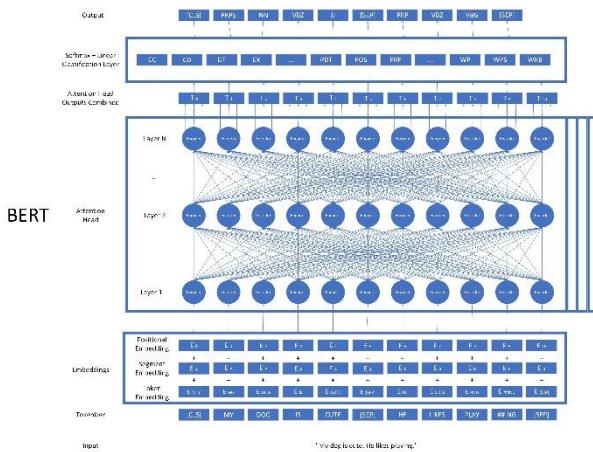
# Language models as knowledge bases?

## Why language models?

- Scalability: pre-trained on a huge corpus of data
- Time/Labor efficiency: does not require annotations/supervision
- Flexibility: more flexible with natural language queries
- Accessibility: can be used off-the-shelf

Do language models really store knowledge?

# LAMA probe





# LAMA Probe



- **Goal:** evaluate **factual + commonsense knowledge** in language models



## LAMA Probe



- **Goal:** evaluate **factual + commonsense knowledge** in **language models**
- Collect set of **knowledge sources** (i.e. set of facts) and test to see how well the model's knowledge captures these facts



## LAMA Probe



- **Goal:** evaluate **factual + commonsense knowledge** in **language models**
- Collect set of **knowledge sources** (i.e. set of facts) and test to see how well the model's knowledge captures these facts
- *How do we know how “knowledgeable” a LM is about a particular fact?*



## LAMA Probe



- **Goal:** evaluate **factual + commonsense knowledge** in **language models**
- Collect set of **knowledge sources** (i.e. set of facts) and test to see how well the model's knowledge captures these facts
- *How do we know how “knowledgeable” a LM is about a particular fact?*

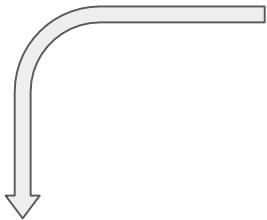
Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower

## Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower

# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower

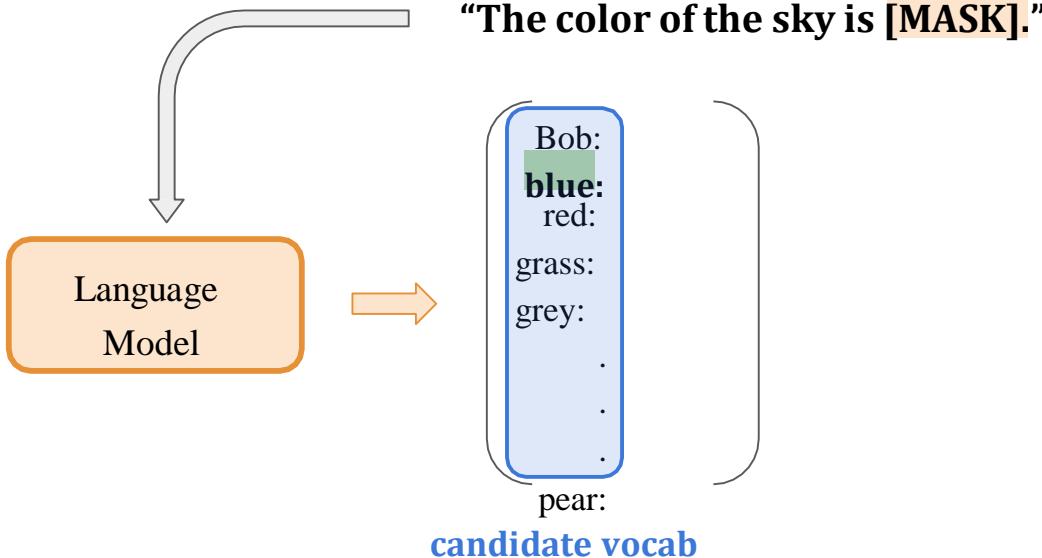


"The color of the sky is [MASK]."

Language  
Model

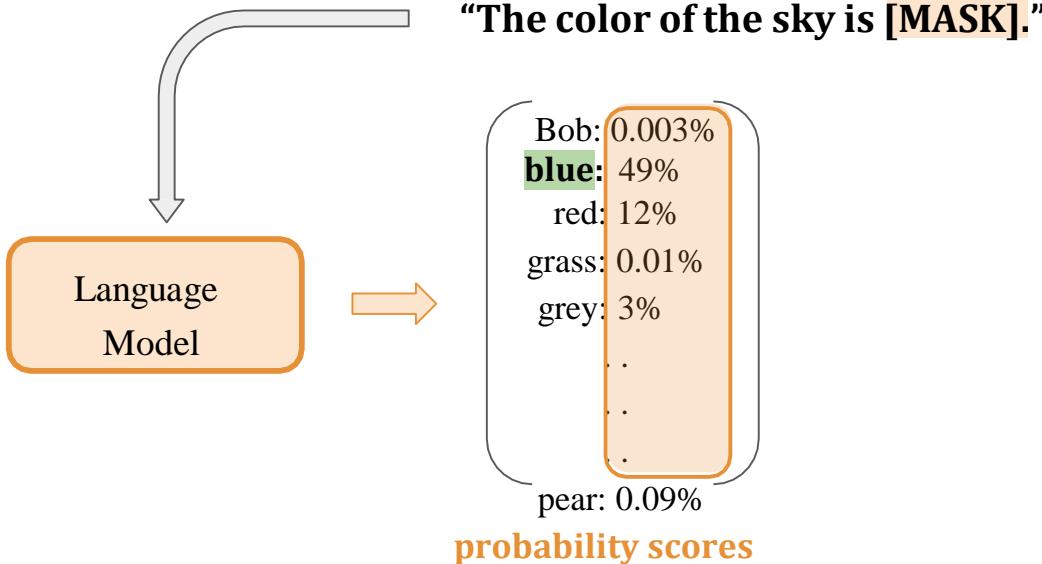
# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



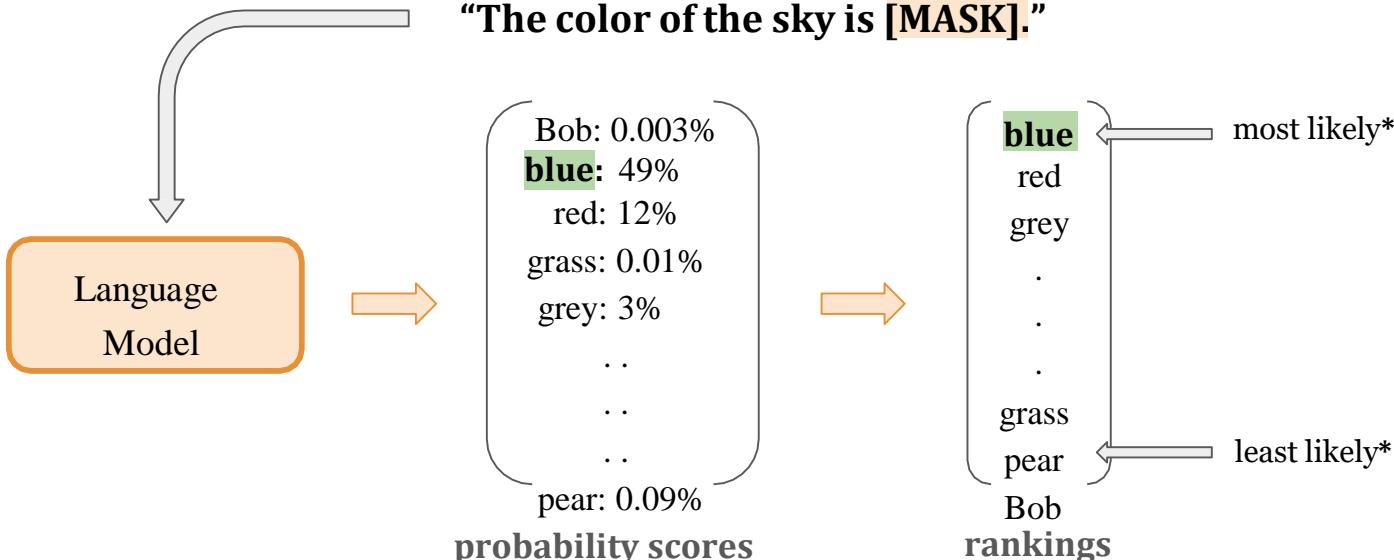
# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



# Evaluation of LM via LAMA

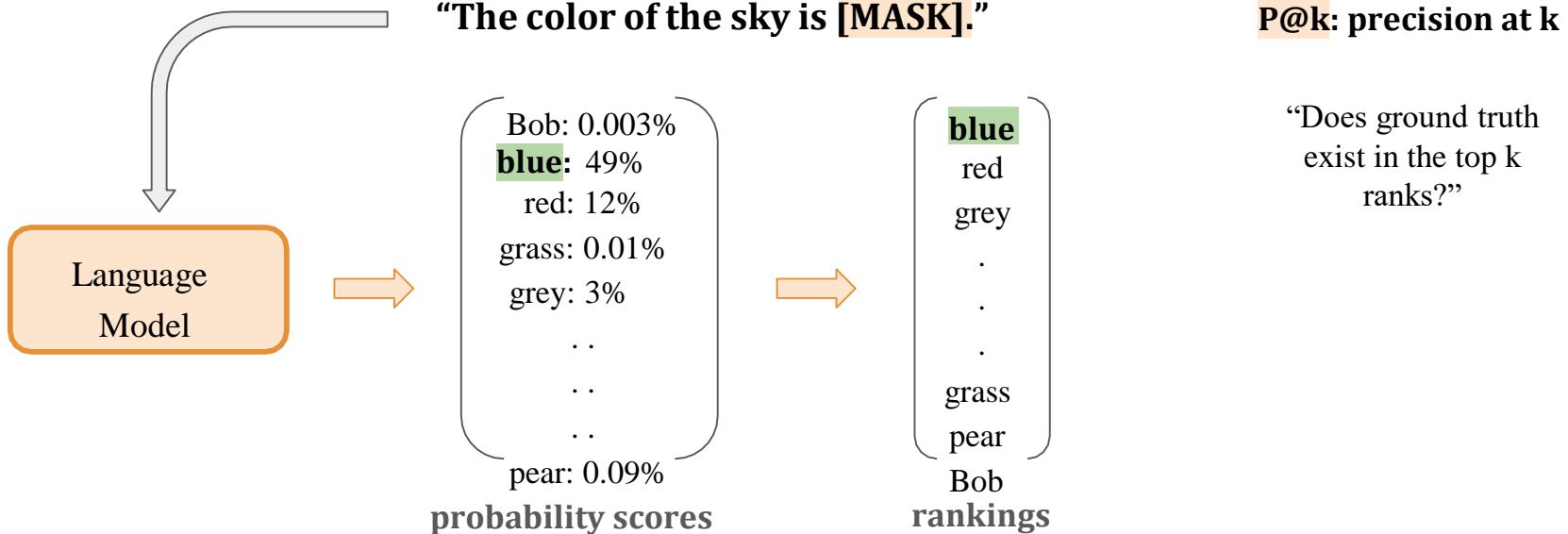
Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



\*according to the LM

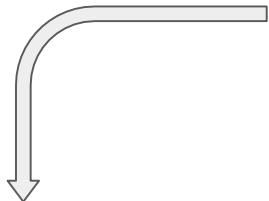
# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower

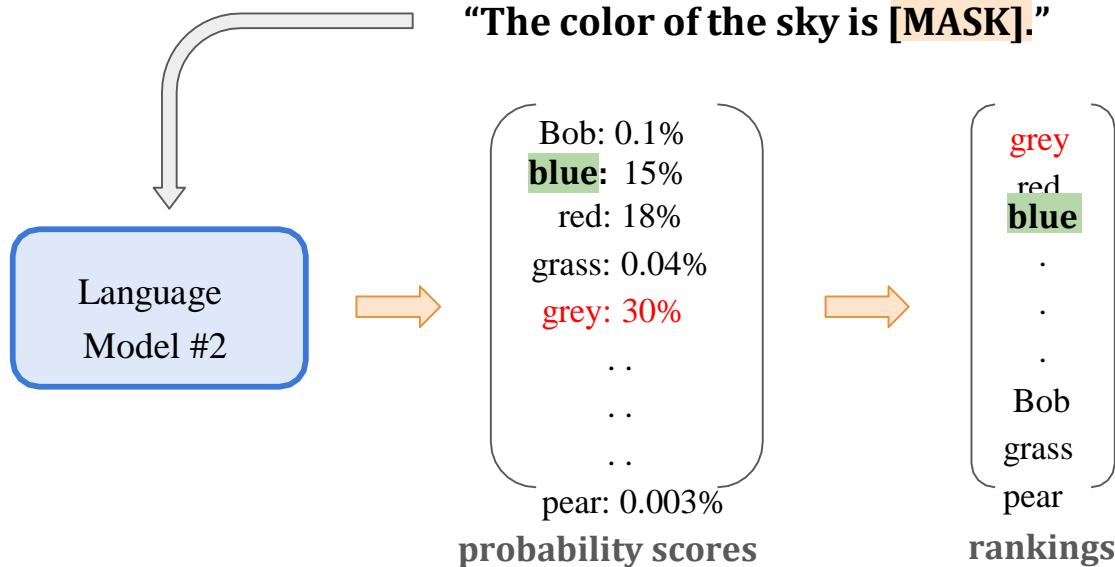


“The color of the sky is [MASK].”

Language  
Model #2

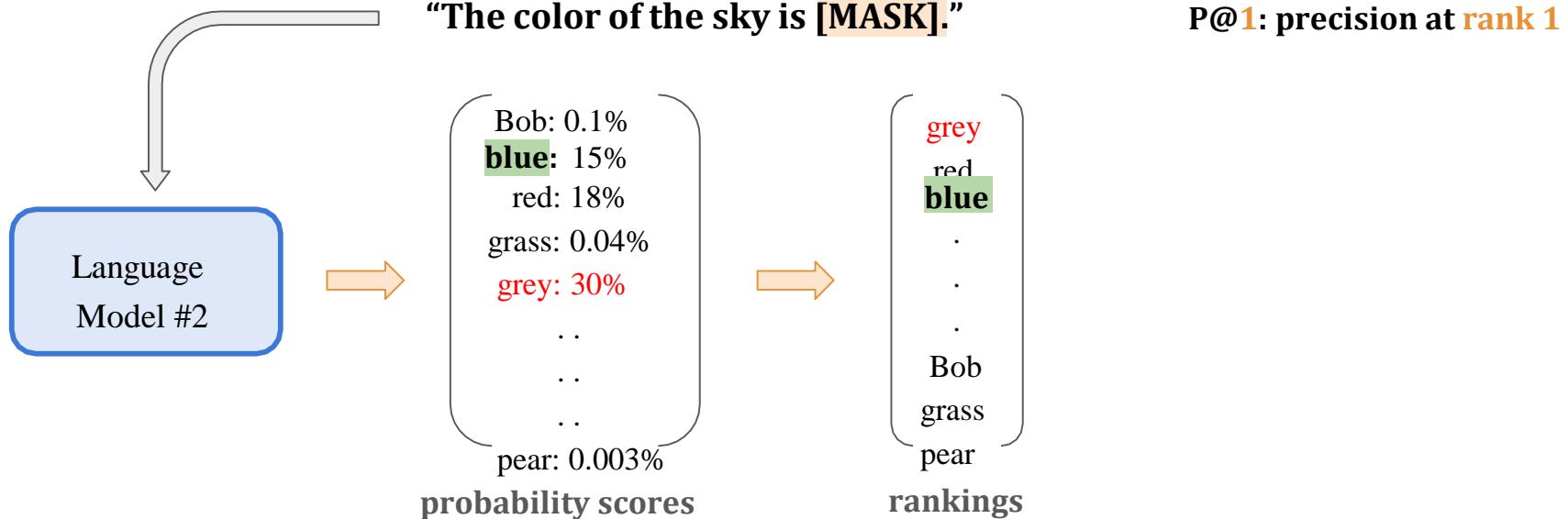
# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



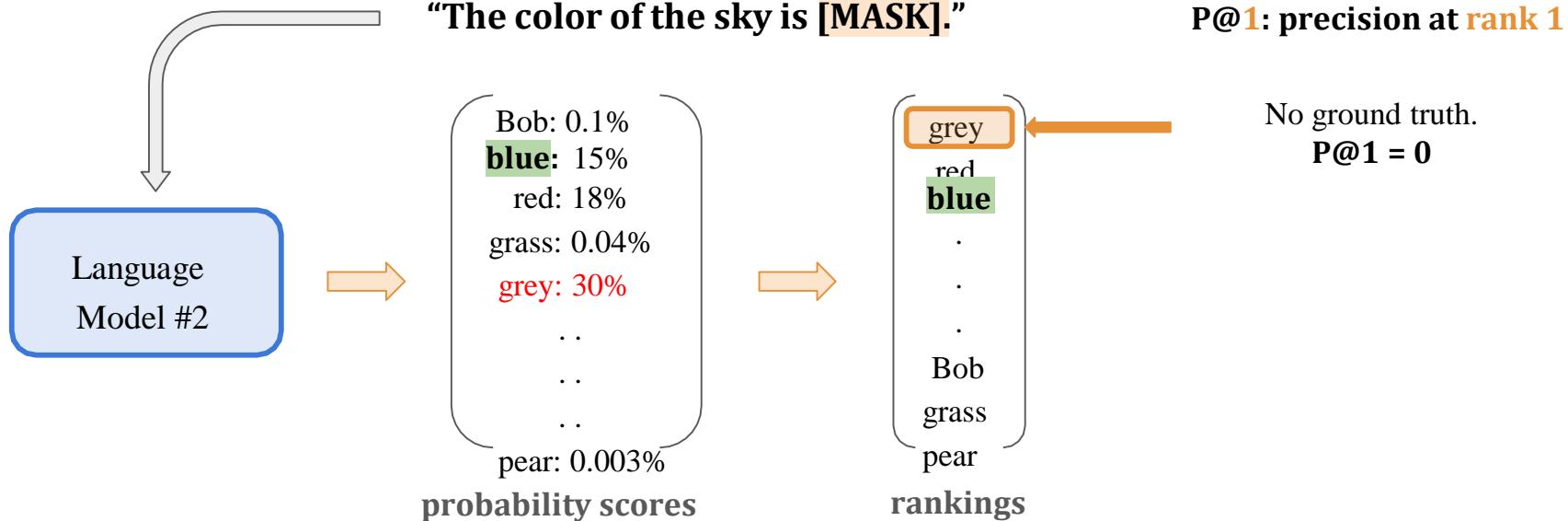
# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



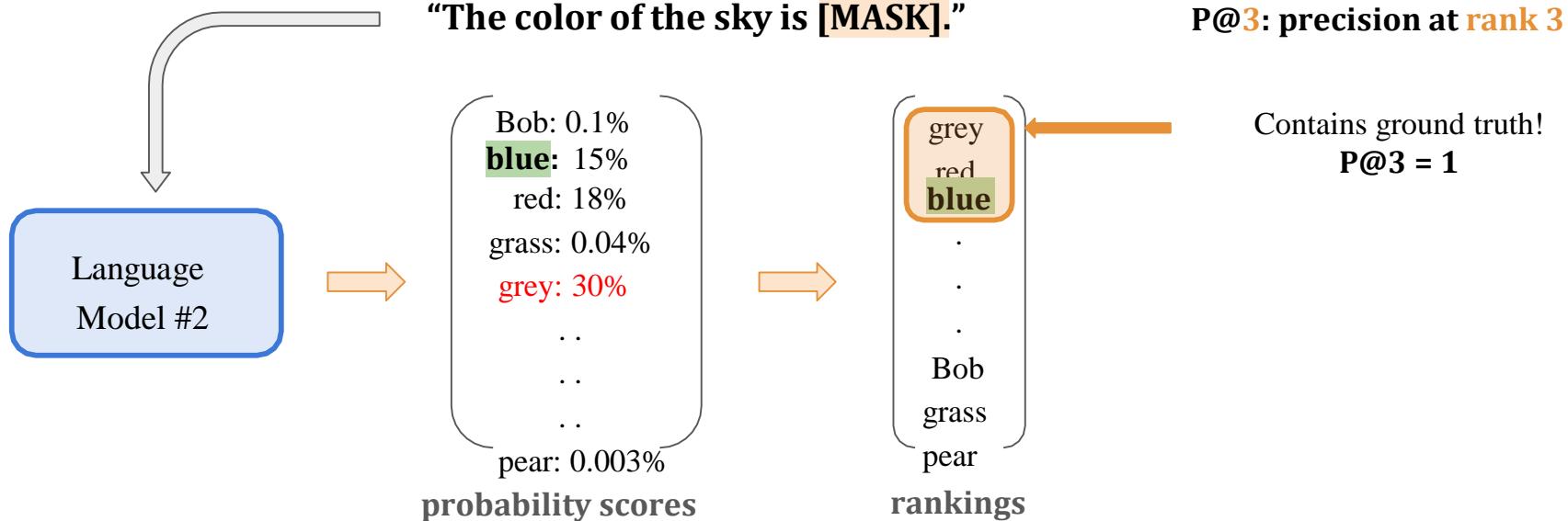
# Evaluation of LM via LAMA

Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



# Evaluation of LM via LAMA

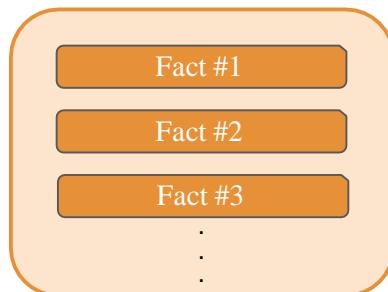
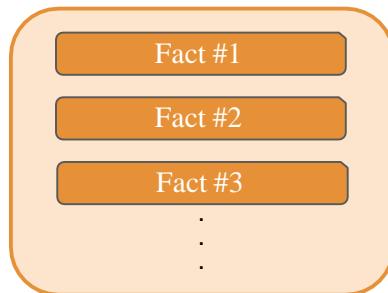
Given a cloze statement that queries the model for a missing token,  
**knowledgeable LMs rank ground truth tokens high** and other tokens lower



# Architecture of the LAMA probe

# Architecture of the LAMA probe

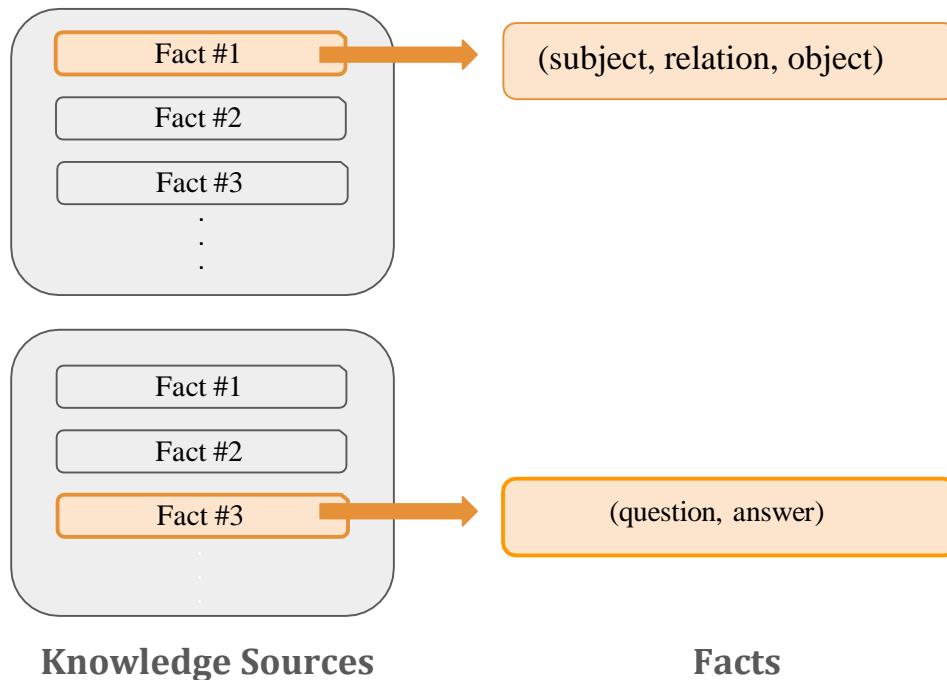
## Step 1: Compile knowledge sources



Knowledge Sources

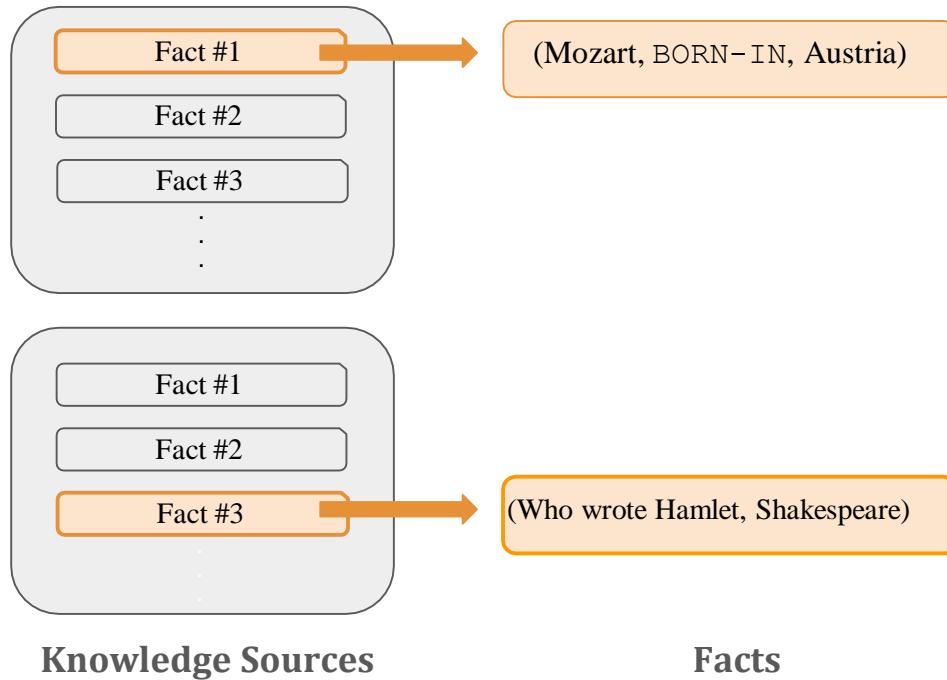
# Architecture of the LAMA probe

## Step 2: Formulate facts into triplets or question-answer pairs



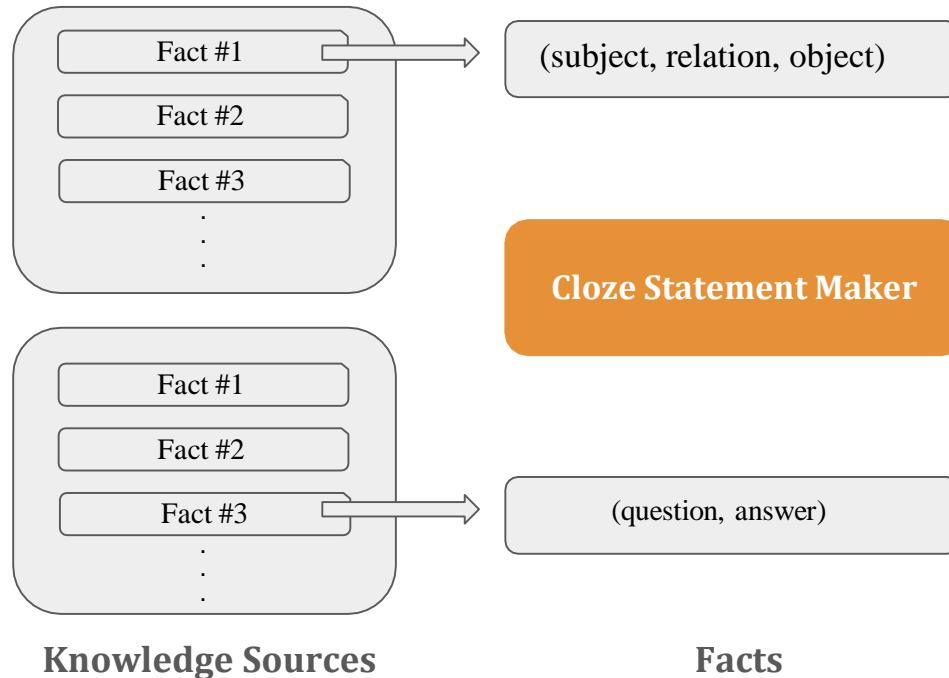
# Architecture of the LAMA probe

## Step 2: Formulate facts into triplets or question-answer pairs



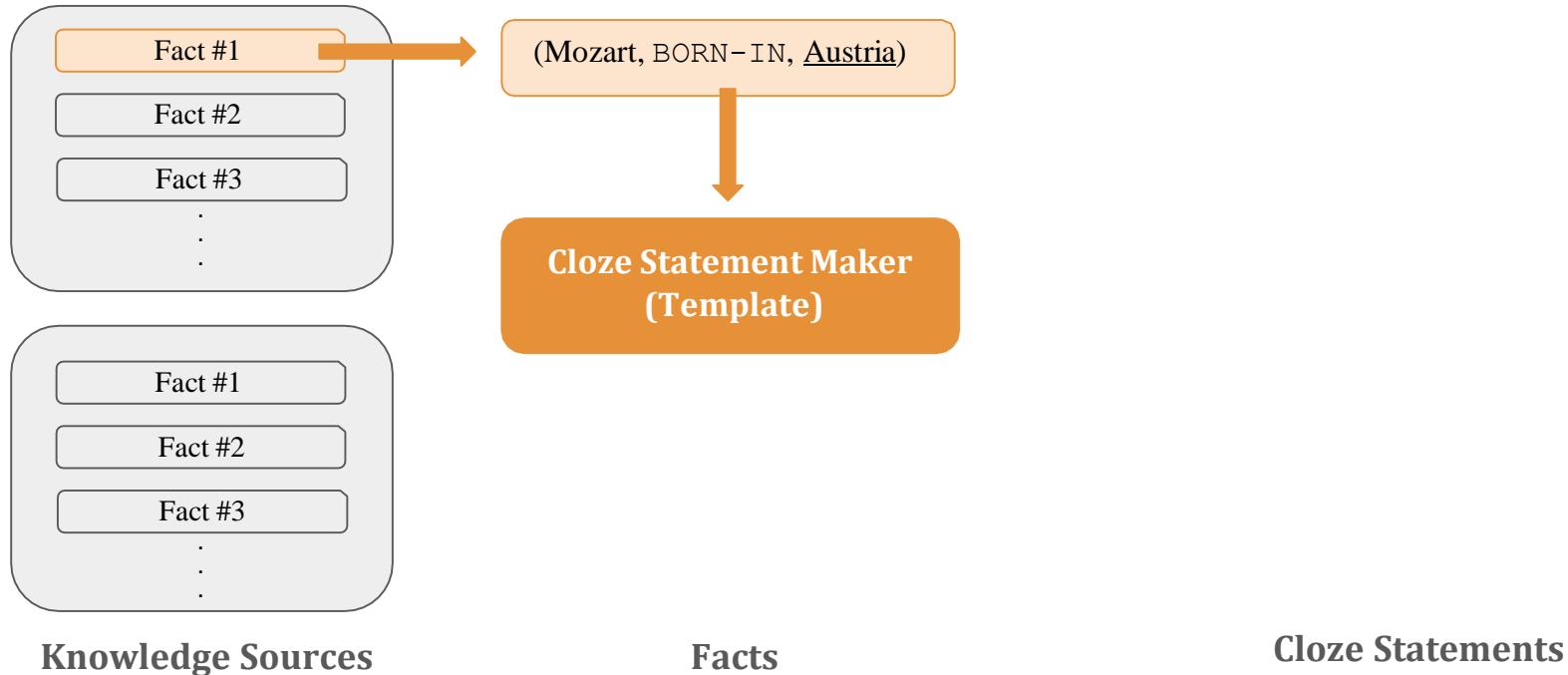
# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



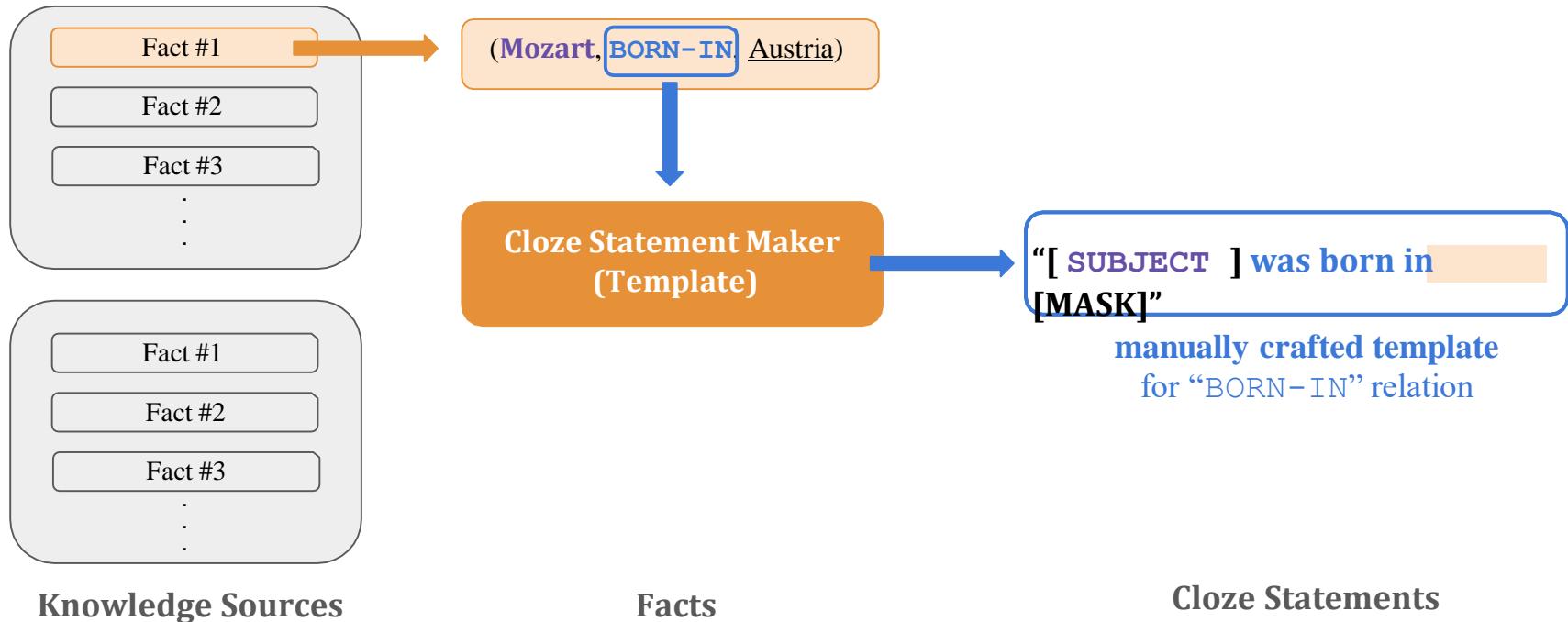
# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



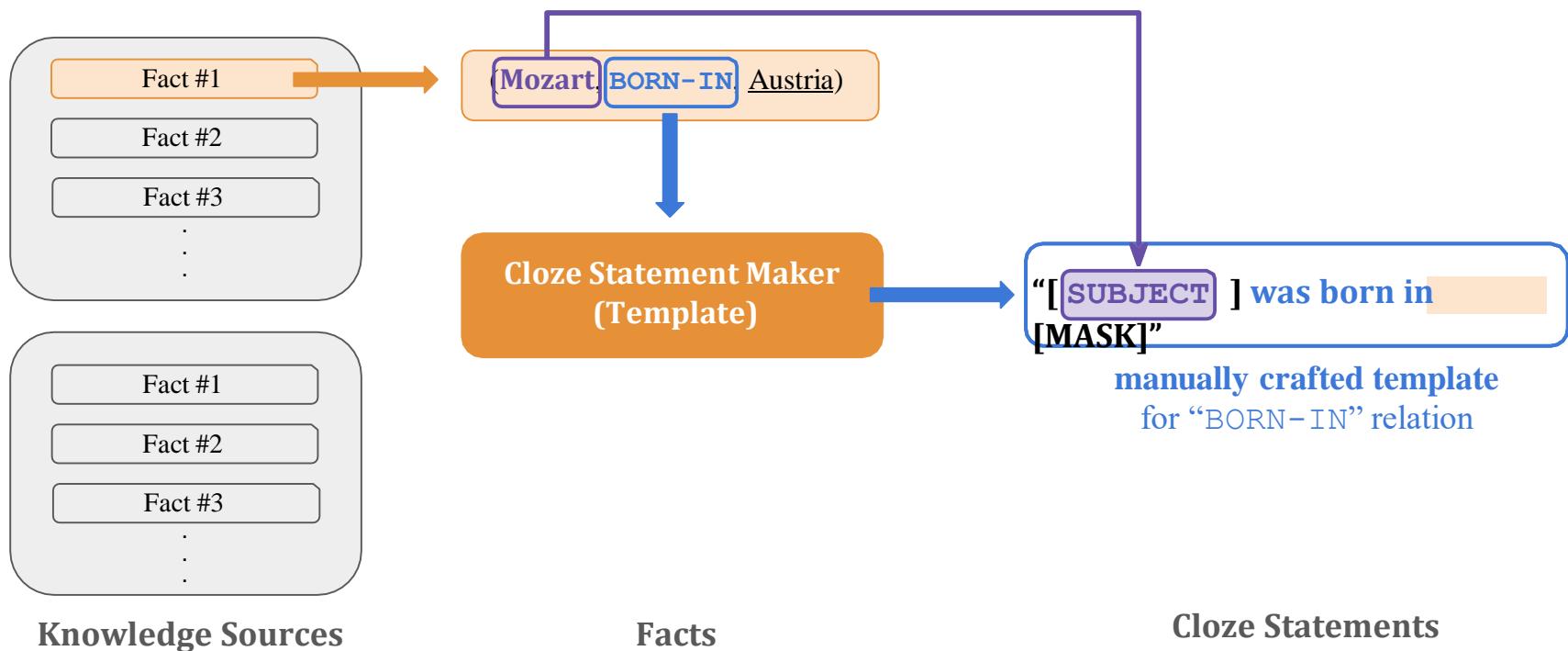
Knowledge Sources

Facts

Cloze Statements

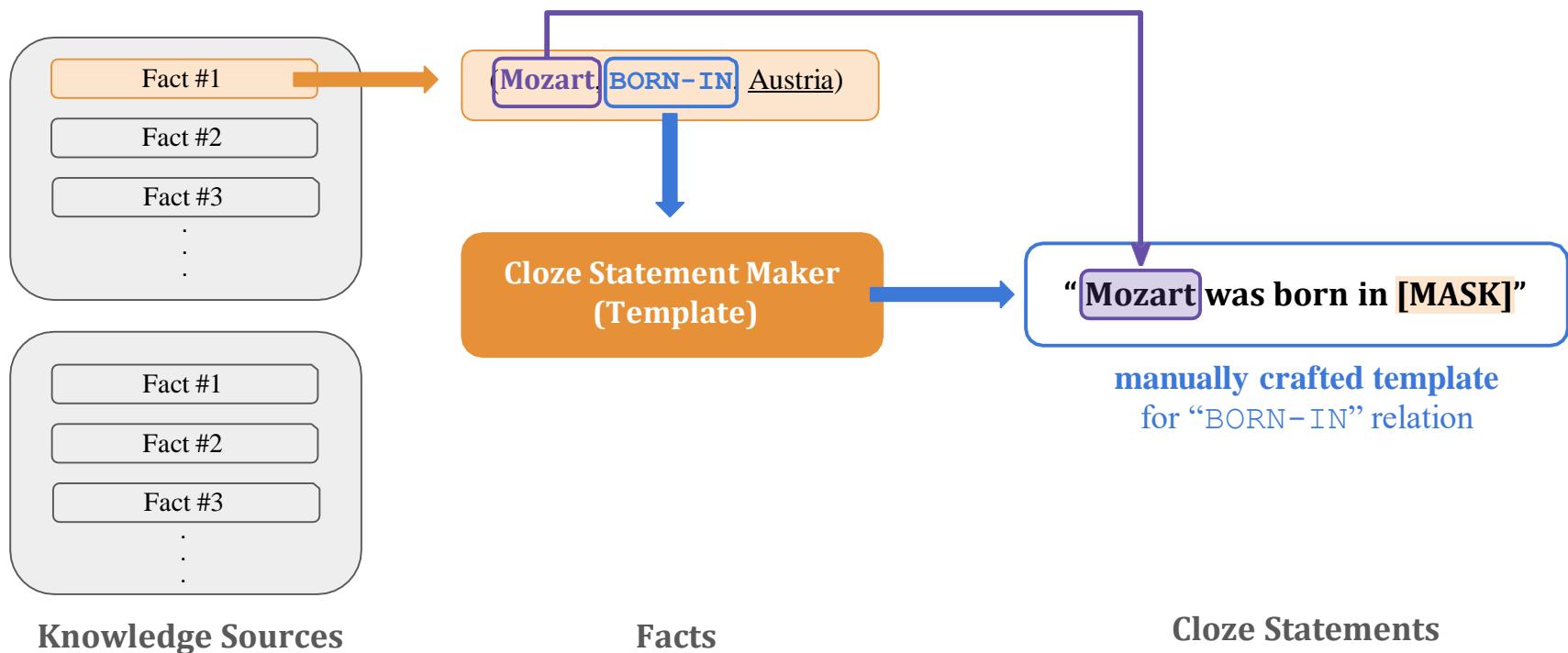
# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



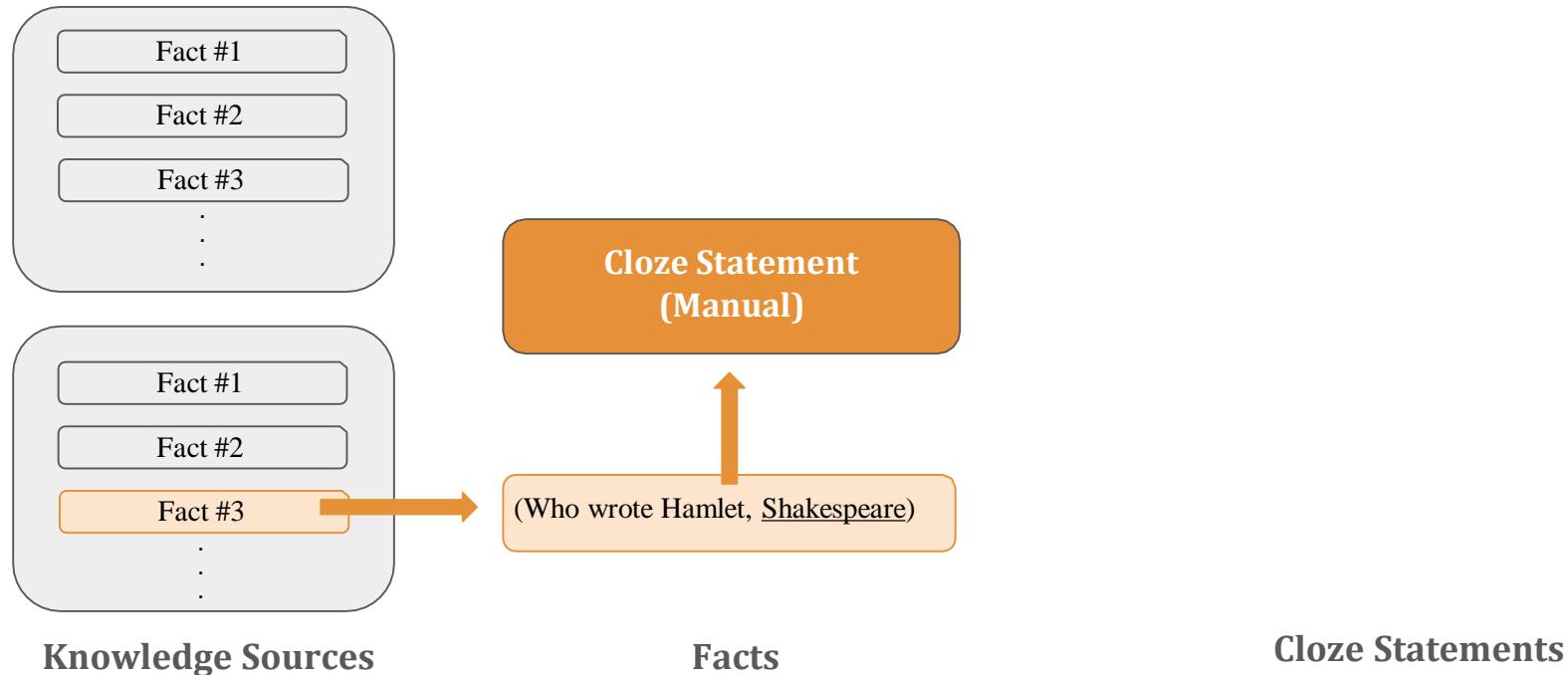
# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



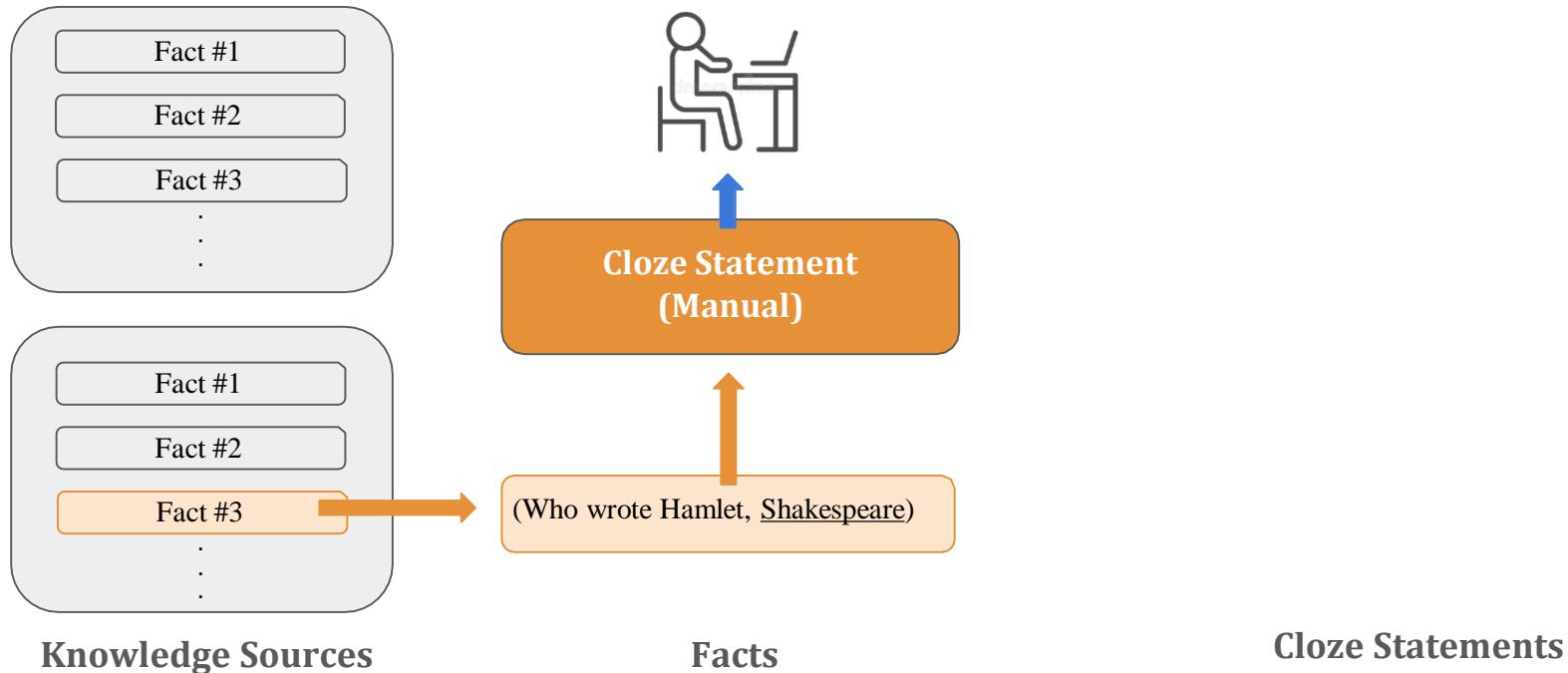
# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



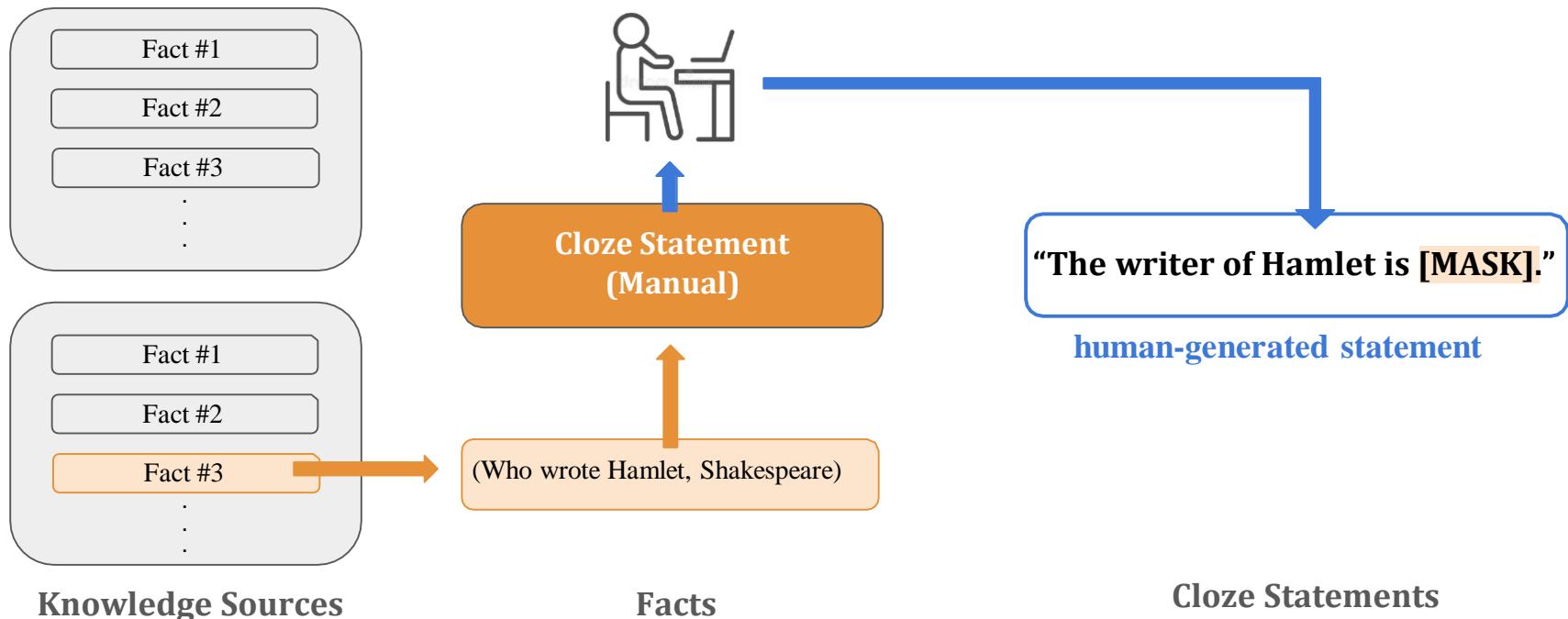
# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



# Architecture of the LAMA probe

## Step 3: Create cloze statements, either manually or via templates



More discussions on LAMA

# Any drawbacks?

**Overall pipeline of the LAMA Probe is in (Petroni et al., 2019)**

- Convert facts to cloze statements (either manually or using templates)
- Ask LM to rank candidate vocabulary and see if ground truth is in top  $k$  rank

**Can you think of any drawbacks of the probes?**

- Answers must be single-token
- Relies on manual templates
- Questions are constrained to very specific and simple types of questions

# Data leakage: **train-test overlap**

- [Testing] Many of the knowledge sources were extracted from **Wikipedia**
- [Training] However, pre-training corpora for language models almost always contain data from **Wikipedia**...
- How much of the amazing knowledge retrieval is due to **train-test overlap** in the knowledge probing benchmarks?

# Train-test overlap is responsible for LM's ability to do knowledge retrieval! ([Lewis et al., 2020](#))

Model	Open Natural Questions				TriviaQA				WebQuestions				
	Total	Question Overlap	Answer Overlap Only	No Overlap	Total	Question Overlap	Answer Overlap Only	No Overlap	Total	Question Overlap	Answer Overlap Only	No Overlap	
Open book	RAG	44.5	70.7	34.9	24.8	56.8	82.7	54.7	29.2	45.5	81.0	45.8	21.1
	DPR	41.3	69.4	34.6	19.3	57.9	80.4	59.6	31.6	42.4	74.1	39.8	22.2
	FID	51.4	71.3	48.3	34.5	67.6	87.5	66.9	42.8	-	-	-	-
Closed book	T5-11B+SSM	36.6	77.2	22.2	9.4	-	-	-	-	44.7	82.1	44.5	22.0
	BART	26.5	67.6	10.2	0.8	26.7	67.3	16.3	0.8	27.4	71.5	20.7	1.6
Nearest Neighbor	Dense TF-IDF	26.7	69.4	7.0	0.0	28.9	81.5	11.2	0.0	26.4	78.8	17.1	0.0
		22.2	56.8	4.1	0.0	23.5	68.8	5.1	0.0	19.4	63.9	8.7	0.0

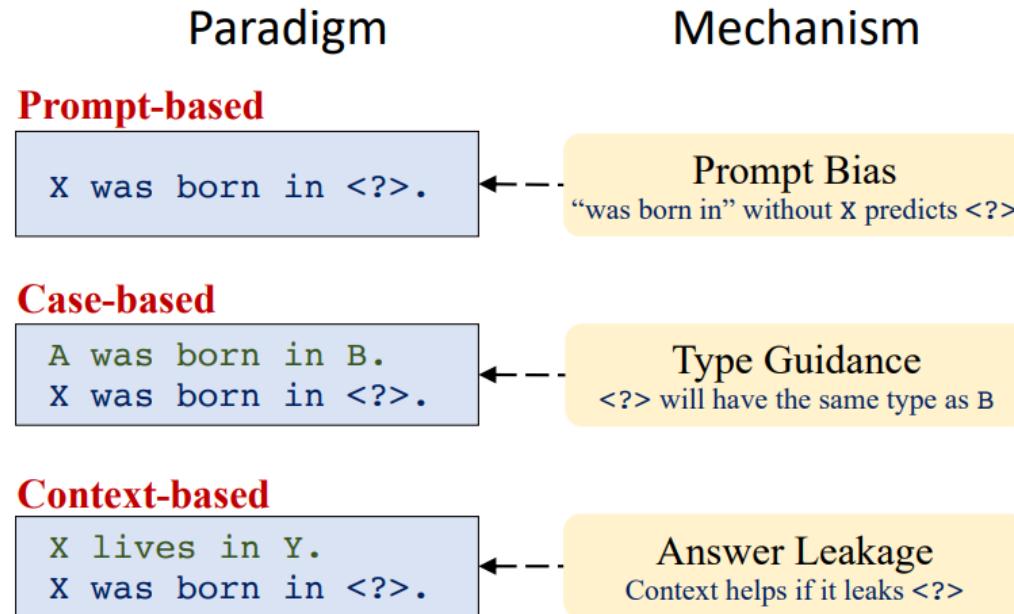
When there is **question overlap**, both open and closed-book LMs perform well

# Train-test overlap is responsible for LM's ability to do knowledge retrieval! ([Lewis et al., 2020](#))

Model	Open Natural Questions				TriviaQA				WebQuestions			
	Total	Question Overlap	Answer Overlap Only	No Overlap	Total	Question Overlap	Answer Overlap Only	No Overlap	Total	Question Overlap	Answer Overlap Only	No Overlap
Open book	RAG	44.5	70.7	34.9	24.8	56.8	82.7	54.7	29.2	45.5	81.0	45.8
	DPR	41.3	69.4	23.6	19.3	57.9	80.4	51.6	31.6	42.4	74.1	29.8
	FID	51.4	71.3	-	34.5	67.6	87.5	-	42.8	-	-	-
Closed book	T5-11B+SSM	36.6	77.2	21.2	9.4	-	-	-	44.7	82.1	44.5	22.0
	BART	26.5	67.6	10.2	0.8	26.7	67.3	16.3	0.8	27.4	71.5	20.7
Nearest Neighbor	Dense TF-IDF	26.7	69.4	7.0	0.0	28.9	81.5	11.2	0.0	26.4	78.8	17.1
		22.2	56.8	4.1	0.0	23.5	68.8	5.1	0.0	19.4	63.9	8.7
		-	-	-	-	-	-	-	-	-	-	-

But with no question or answer overlap, performance drops sharply!

# Revising LAMA – underlying mechanisms



# Revising LAMA – Reporting Bias

---

It is uninterested to say one is  
**thinking or breathing.**  
But something related to  
**murders** seems interesting to  
share

Action	Actual Frequency for Lifetime (Source)
thinking	1,433,355,000 ( <a href="#">50,000 per day</a> )
breathing	660,489,984 ( <a href="#">23,040 per day</a> )
blinking	344,005,200 ( <a href="#">12,000 per day</a> )
eating	86001.3: 3 times per day
sleeping	28667.1: 1 time per day
working	20420.4: 5 times a week
exercising	8168.16: 2-3 times a week
getting married	1.66: 0-3 times per life
getting divorced	1: 0-2 times per life
being born	1
being named	1
dying	1
being abused	0.5 ( <a href="#">source</a> )
being injured	0.1263 ( <a href="#">Episodes per 1,000 population: 126.3</a> )
being raped	0.01 ( <a href="#">18.3% of women (50.8% of population) and 1.4% of men (49.2% of population)</a> )
being killed	$4.01 \times 10^{-2}$ ( <a href="#">murder + 1 out 28 in accident</a> )
being arrested	0.031526 ( <a href="#">3,152.6 arrests per 100,000</a> )
being adopted	0.021 ( <a href="#">7 million out of 328.2</a> )
being murdered	$4.37 \times 10^{-3}$ ( <a href="#">1 in 229 deaths</a> )
being abandoned	0.000175 ( <a href="#">7000 each year, out of 4M births</a> )

---

# Revising LAMA – Reporting Bias

	BERT	RoBERTa	GPT-2	BERT	RoBERTa	GPT-2
The person ____.	wins (11.4)	said (5.8)	let (4.3)	killed (7.5)	gone (6.3)	let (4.3)
	died (11.4)	responds (4.0)	see (3.9)	married (6.6)	deceased (3.8)	see (3.9)
	dies (10.6)	replied (3.4)	make (2.4)	dying (4.2)	arrested (2.9)	make (2.4)
	won (7.8)	dies (3.3)	get (2.1)	deceased (3.8)	missing (2.5)	get (2.1)
	lost (3.5)	died (2.9)	look (2.1)	eliminated (2.6)	responding (1.9)	look (2.1)
	said (2.4)	responded (2.5)	take (1.2)	retired (2.2)	involved (1.9)	take (1.2)
	speaks (1.9)	says (2.4)	set (1.2)	lost (2.0)	reading (1.9)	set (1.2)
	answered (1.6)	replies (2.2)	give (1.1)	arrested (2.0)	dying (1.9)	give (1.1)
	replied (1.3)	asked (2.1)	using (1.1)	elected (1.5)	confused (1.5)	using (1.1)
	loses (1.3)	commented (2.1)	go (1.1)	disabled (1.5)	reporting (1.5)	go (1.1)

Table 1: Top LM predictions for actions performed by people along with their scores (percents).

Reporting bias: due to Grice’s conversational maxim of quantity (Grice et al., 1975), people rarely state the obvious, thus many trivial facts (“people breathe”) are rarely mentioned in text, while uncommon events (“people murder”) are reported disproportionately (Gordon and Van Durme, 2013; Sorower et al., 2011).

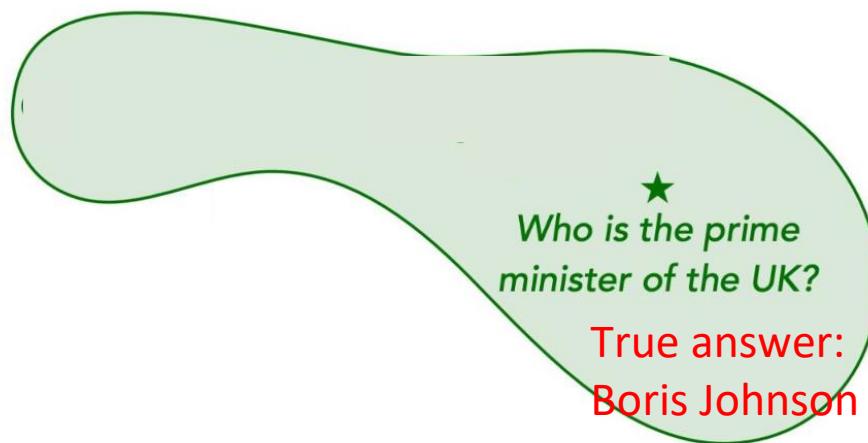
# Today's Lecture

- **Knowledge** in LLMs
  - LLMs as knowledge bases
  - Facts updating for LLMs
- **Reasoning** in LLMs
  - Why reasoning is special in LLMs
- **Prompt** Techniques for better reasoning

How to update knowledge in  
pre-trained models?

# Edit What, Exactly?

## Defining the problem



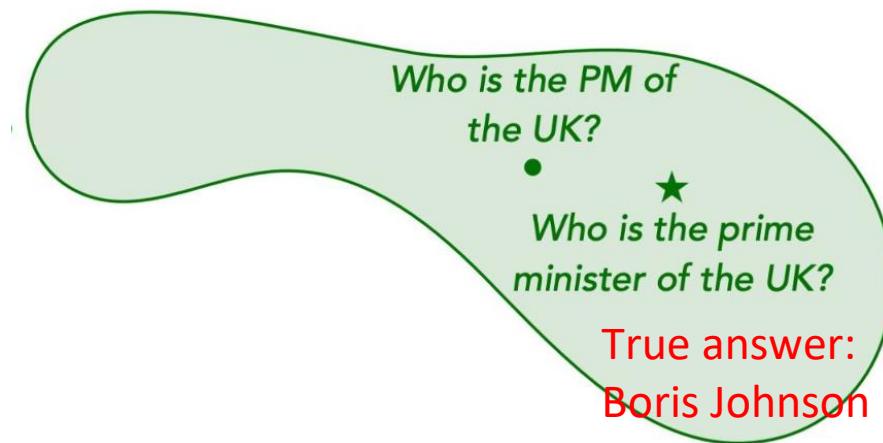
Edit example

Edit scope



# Edit What, Exactly?

## Defining the problem



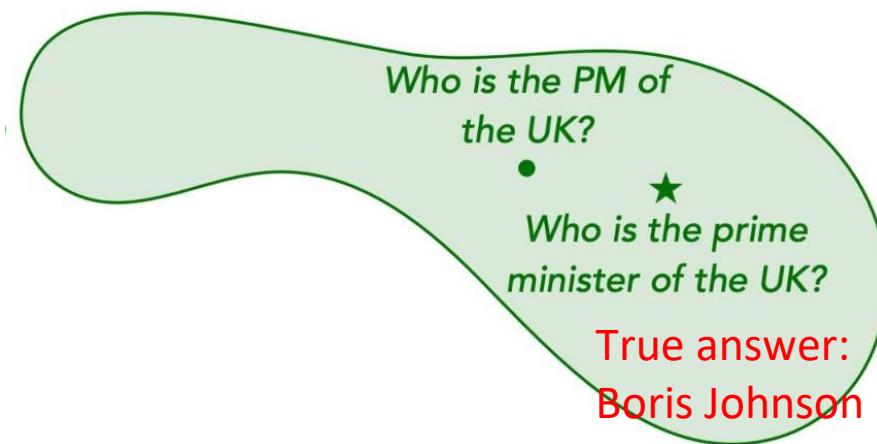
Edit example   Edit scope   In-scope



# Edit What, Exactly?

## Defining the problem

■ *Why is the sky blue?*



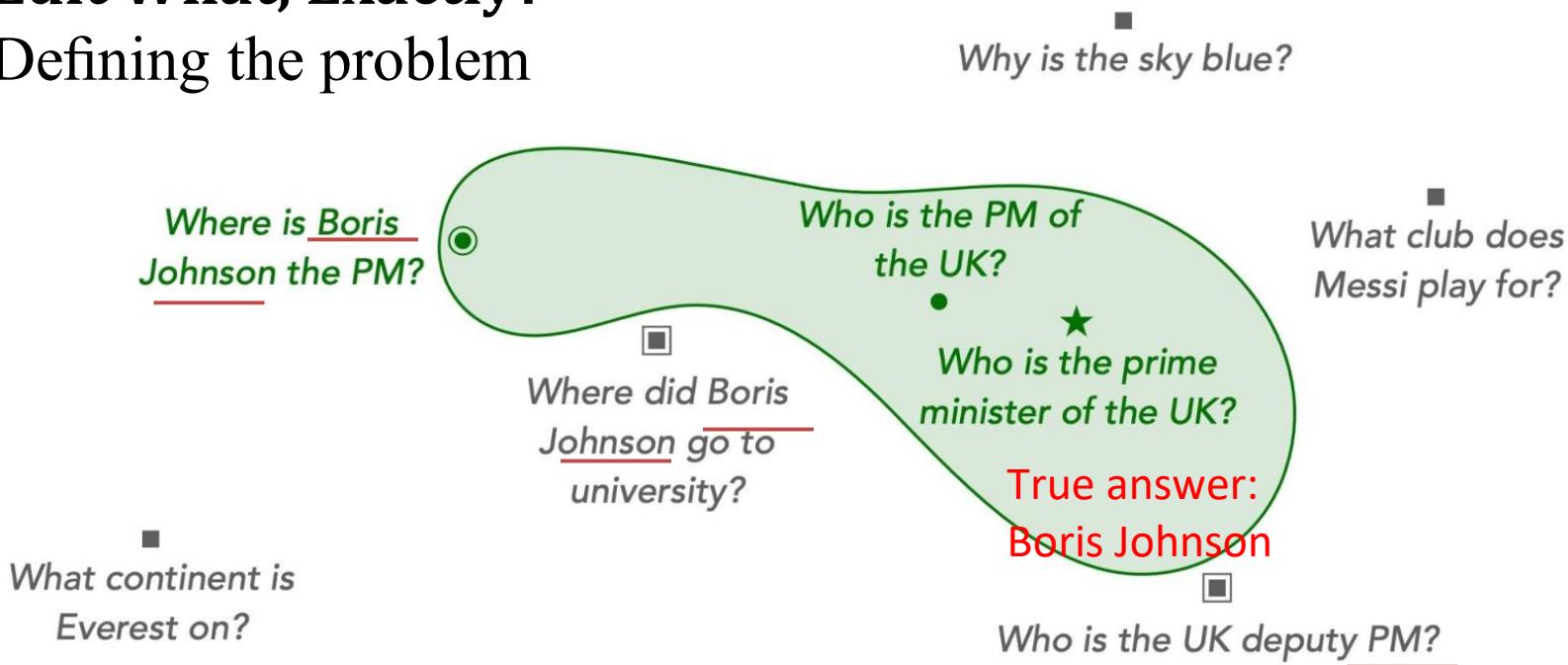
■ *What continent is Everest on?*

Edit example   Edit scope   In-scope   Out-of-scope



# Edit What, Exactly?

## Defining the problem

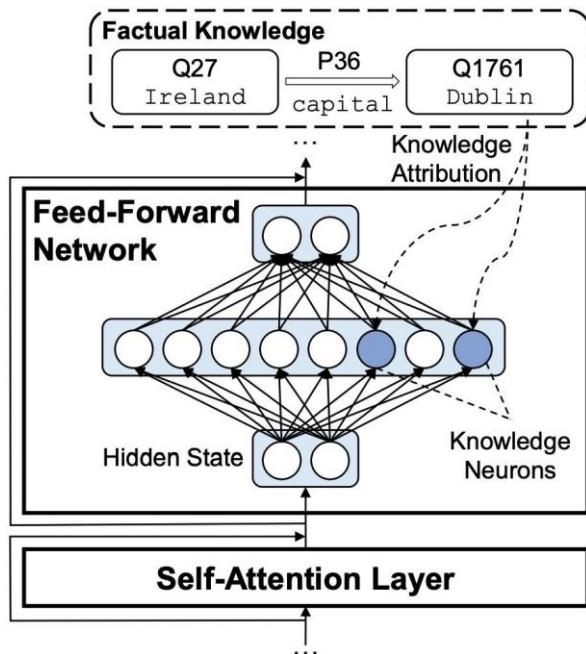


Edit example   Edit scope   In-scope   Out-of-scope   Hard in/out-of-scope



# How to edit knowledge in pre-trained models?

# Knowledge Neurons



- What is a knowledge neuron
  - **Activations** after the first feed-forward layer
- Assumption
  - Knowledge neurons are associated with factual knowledge
- Implications
  - If we can identify these neurons, we can alter them to edit (update/erase) knowledge.
  - No additional training is involved.

## Identify knowledge neurons

Given a relational fact e.g. (Mozart, BORN-IN, Austria)

1. produce  $N$  diverse prompts;
2. for each prompt, calculate the knowledge attribution scores of neurons;
3. for each prompt, retain the neurons with attribution scores greater than the attribution threshold  $T$ , obtaining the coarse set of knowledge neurons;
4. considering all the coarse sets together, retain the knowledge neurons shared by more than  $p\%$  prompts.

# Knowledge neuron editing

Knowledge neuron: activations after the first feed-forward layer

**Suppressing** the neuron: activation = 0

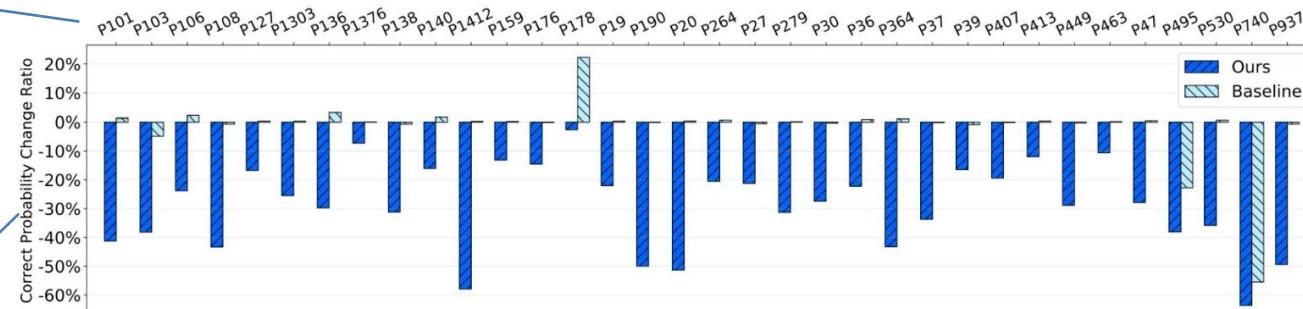
**Amplifying** the neuron: activation = 2\*activation

# Suppressing or Amplifying Knowledge Neurons

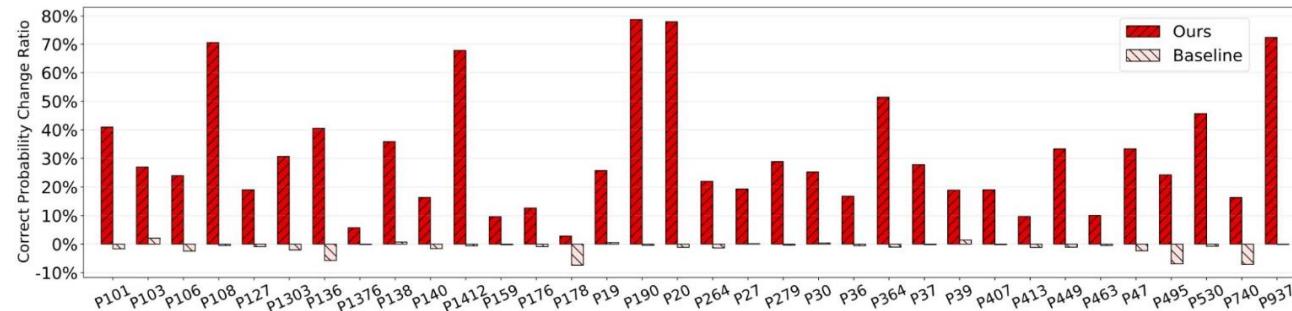
Topic:

P176:

*manufacture*



Probability  
change ratio  
for the  
correct  
answer



**Suppressing** the neurons **hurt** performance and **amplifying** neurons **increase** performance by up to 30% on average.

# Drawback

**Sensitive** to the format of the prompt collected by human

Relations	Template #1	Template #2	Template #3
P176 (manufacturer)	[X] is produced by [Y]	[X] is a product of [Y]	[Y] and its product [X]
P463 (member_of)	[X] is a member of [Y]	[X] belongs to the organization of [Y]	[X] is affiliated with [Y]
P407 (language_of_work)	[X] was written in [Y]	The language of [X] is [Y]	[X] was a [Y]-language work

Table 1: Example prompt templates of three relations in PARAREL. [X] and [Y] are the placeholders for the head and tail entities, respectively. Owing to the page width, we show only three templates for each relation. Prompt templates in PARAREL produce 253,448 knowledge-expressing prompts in total for 27,738 relational facts.

# Today's Lecture

- **Knowledge** in LLMs
  - LLMs as knowledge bases
  - Facts updating for LLMs
- **Reasoning** in LLMs
  - Why reasoning is special in LLMs
- **Prompt** Techniques for better reasoning

# Human Intelligence vs. Traditional machine learning? (Hint: reasoning)

Humans	Traditional machine learning
Learn from only a few examples	Large amounts of labeled data
Can explain rationale for decisions	Black box
Out-of-distribution generalization	No

Attempts to fill the gap in the past decades

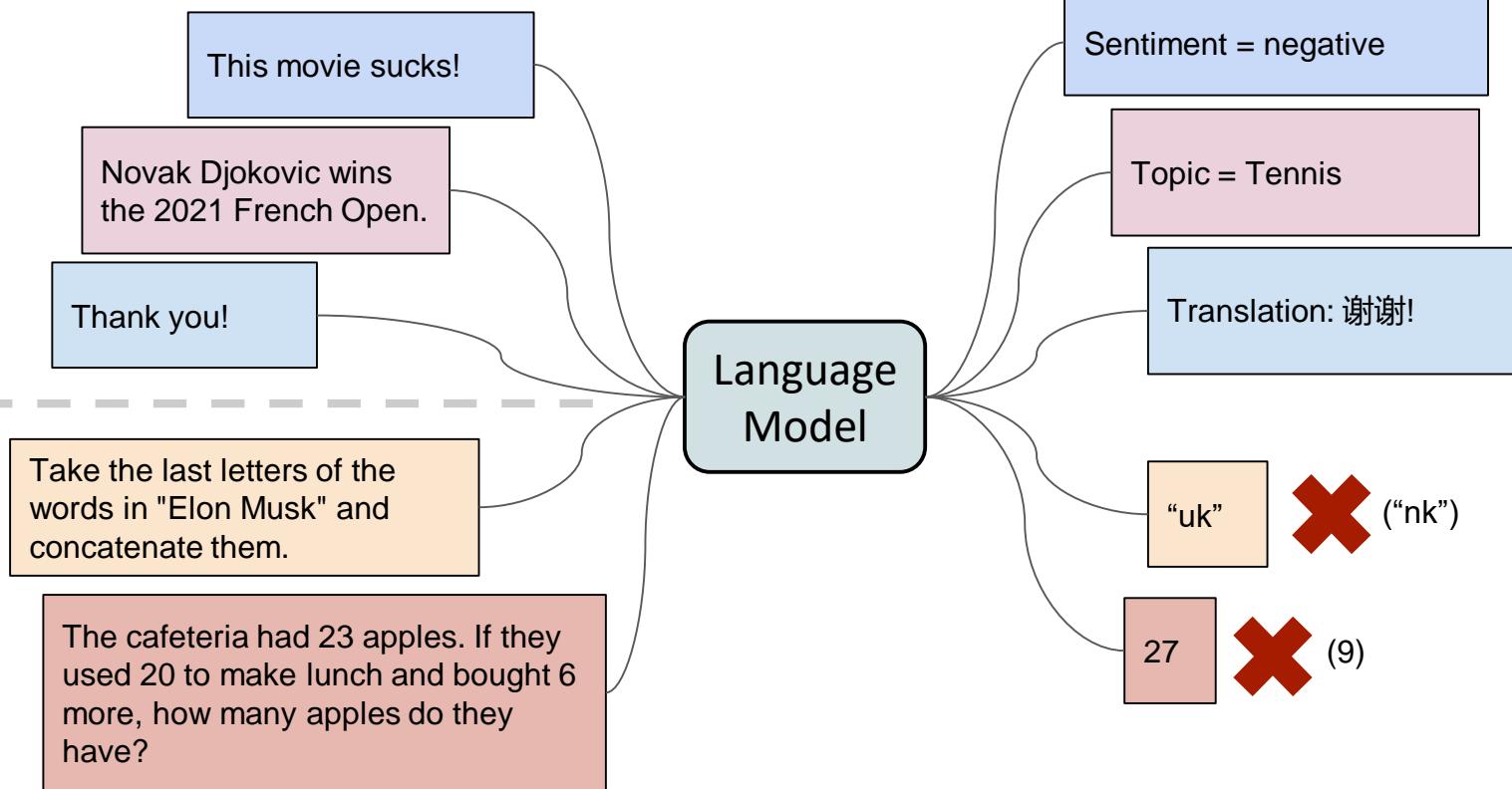
- Bayesian machine learning, kernel machines, nonparametric, sparsity, large-margin, semi-supervised learning, manifold learning, graph-based learning, transductive learning, meta learning, transfer learning, active learning, domain adaptation, structural learning, low-rank, ...



*Teaching language models to reason (Denny Zhou), 2023.*

# Multi-step reasoning is hard for language models

“type 1”



# What do language models learn from next-word prediction?

<i>Grammar</i>	In my free time, I like to { <u>run</u> , <b>banana</b> }
<i>Lexical semantics</i>	I went to the zoo to see giraffes, lions, and { <u>zebras</u> , <b>spoon</b> }
<i>World knowledge</i>	The capital of Denmark is { <u>Copenhagen</u> , <b>London</b> }
<i>Sentiment analysis</i>	Movie review: I was engaged and on the edge of my seat the whole time. The movie was { <u>good</u> , <b>bad</b> }
<i>Harder sentiment analysis</i>	Movie review: Overall, the value I got from the two hours watching it was the sum total of the popcorn and the drink. The movie was { <u>bad</u> , <b>good</b> }
<i>Translation</i>	The word for “pretty” in Spanish is { <u>bonita</u> , <b>hola</b> }
<i>Spatial reasoning</i>	[...] Iroh went into the kitchen to make some tea. Standing next to Iroh, Zuko pondered his destiny. Zuko left the { <u>kitchen</u> , <b>store</b> }
<i>Math question</i>	First grade arithmetic exam: $3 + 8 + 4 =$ { <u>15</u> , <b>11</b> }

[thousands (millions?) more]

Extreme multi-task learning!

# What can't language models learn from next-word prediction?

<i>Current world knowledge</i>	The stock price of APPL on March 1st, 2023 is {???
<i>Arbitrarily long arithmetic</i>	$36382894730 + 238302849204 = \{???$
<i>Many-step reasoning</i>	Take the nineteenth digit of Pi and multiply it by the e to the fourth power. The resulting ones-digit of the resulting number is {???
<i>Predict the future</i>	The winner of the FIFA world cup in 2026 is {???
<i>Information not in the training data</i>	Jason Wei's favorite color is {???
<i>Extremely long inputs</i>	[2,000 page Harry Potter fan-fiction] What happened after Harry opened the chest for the second time? {???

# Jason Wei's rule of thumb (经验法则)

language models can do (with decent accuracy)  
most text tasks that **an average human can do in 1 minute.**



2018

...
Protein discovery
Clinical diagnosis
Play chess well
High-level planning
Abstract reasoning
Simple math
Commonsense reasoning
Know world knowledge
Translation
Sentiment analysis
Generate coherent text
Be grammatically correct

Today (2023)

...
Protein discovery
Clinical diagnosis
Play chess well
High-level planning
Abstract reasoning
Simple math
Commonsense reasoning
Know world knowledge
Translation
Sentiment analysis
Generate coherent text
Be grammatically correct

Future ...?

...
(?) Protein discovery
(?) Clinical diagnosis
(?) Play chess well
(?) High-level planning
(?) Abstract reasoning
Simple math
Commonsense reasoning
Know world knowledge
Translation
Sentiment analysis
Generate coherent text
Be grammatically correct

# Hard Language Tasks: Reasoning

# Reasoning Problems

Q: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?  
A: The answer is **5**

## Mathematical Reasoning

Q: Take the last letters of the words in "Elon Musk" and concatenate them  
A: The answer is **nk**.

## Symbolic Reasoning

Q: What home entertainment equipment requires cable?  
Answer Choices: (a) radio shack (b) substation  
(c) television (d) cabinet  
A: The answer is **television**.

## Commonsense Reasoning

Q: Wolves are afraid of mice. Sheep are afraid of wolves. Emily is a wolf. What is Emily afraid of?  
A: The answer is **mice**.

## Logical Reasoning

# What we are doing: Mathematical modeling

## #### 模型背景

物资调拨模型关注于如何高效、成本效益地将物资从供应地点分配到需求地点。在供应链管理、紧急物资分配等场景中，合理的物资调拨能够确保物资及时供应，同时降低总体成本。

## #### 模型描述

假设有一组供应点集合 $S$ 和一组需求点集合 $D$ 。每个供应点 $i \in S$ 具有供应能力 $a_i$ ，每个需求点 $j \in D$ 有需求量 $b_j$ 。从供应点 $i$ 到需求点 $j$ 运输单位物资的成本为 $c_{ij}$ 。模型的目标是确定从每个供应点到每个需求点的物资调拨量 $x_{ij}$ ，以满足所有需求点的需求，同时最小化总运输成本。

## #### 模型目标

1. \*\*服务水平最大化\*\*：提高物资调拨的服务水平，包括减少配送时间或提高配送的可靠性。

- 公式： $\max S = \sum_{j \in D} [\alpha_j \left(1 - \frac{\sum_{i \in S} x_{ij}}{T_j}\right)]$   
其中， $\alpha_j$ 是从供应点 $i$ 到需求点 $j$ 的运输时间， $T_j$ 是需求点 $j$ 对运输时间的最大容忍值， $\alpha_j$ 是与需求点 $j$ 的服务水平重要性相关的权重。

2. \*\*环境影响最小化\*\*：在物资调拨过程中，尽量减少对环境的负面影响，例如减少碳排放。

- 公式： $\min E = \sum_{i \in S} \sum_{j \in D} c_{ij} x_{ij}$   
其中， $E$ 是从供应点 $i$ 到需求点 $j$ 运输单位物资产生的碳排放量。

3. \*\*成本最小化\*\*：最小化从所有供应点到所有需求点的总运输成本。

- 公式： $\min Z = \sum_{i \in S} \sum_{j \in D} c_{ij} x_{ij}$

4. \*\*库存水平平衡\*\*：优化各供应点和需求点的库存水平，以减少过剩或短缺的情况，保持库存的稳定。

- 公式： $\min I = \sum_{i \in S} [x_i - L_i] + \sum_{j \in D} [x_j - L_j]$   
其中， $x_i$ 表示供应点 $i$ 的库存水平， $x_j$ 表示需求点 $j$ 的实际情况， $L_i$ 和 $L_j$ 分别表示供应点 $i$ 和需求点 $j$ 的理想库存水平。

5. \*\*多模式运输优化\*\*：考虑不同运输方式（如公路、铁路、海运等）的成本和效率，优化运输模式的选择。

- 公式： $\min M = \sum_{i \in S} \sum_{j \in D} \sum_{k \in K} c_{ijk} x_{ijk}$   
其中， $K$ 是运输模式的集合， $c_{ijk}$ 是使用运输模式 $k$ 从供应点 $i$ 到需求点 $j$ 的单位运输成本， $x_{ijk}$ 是通过运输模式 $k$ 从 $i$ 到 $j$ 的物资量。

## #### 模型约束

1. \*\*供应约束\*\*：从每个供应点发出的物资总量不能超过该点的供应能力。

- 公式： $\sum_{j \in D} x_{ij} \leq a_i$   $\quad \forall i \in S$

2. \*\*需求约束\*\*：每个需求点接收的物资总量必须满足该点的需求量。

- 公式： $\sum_{i \in S} x_{ij} = b_j$   $\quad \forall j \in D$

3. \*\*非负约束\*\*：调拨的物资量不能为负。

- 公式： $x_{ij} \geq 0$   $\quad \forall i \in S, \forall j \in D$

4. \*\*可选的库存水平优化约束\*\*：对于有库存管理需求的场景，可能需要考虑库存水平的约束，确保库存水平在安全库存和最大库存水平之间。

- 公式： $S_{i,j} \leq U_i$   $\quad \forall i \in S$

其中， $S_{i,j}$ 和 $U_i$ 分别表示供应点 $i$ 的安全库存水平和最大库存水平。

5. \*\*运输模式选择约束\*\*：确保每条运输线路只选择一种运输模式。

- 公式： $\sum_{k \in K} x_{ijk} = x_{ij}$   $\quad \forall i \in S, \forall j \in D$

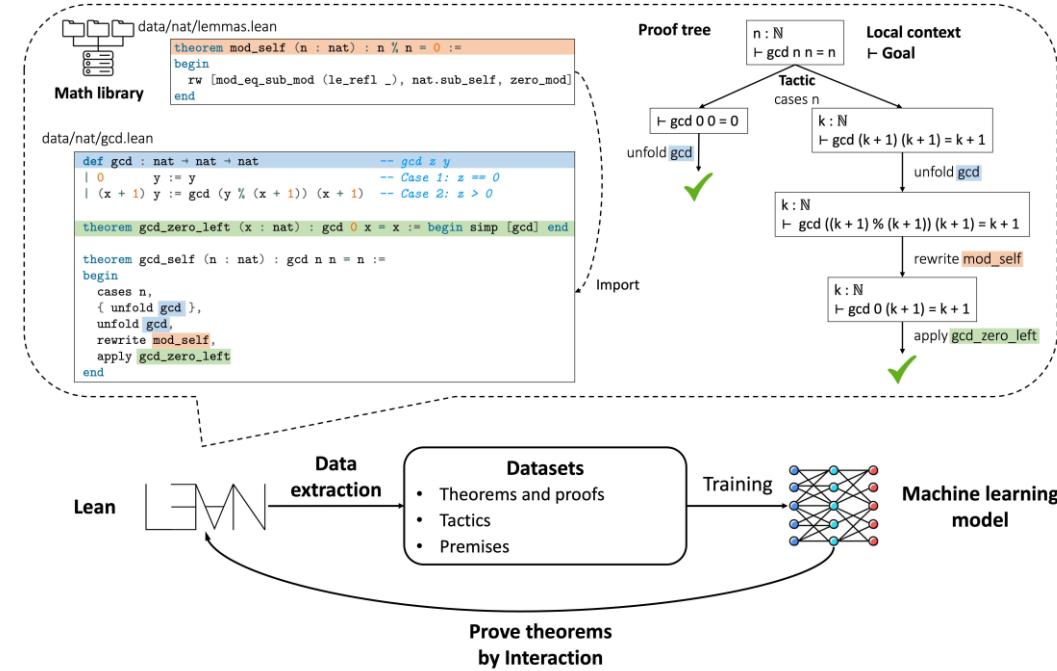
这确保了从供应点 $i$ 到需求点 $j$ 的物资量 $x_{ij}$ 通过某一种运输模式 $k$ 进行调拨。

6. \*\*时间窗约束\*\*：满足需求点的特定配送时间窗要求。

- 公式： $S_{t_{ij}, t_{ij}} \leq t_{ij} \leq t_{end}$   $\quad \forall i \in S, \forall j \in D$

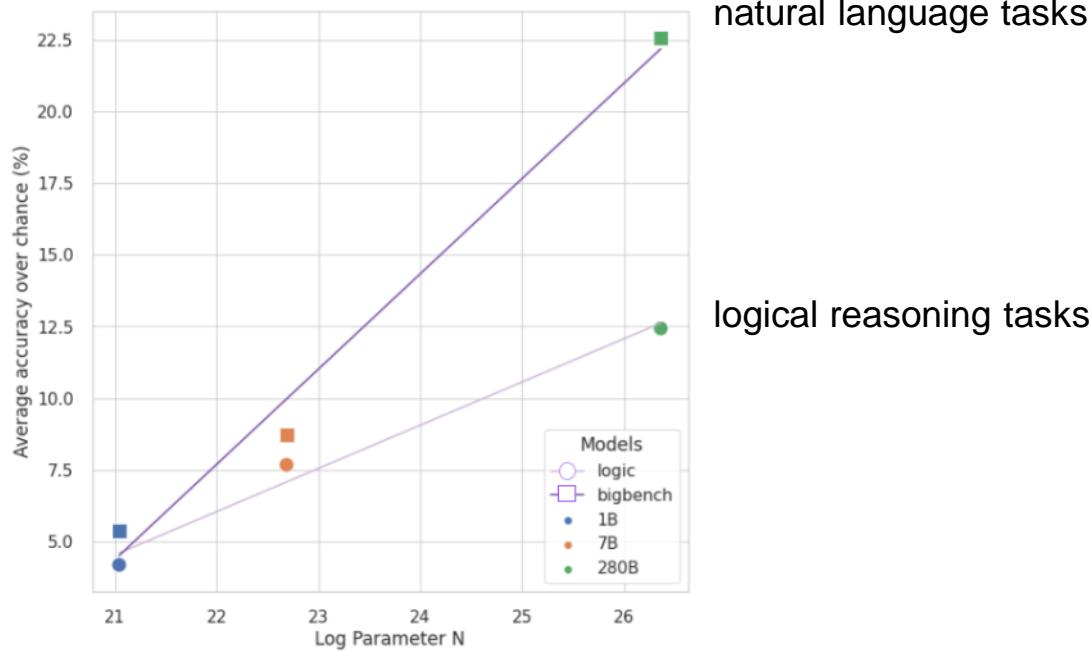
其中， $t_{ij}$ 表示从供应点 $i$ 到需求点 $j$ 的配送允许的最早开始时间和最晚结束时间， $t_{ij}$ 是实际配送时间。

# What we are doing: Automatic theorem proving



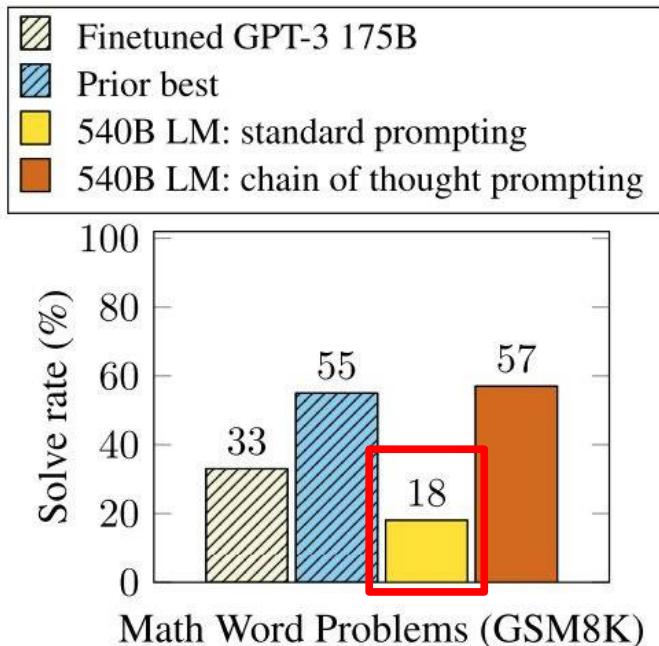
# Scaling laws are worse for logical reasoning

(Creswell et al. 2022)



# LLM fails in challenging mathematical reasoning

GSM8K (mathematical reasoning):



It is not trivial for even a 540B model to deal with the problem

# Today's Lecture

- **Knowledge** in LLMs
  - LLMs as knowledge bases
  - Facts updating for LLMs
- **Reasoning** in LLMs
  - Why reasoning is special in LLMs
- **Prompt** Techniques for better reasoning

# What is In-Context Learning?

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

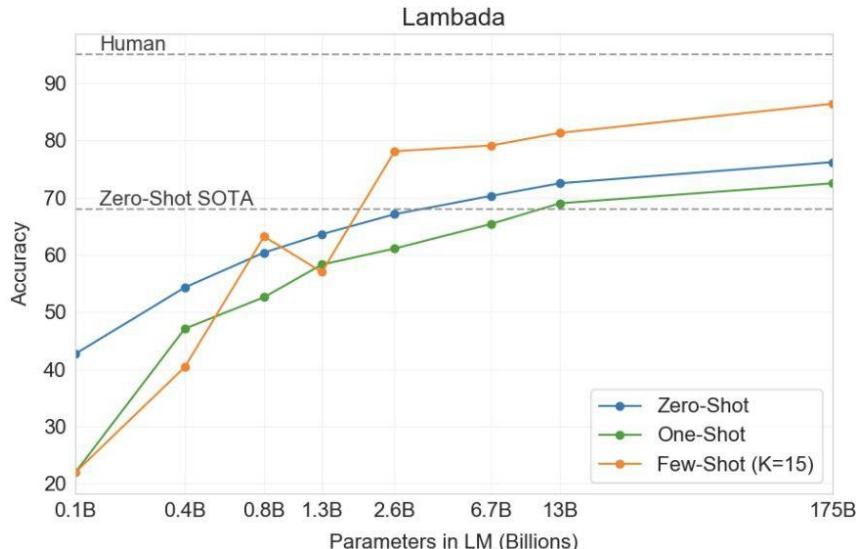
The company anticipated its operating profit to improve. // \_\_\_\_\_

LM

LM

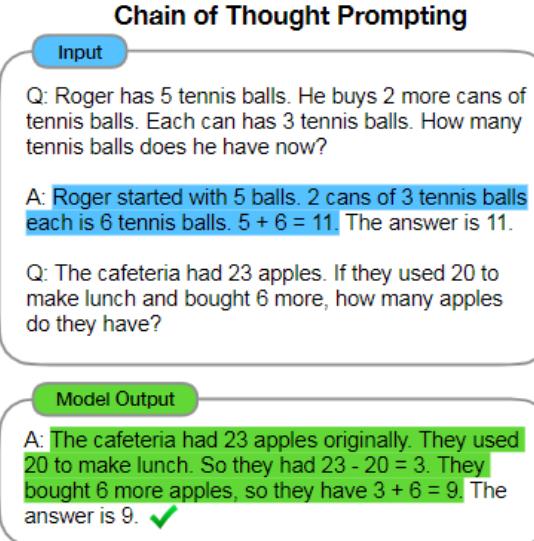
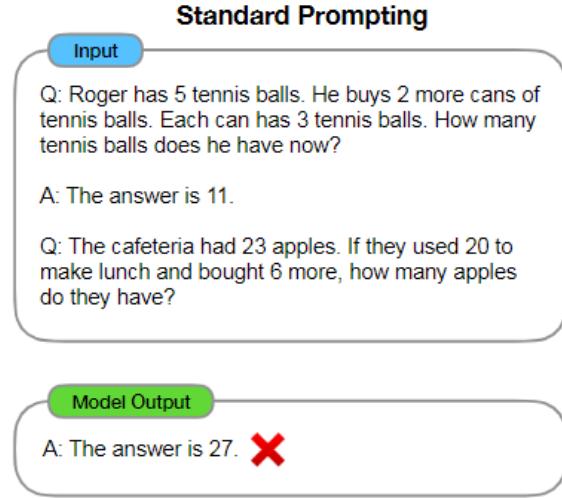
# What Can In-Context Learning Do?

- No parameter tuning need
- Only need few examples for downstream tasks
- GPT-3 improved SOTA on LAMBADA(last word prediction task) by 18%!



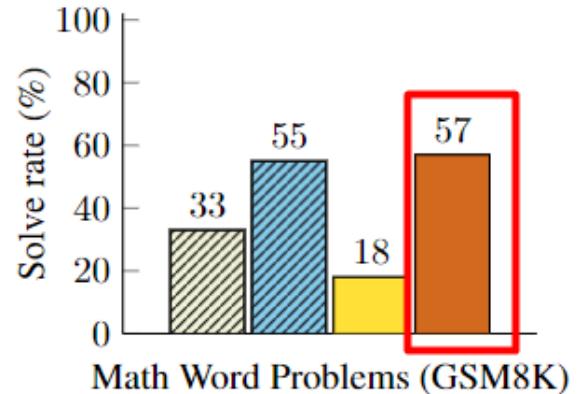
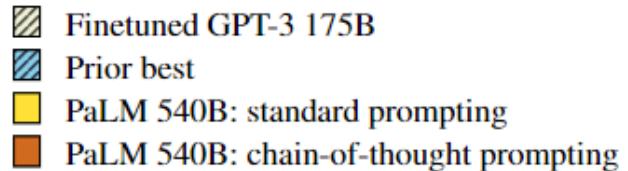
Works like magic!

# A simple way to improve the performance – Chain of Thought prompting (CoT)



<input, intermediate results, output>

- decompose into easier intermediate steps
- interpretable



# Zero-Shot CoT – Let's think step by step

## Examples

### (a) Few-shot

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The answer is 8. **X**

### (c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: The answer (arabic numerals) is

(Output) 8 **X**

### (b) Few-shot-CoT (Wei et al., 2022)

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: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are  $16 / 2 = 8$  golf balls. Half of the golf balls are blue. So there are  $8 / 2 = 4$  blue golf balls. The answer is 4. **✓**

## CoT Examples

## Step-by-step Answer

### (d) Zero-shot-CoT (KoJima et al., 2022)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: **Let's think step by step.**

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. **✓**

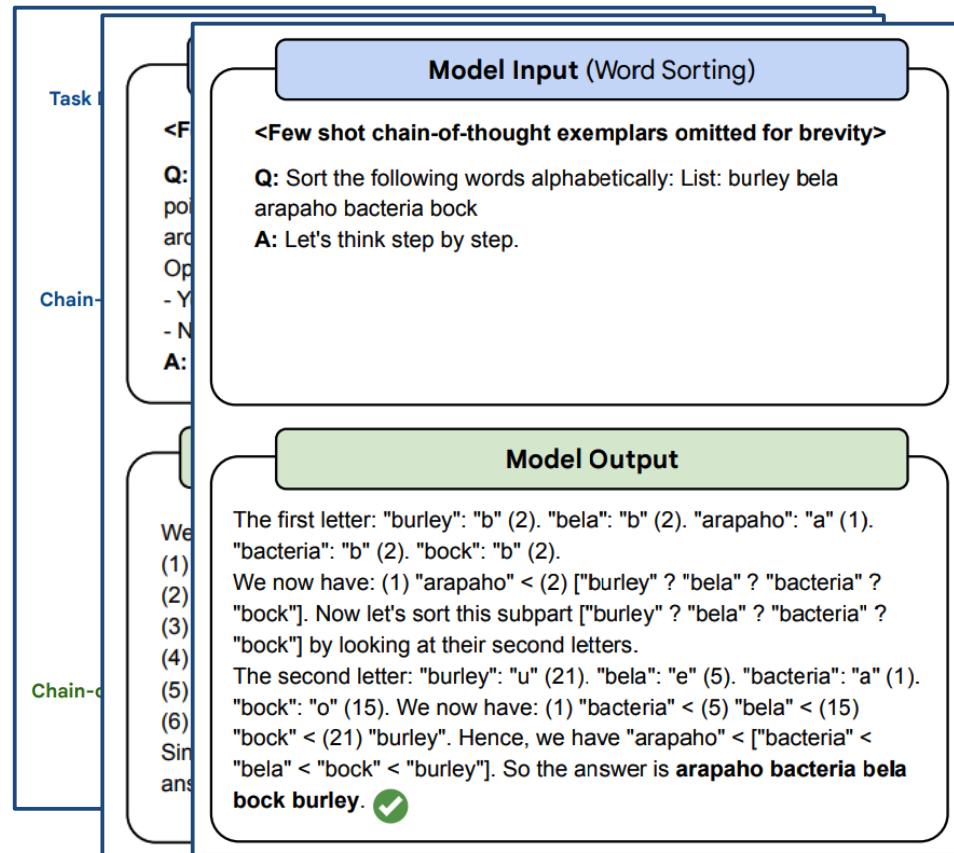
## Step-by-step Answer

# CoT on BIG-Bench: Benchmark

## BIG-Bench Hard (BBH):

- 23 challenging tasks from BIG-Bench benchmark where no model beats avg. human rater

[Challenging BIG-Bench tasks and whether chain-of-thought can solve them \(2022\).](#)



# CoT on BIG-Bench: Result summary

	BBH all (23 tasks)	# tasks above avg. human-rater	
Average human-rater	67.7	N/A	
Max human-rater	94.4	23 / 23	
Best prior BIG-Bench result	50.9	0 / 23	Model much lower than average human rater
Codex (code-davinci-002)			
- Answer-only prompting	56.6	5 / 23	Detail: better formatting (options, task description) already beats prior best
- CoT prompting	73.9 (+16.7)	17 / 23	

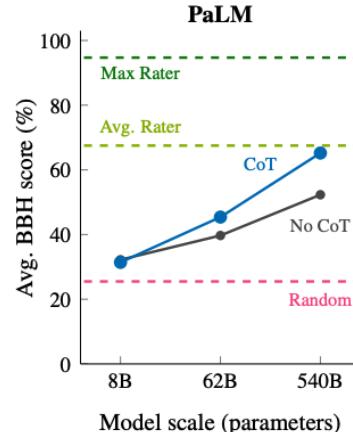
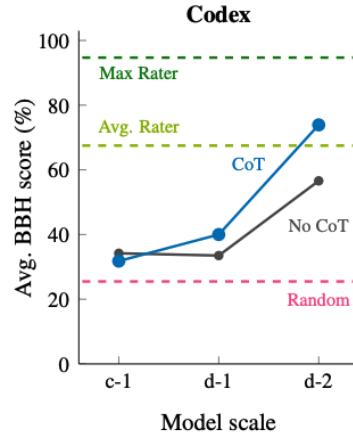
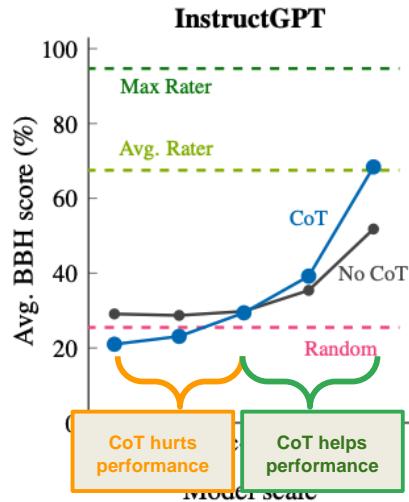
[Challenging BIG-Bench tasks and whether chain-of-thought can solve them \(2023\).](#)

CoT prompting improves by performance by +16.7%, passes avg. human on majority of tasks

# CoT on BIG-Bench: Scaling

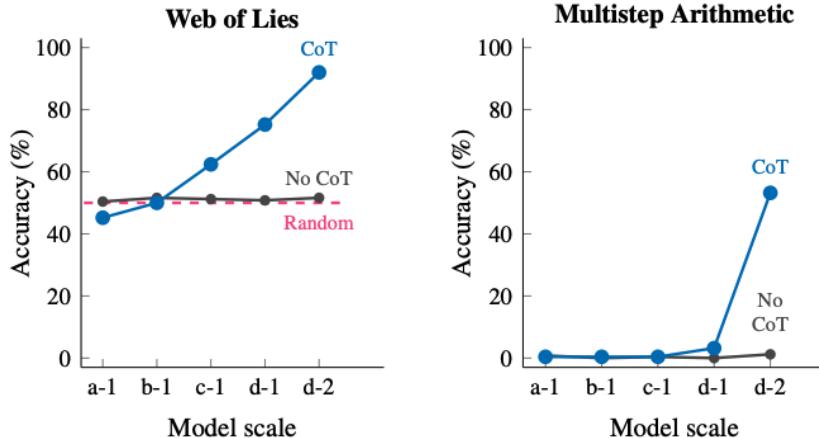
- CoT requires sufficient model scale for positive delta

[Challenging BIG-Bench tasks and whether chain-of-thought can solve them \(2023\)](#).



# CoT on BIG-Bench: Emergence

- No-CoT performance is flat, i.e., hasn't unlocked emergence (yet :))
- CoT unlocks emergent performance



[Challenging BIG-Bench tasks and whether chain-of-thought can solve them \(2023\).](#)

**Model Input (Multistep Arithmetic)**

<Few shot chain-of-thought exemplars omitted for brevity>

Q:  $((4 + 7 * 4 - -5) - (-4 - 1 - -4 - 4)) =$   
A: Let's think step by step.

**Model Output**

Let's recall that the order of operations in mathematics is as follows: (1) Parentheses, (2) exponents, (3) multiplication and division (from left to right), (4) addition and multiplication (from left to right). So, remember to always compute the expressions inside parentheses or brackets first.

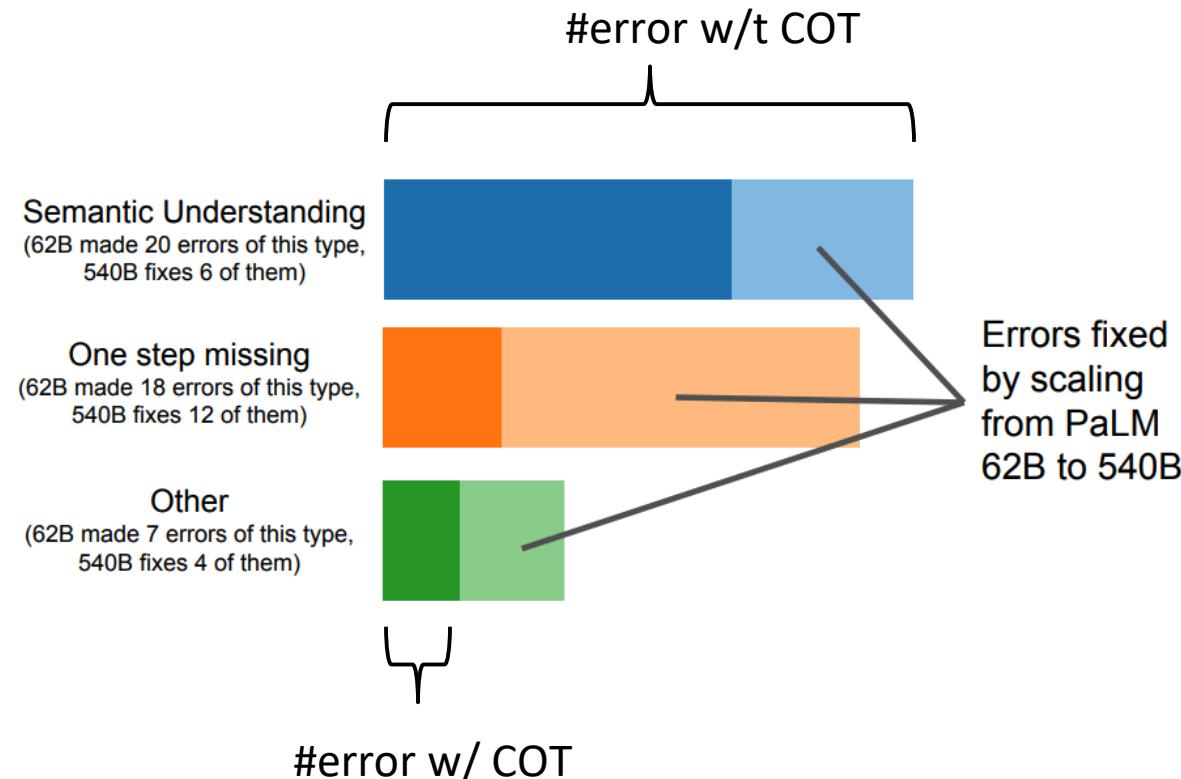
This equation can be written as "A - B", where A =  $(4 + 7 * 4 - -5)$  and B =  $(-4 - 1 - -4 - 4)$ .

Let's calculate A =  $(4 + 7 * 4 - -5) = (4 + (7 * 4) - -5) = (4 + (28) - -5) = (4 + 28 - -5) = (4 + 28 + 5) = 37$ .

Let's calculate B =  $(-4 - 1 - -4 - 4) = ((-4 - 1) - -4 - 4) = ((-5) - -4 - 4) = ((-5 - -4) - 4) = ((-5 + 4) - 4) = (-1 - 4) = -5$ .

Then, the final equation is A - B = 37 - -5 = 37 + 5 = 42. So the answer is 42. ✓

# Why does scaling up improve chain-of-thought?



# Multilingual chain-of-thought prompting

Prompt the model with Bengali math problems and Bengali reasoning

Input is highly improbable (Bengali is 0.01% of pre-training data)

## Model Input

প্রশ্ন: রজারের ৫টি টেনিস বল আছে। সে আরও 2 ক্যান টেনিস বল কিনেছে। প্রতিটি ক্যানে 3টি করে টেনিস বল আছে। তার কাছে এখন কতগুলি টেনিস বল আছে?

ধাপে ধাপে উত্তর: **রজারের প্রথম ৫টি বল ছিল। ২টি ক্যানের প্রতিটিতে ৩টি টেনিস বল মানে ৬টি টেনিস বল।  $5 + 6 = 11$ ।** উত্তর হল 11।

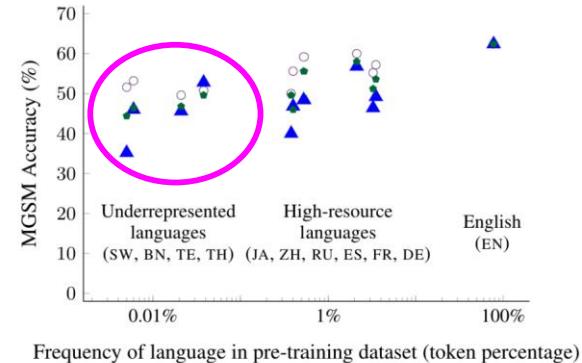
প্রশ্ন: জেনেভের হাঁসগুলি প্রতিদিন 16টি করে ডিম পাঠে। তিনি প্রতিদিন প্রাত়রাশে তিনটি করে ডিম খান এবং বন্দুদের জন্য প্রতিদিন চারটি ডিম দিয়ে মাফিন তৈরি করেন। অবশিষ্ট হাঁসের ডিমগুলি তিনি প্রতিদিন কৃষকদের বাজারে প্রতি ডিম \$2 দরে বিক্রয় করেন। তিনি কৃষকদের বাজারে প্রতিদিন কত ডলার উপর্যুক্ত করেন?

## Model Output

ধাপে ধাপে উত্তর: **প্রতিদিন 16টি ডিম পাঠে। প্রতিদিন তিনটি ডিম খান এবং চারটি ডিম দিয়ে মাফিন তৈরি করেন। তাই প্রতিদিন  $16 - 3 - 4 = 9$  টি ডিম অবশিষ্ট থাকে। প্রতিটি ডিমের মূলা \$2 হলে প্রতিদিন  $9 * 2 = 18$  ডলার উপর্যুক্ত করেন।**  $18 + 11 = 29$ ।

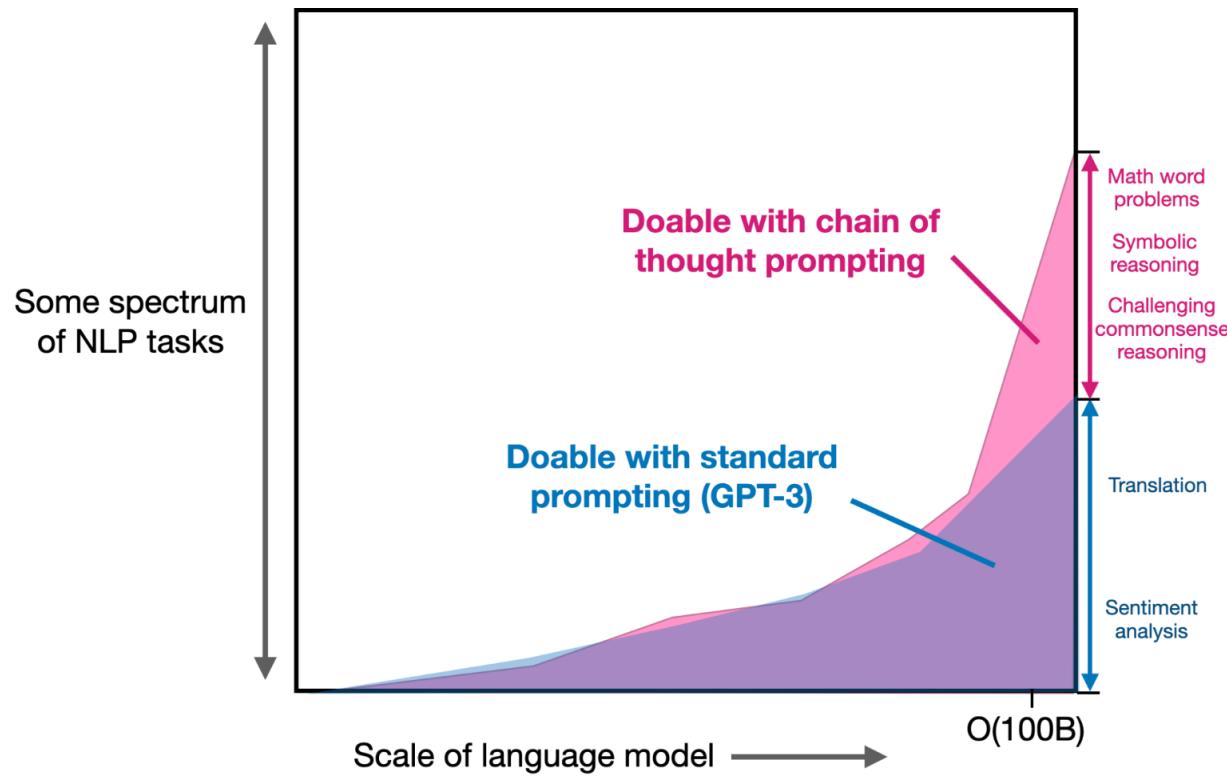
[Language models are multilingual chain-of-thought reasoners \(2022\).](#)

- Translate to English with Google Translate and solve with English intermediate steps
- ▲ Intermediate reasoning steps in the language of the question
- Intermediate reasoning steps in English



Underrepresented languages did surprisingly well, demonstrating the compositionality of the model

(model is neither multilingual nor trained to do reasoning)



# A trick for COT: Self-consistency

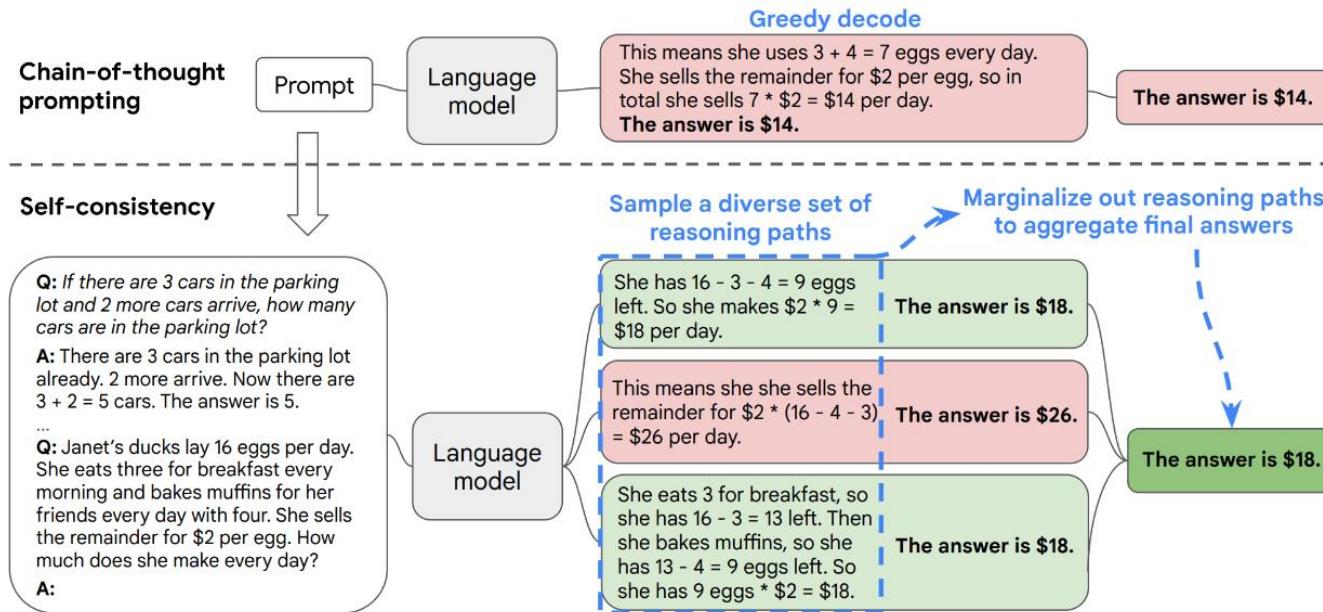
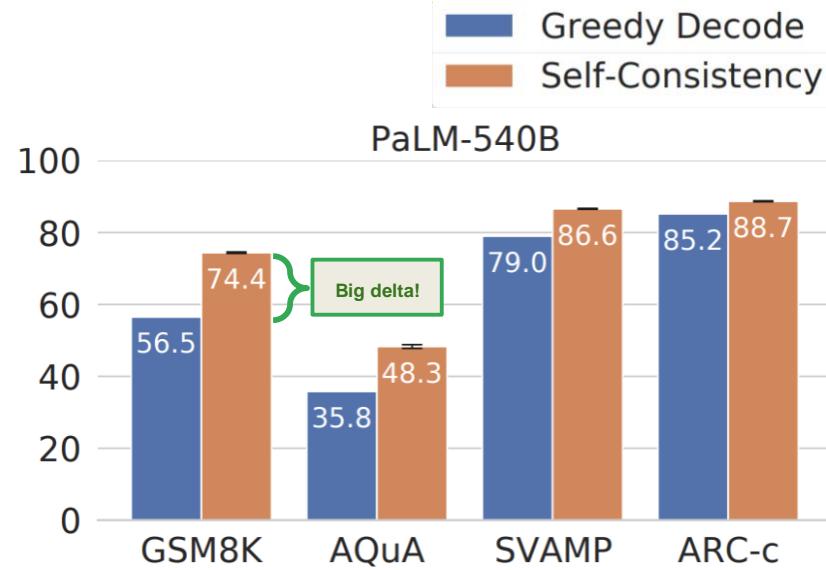


Figure 1: The self-consistency method contains three steps: (1) prompt a language model using chain-of-thought (CoT) prompting; (2) replace the “greedy decode” in CoT prompting by sampling from the language model’s decoder to generate a diverse set of reasoning paths; and (3) marginalize out the reasoning paths and aggregate by choosing the most consistent answer in the final answer set.

# Self-consistency works really well



# A surprising result on math word problems

**Question:** A particle moves so that it is at  $(3 \sin \frac{t}{4}, 3 \cos \frac{t}{4})$  at time  $t$ . Find the speed of the particle, measured in unit of distance per unit of time.

**Model output:** The speed is the magnitude of the velocity vector. Therefore, we must first find the velocity vector, which is the derivative of the position vector. We have

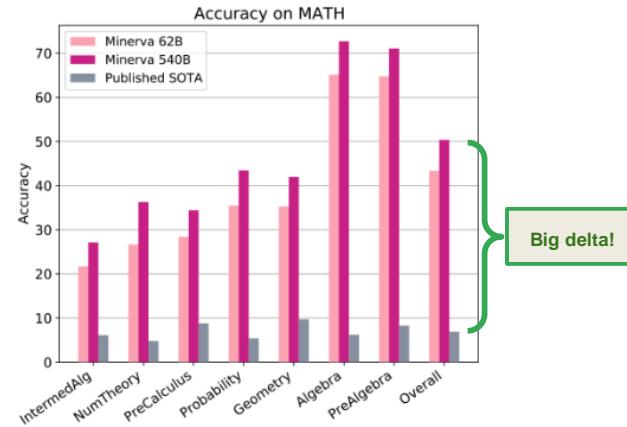
$$\mathbf{v} = \frac{d}{dt} \left( 3 \sin \frac{t}{4}, 3 \cos \frac{t}{4} \right) = 3 \left( \cos \frac{t}{4}, -\sin \frac{t}{4} \right) \frac{1}{4} = \left( \frac{3}{4} \cos \frac{t}{4}, -\frac{3}{4} \sin \frac{t}{4} \right).$$

Therefore, the magnitude of the velocity vector is

$$\begin{aligned} |\mathbf{v}| &= \sqrt{\left(\frac{3}{4} \cos \frac{t}{4}\right)^2 + \left(-\frac{3}{4} \sin \frac{t}{4}\right)^2} = \sqrt{\frac{9}{16} \cos^2 \frac{t}{4} + \frac{9}{16} \sin^2 \frac{t}{4}} \\ &= \frac{3}{4} \sqrt{\cos^2 \frac{t}{4} + \sin^2 \frac{t}{4}} = \frac{3}{4}. \end{aligned}$$

Therefore, the speed of the particle is  $\boxed{\frac{3}{4}}$ .

[Solving quantitative reasoning problems with language models \(2022\)](#).



# Chain-of-thought analysis

## **Benefits**

Expands the range of abilities for language models  
[Multi-step reasoning can now be solved!](#)

Works for any text (and image?) task  
[Every task has a chain-of-thought.](#)

No fine-tuning needed.  
[Single model, many tasks](#)

Some interpretability (can read chain-of-thought)  
[Though it's not necessarily how the model reasons](#)

## **Drawbacks**

Requires a large language model  
[Emergent ability](#)

Higher inference cost than directly answering  
[CoT can be hundreds of tokens](#)

Requires manually writing chains-of-thought in the prompts via exemplars  
[\(Some zero-shot that works for common multi-step reasoning problems\)](#)

Suggested further reading:  
[Large language models are zero-shot reasoners.](#)

# CoT is not enough

- Error propagation: one incorrect step leads to cumulative errors
- Chain structure limitation: the scope of exploration is limited
- Uncertainty: greedy decoding may not lead to a great reasoning path

Q: Calculate  $(2+3)*5$

A:

Calculate  $2+3$ , we get 6

$6*5 = 30$

The final answer is 30

Cumulative error

Q: Can 1, 2, 3, 4 get 24 in game 24?

A:

$1+2 = 3$

$3*3 = 9$

$9+4 = 13$

$13 \neq 24$

So 1,2,3,4 cannot get 24 in game 24.

Limited exploration

Q: What is  $1+2+3+\dots+6$ ?

A:

$1+2 = 3$

$3+3 = 6$

$6+4 = 10$

$10+5 = 15$

$15+6 = 21$

So  $1+2+3+4+5+6=21$ .

Correct yet not good

# Improve CoT in different phases of reasoning

- Pre-process of the reasoning task:
  - Decomposition: e.g. Least-to-most prompting
- Improvement in the reasoning phase:
  - Tool using: e.g. PoT
  - Planning: e.g. ToT
- Utilization of the reasoning result:
  - Major voting: e.g. Self-consistency
  - Verify: e.g. Verifier
  - Refine: e.g. Self-refine

# Least-to-most prompting

## Explicitly decompose into subquestions

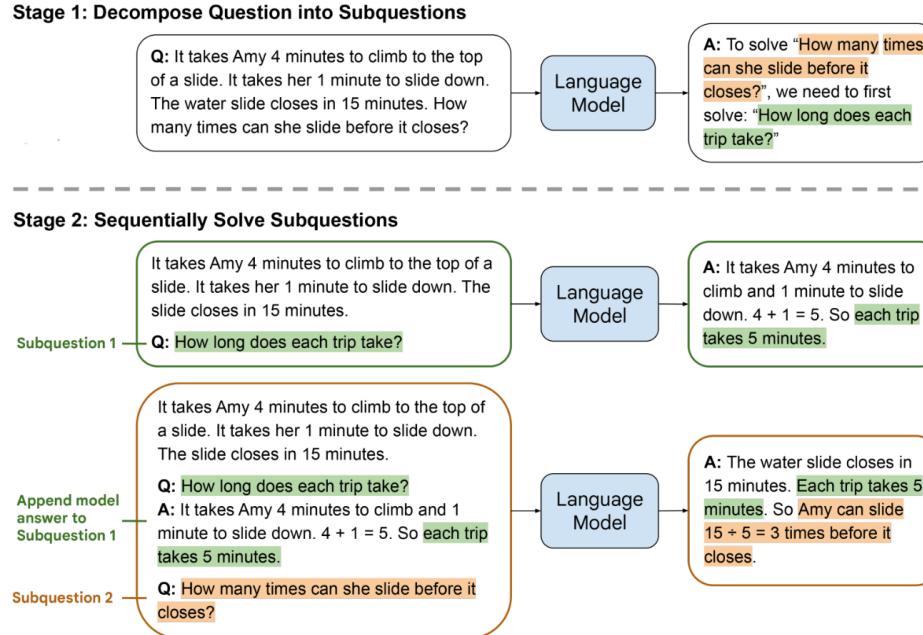


Figure 1: Least-to-most prompting solving a math word problem in two stages: (1) query the language model to decompose the problem into subproblems; (2) query the language model to sequentially solve the subproblems. The answer to the second subproblem is built on the answer to the first subproblem. The demonstration examples for each stage's prompt are omitted in this illustration.

# Program of Thoughts(PoT)

Output Python programs and call Python interpreter to calculate the answers

Question: In Fibonacci sequence, it follows the rule that each number is equal to the sum of the preceding two numbers. Assuming the first two numbers are 0 and 1, what is the 50th number in Fibonacci sequence?

The first number is 0, the second number is 1, therefore, the third number is  $0+1=1$ . The fourth number is  $1+1=2$ . The fifth number is  $1+2=3$ . The sixth number is  $2+3=5$ . The seventh number is  $3+5=8$ . The eighth number is  $5+8=13$ .  
..... (Skip 1000 tokens)  
The 50th number is 32,432,268,459.

Cot

```
length_of_fibonacci_sequence = 50  
fibonacci_sequence = np.zeros(length_of_ )  
fibonacci_sequence[0] = 0  
fibonacci_sequence[1] = 1  
for i in range(3, length_of_fibonacci_sequence):  
    fibonacci_sequence[i] = fibonacci_sequence[i-1] +  
        fibonacci_sequence[i-2]  
ans = fibonacci_sequence[-1]
```

PoT

32,432,268,459



12,586,269,025



Question: Ketty saves 20000 dollars to the bank. After three years, the sum with compound interest rate is 1000 dollars more than the sum with simple interest rate. What is the interest rate of the bank?

Assuming the interest rate is  $x$ . The sum after two years with simple interest rate is  $20000 + x * 20000 * 3 = 20000 + 60000x$ . The sum after two years with compound interest rate is  $20000 * (1 + x)^3 = 200000 + 60000 * x + 60000x^2 + 20000x^3$ . The difference can be written as  $60000x^2 + 20000x^3 = 1000$ . In order to solve  $x$ , we can use the quadratic formula.  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ , ...,  $x = (-20000 \pm 6160) / 120000$ ,  $x = -0.051333$ .

Cot

```
interest_rate = Symbol('x')  
sum_in_two_years_with_simple_interest = 20000 +  
    interest_rate * 20000 * 3  
sum_in_two_years_with_compound_interest = 20000 * (1 +  
    interest_rate)**3  
# Since compound interest is 1000 more than simple interest.  
ans = solve(sum_after_in_yeras_with_compound_interest -  
    sum_after_two_years_in_compound_interest - 1000,  
    interest_rate)
```

PoT

-0.051333



x = 0.24814

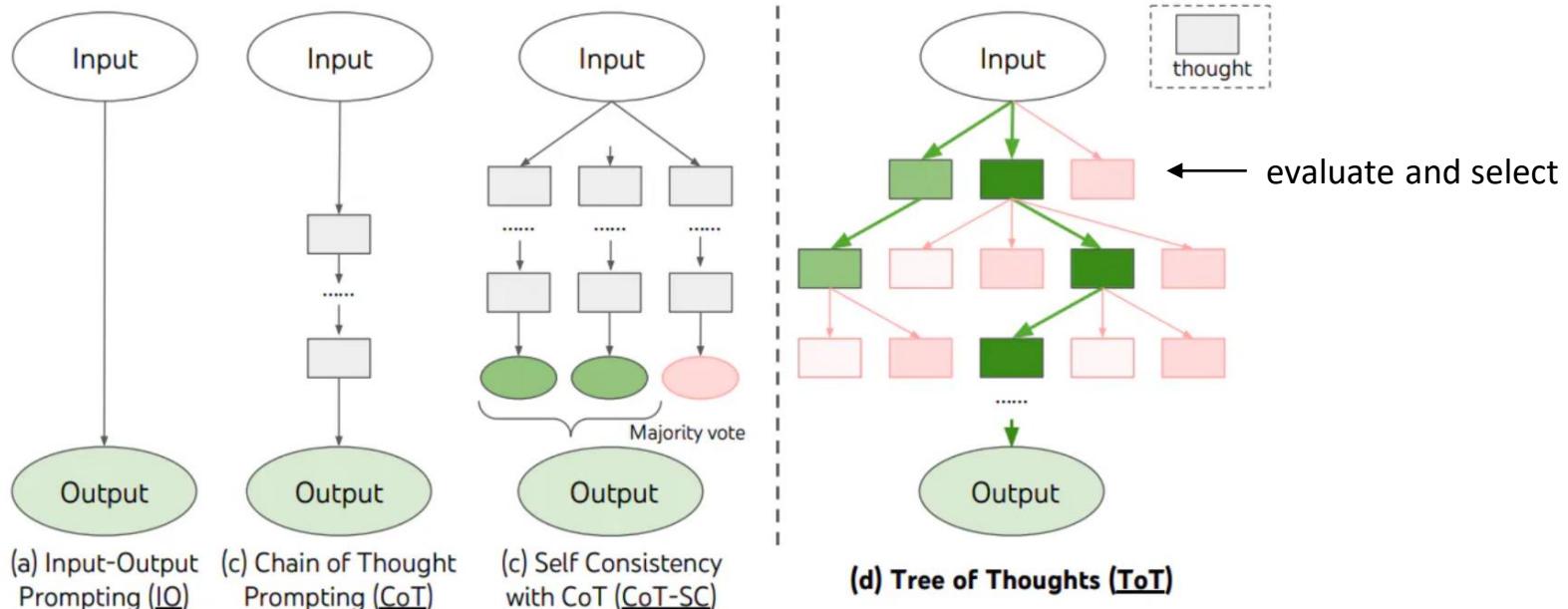


Figure 1: Comparison between Chain of Thoughts and Program of Thoughts.

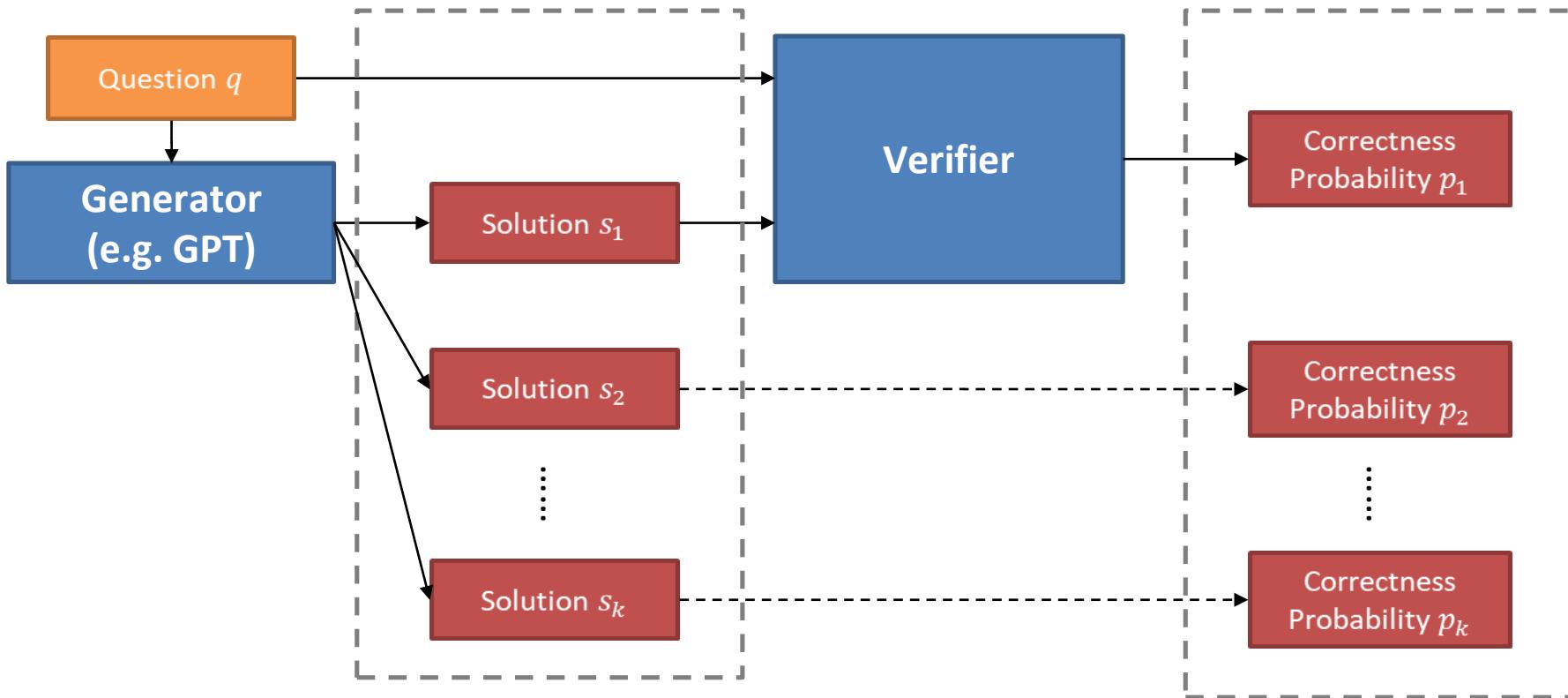
# Tree of Thoughts(ToT)

Explore over units of text that serve as intermediate steps

The ToT framework is illustrated below:



# Verifier in COT



# Exceed GPT 4 using our OVM in GSM-8K

1	GPT-4 Code Interpreter (CSV, K≤5)	97.0	x	Solving Challenging Math Word Problems Using GPT-4 Code Interpreter with Code-based Self-Verification	2023	
2	GPT-4 (Model Selection, SC K=15)	96.8	x	Automatic Model Selection with Large Language Models for Reasoning	2023	
3	GPT-4 (PHP, SC K=40)	96.5	x	Progressive-Hint Prompting Improves Reasoning in Large Language Models	2023	
4	GPT-4 (Model Selection, SC K=5)	96.5	x	Automatic Model Selection with Large Language Models for Reasoning	2023	
5	SFT-Mistral-7B (Metamath, OVM, Smart Ensemble)	96.4	7	✓		2024
6	SFT-Mistral-7B (AugData + ovm + ensemble)	95.9	7	✓		2024
7	GPT-4 (PHP)	95.5	x	Progressive-Hint Prompting Improves Reasoning in Large Language Models	2023	
8	MindOpt Copilot Mistral-7B (MetaMath, OVM, BS, Ensemble)	95.1	7	✓		2024
9	Claude 3 Opus (0-shot chain-of-thought)	95	x	The Claude 3 Model Family: Opus, Sonnet, Haiku	2024	
10	Gemini Ultra (Maj1@32)	94.4	x	Gemini: A Family of Highly Capable Multimodal Models	2023	
11	SFT-Mistral-7B (Metamath + ovm + ensemble)	94.13	7	✓		2024
12	GPT-4 (Ask, Refine, Trust)	94.08	x	The ART of LLM Refinement: Ask, Refine, and Trust	2023	
13	Shepherd + DeepSeek-67B (SFT on MetaMATH + PRM rerank, k=256)	93.3	67	✓	Math-Shepherd: Verify and Reinforce LLMs Step-by-step without Human Annotations	2023
14	MindOpt Copilot Mistral-7B (MetaMath, OVM, Ensemble)	93.2	7	✓		2024

<https://paperswithcode.com/sota/arithmetic-reasoning-on-gsm8k>

# Self-refine

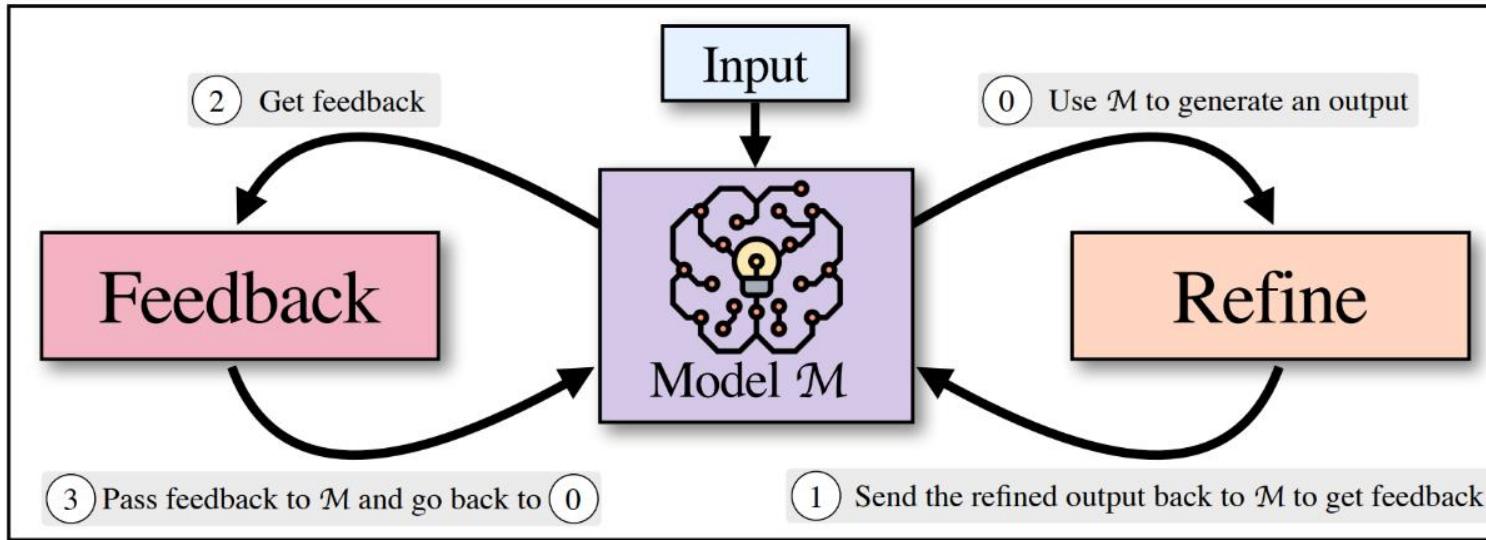


Figure 1: SELF-REFINE starts by taking an initially generated output (①), and passing it back to the same model  $M$  (②) to get feedback (③); feedback on the initial output is passed back to the model (④), to iteratively refine (①) the previously generated output. SELF-REFINE is instantiated with a powerful language model such as GPT-3.5 and does not involve human assistance.

# Should we employ all the techniques above?

- Usually CoT can perform well under many situations
- Accuracy vs **Cost**:
  - Additional techniques need more computational sources (self-consistency) or additional data processing(PoT) although these techniques can usually improve the performance.
  - the trade off depends on the real application.

# COT in Medical LLMs

ChatGPT 4 ▾



You

糖尿病人可以吃牛油果吗？先给出详细的理由再给答案。

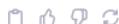


ChatGPT

牛油果是一种营养丰富的水果，含有高比例的单不饱和脂肪，这种脂肪对心脏健康有益，可以帮助降低胆固醇。对于糖尿病人来说，牛油果中的单不饱和脂肪有助于调节血糖水平，因为它们可以减缓消化过程，从而减少血糖水平的波动。此外，牛油果还含有丰富的纤维，对维持血糖水平稳定也很有帮助。

牛油果中的钾含量高，有助于预防糖尿病相关的并发症，如高血压。不过，牛油果的热量相对较高，所以糖尿病人在食用时应注意分量控制。

总的来说，糖尿病人是可以吃牛油果的，但应该将其作为平衡饮食的一部分，并注意总的热量摄入，避免过量食用。在任何饮食调整之前，最好先咨询医生或营养师的建议。



# COT in Multi-modal LLMs

GPT-4



Please describe the image as detail as possible and then answer the following questions:  
what is the logo in the image?  
The answer should be as detailed as possible and provide some relevant information

The description and the detailed is separated by one line of "-----"

 The image features a brightly colored juice carton placed on a table with a blurred background that suggests a domestic setting. The carton is predominantly orange, with blue and green accents and features graphic elements and text.

The front of the carton has a large central graphic of an anthropomorphic lion character dressed in sporty attire, including a white tank top and blue shorts. The lion is playing tennis, as indicated by the tennis racket in one hand and a ball in the other. Surrounding the lion character are images of fruits, which indicate the flavors of the juice inside the carton. There are depictions of an apple, a peach, and oranges. These fruit illustrations are quite vibrant and realistic in appearance, in contrast to the cartoonish lion character.

Above the lion, the product name "LEON" is written in large, bold, blue letters with a white outline, which makes it stand out against the orange background. Below the name, there are smaller white letters that read "jabłko, brzoskwinie, pomarańcze," which are the Polish words for "apples, peaches, oranges," describing the juice flavors.

At the bottom of the carton, there is a circular blue logo with a checkmark symbol and the text "ZRÓDŁO WITAMIN" which translates to "SOURCE OF VITAMINS" in Polish. This suggests that the juice is being marketed as a healthy option, rich in vitamins.

The background contains other items that are out of focus, including a wicker basket with a pink ribbon, green grapes, and what appears to be a decorative white miniature bicycle. To the right, there's a glimpse of a red rose and a bowl that contains some sort of food, possibly pastries or bread.

The logo on the carton is the "Hortex" logo, which is placed at the top of the carton. The Hortex logo is relatively small compared to the rest of the graphics and is characterized by bold, white capital letters spelling "Hortex" on a red rectangular background. The text has a shadow effect that gives a sense of depth, and there appears to be a green leaf graphic element to the left of the text, which is likely a part of the logo's design, emphasizing the company's association with natural and plant-based products.

Hortex is a well-known brand in Poland, recognized for its wide range of food products, including frozen fruits and vegetables, as well as various types of juices and beverages. The company has a history dating back several decades and is reputed for providing products made from natural ingredients without the addition of artificial preservatives and colorants.

 Send a message 

# Acknowledgement

- Princeton COS 597G:  
<https://www.cs.princeton.edu/courses/archive/fall22/cos597G/>
- Scaling, emergence, and reasoning (Jason Wei, NYU):  
[https://docs.google.com/presentation/d/1EUV7W7X\\_w0BDrscDhPg7IMGzJCkeaPkGCJ3bN8dluXc/edit?resourcekey=0-7Nz5A7y8JozyVrnDtcEKJA#slide=id.g16197112905\\_0\\_0](https://docs.google.com/presentation/d/1EUV7W7X_w0BDrscDhPg7IMGzJCkeaPkGCJ3bN8dluXc/edit?resourcekey=0-7Nz5A7y8JozyVrnDtcEKJA#slide=id.g16197112905_0_0)
- Prompting engineering lectures(DAIR-AI): <https://github.com/dair-ai/Prompt-Engineering-Guide/blob/main/lecture/Prompt-Engineering-Lecture-Elvis.pdf>
- Prompt engineering guide: <https://www.promptingguide.ai/>

# Optional reading material

In-context learning:

- An Explanation of In-context Learning as Implicit Bayesian Inference(<https://arxiv.org/abs/2111.02080>)
- Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?(<https://arxiv.org/abs/2202.12837>)

Knowledge probing:

- How Much Knowledge Can You Pack Into the Parameters of a Language Model?(<https://arxiv.org/abs/2002.08910>)

Knowledge editing

- Fast model editing at scale(<https://arxiv.org/abs/2110.11309>)