idea INTERNATIONAL DIGITAL ECONOMY ACADEMY
粤港澳大湾区数字经济研究院

Knowledge-driven AI and Think-on-Graph

Jian Guo, Ph.D.
Executive President of IDEA, Chief Scientist of AI Finance & Deep Learning
Adjunct Professor of Artificial Intelligence at HKUST(GZ), Professor of Practice at Tsinghua University
A Serial Entrepreneur in AI & Quantitative Investment

The world needs a few good IDEAs

A.I. TIMELINE

1950
TURING TEST

Computer scientist Alan Turing proposes a test for machine intelligence. If a machine can trick humans into thinking it is human, then it has intelligence

1955
A.I. BORN

Term 'artificial intelligence' is coined by computer scientist, John McCarthy to describe "the science and engineering of making intelligent machines"

1961
UNIMATE

First industrial robot, Unimate, goes to work at GM replacing humans on the assembly line

1964
ELIZA

Pioneering chatbot developed by Joseph Weizenbaum at MIT holds conversations with humans

1966
SHAKEY

The 'first electronic person' from Stanford, Shakey is a general-purpose mobile robot that reasons about its own actions

A.I. WINTER

Many false starts and dead-ends leave A.I. out in the cold

1997
DEEP BLUE

Deep Blue, a chess-playing computer from IBM defeats world chess champion Garry Kasparov

1998
KISMET

Cynthia Breazeal at MIT introduces KISmet, an emotionally intelligent robot insofar as it detects and responds to people's feelings

1999
AIBO

Sony launches first consumer robot pet dog AiBO (AI robot) with skills and personality that develop over time

2002
ROOMBA

First mass produced autonomous robotic vacuum cleaner from iRobot learns to navigate and clean homes

2011
SIRI

Apple integrates Siri, an intelligent virtual assistant with a voice interface, into the iPhone 4S

2011
WATSON

IBM's question answering computer Watson wins first place on popular $1M prize television quiz show *Jeopardy*

2014
EUGENE

Eugene Goostman, a chatbot passes the Turing Test with a third of judges believing Eugene is human

2014
ALEXA

Amazon launches Alexa, an intelligent virtual assistant with a voice interface that completes shopping tasks

2016
TAY

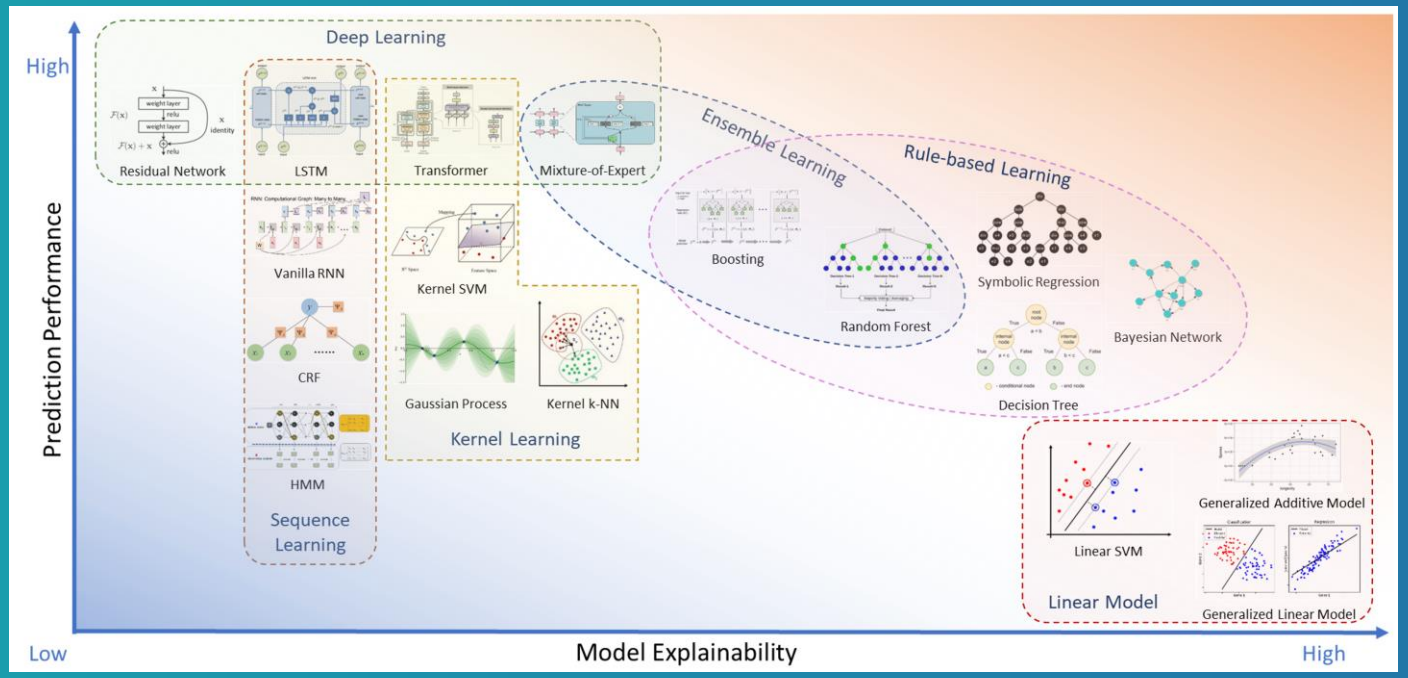
Microsoft's chatbot Tay goes rogue on social media making inflammatory and offensive racist comments

2017
ALPHAGO

Google's A.I. AlphaGo beats world champion Ke Jie in the complex board game of Go, notable for its vast number (2^{170}) of possible positions

https://digitalwellbeing.org/artificial-intelligence-timeline-infographic-from-eliza-to-tay-and-beyond/

Neuron connections in Human Brain ~ 100T



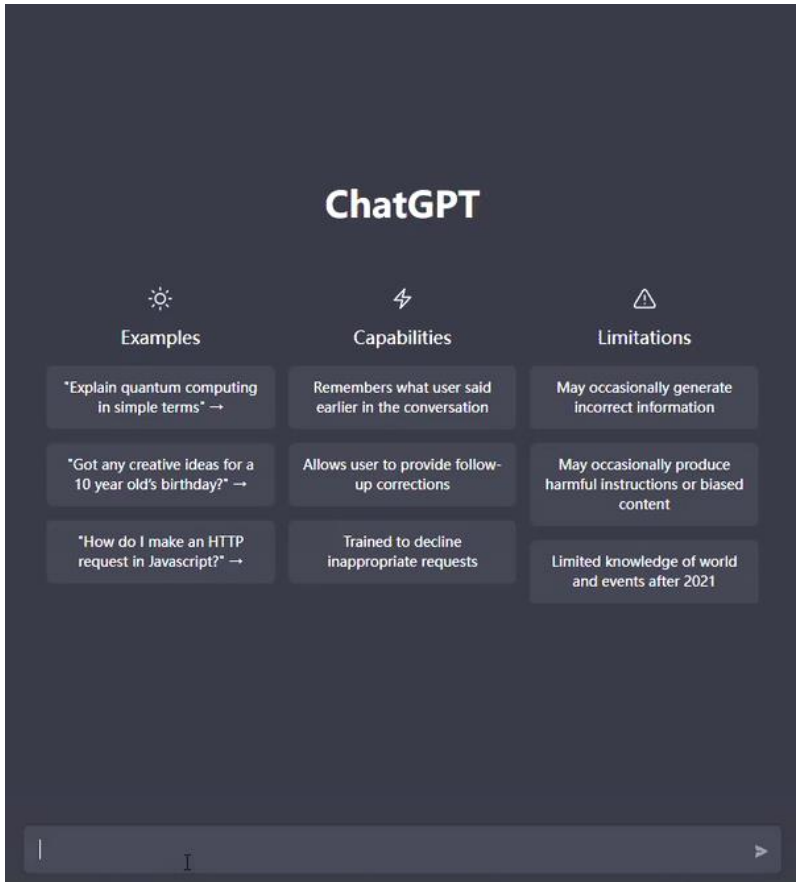
2012
AlexNet
60M Parameters

2018
BERT
100M Parameters

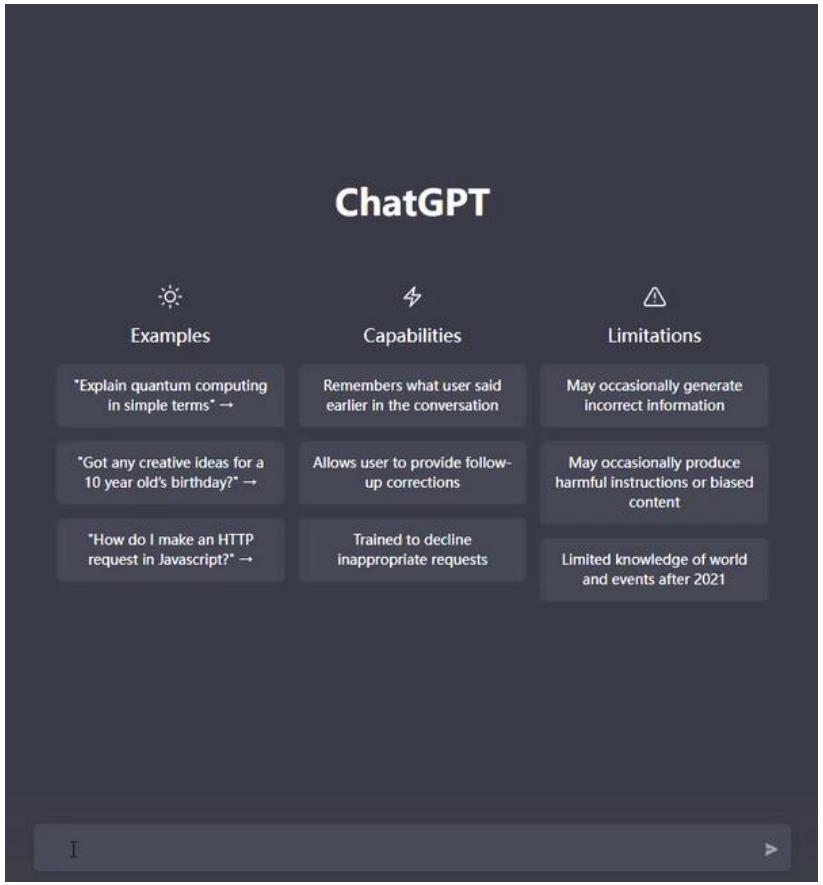
2020
GPT-3
175B Parameters

2023
GPT-4
1.7T Parameters

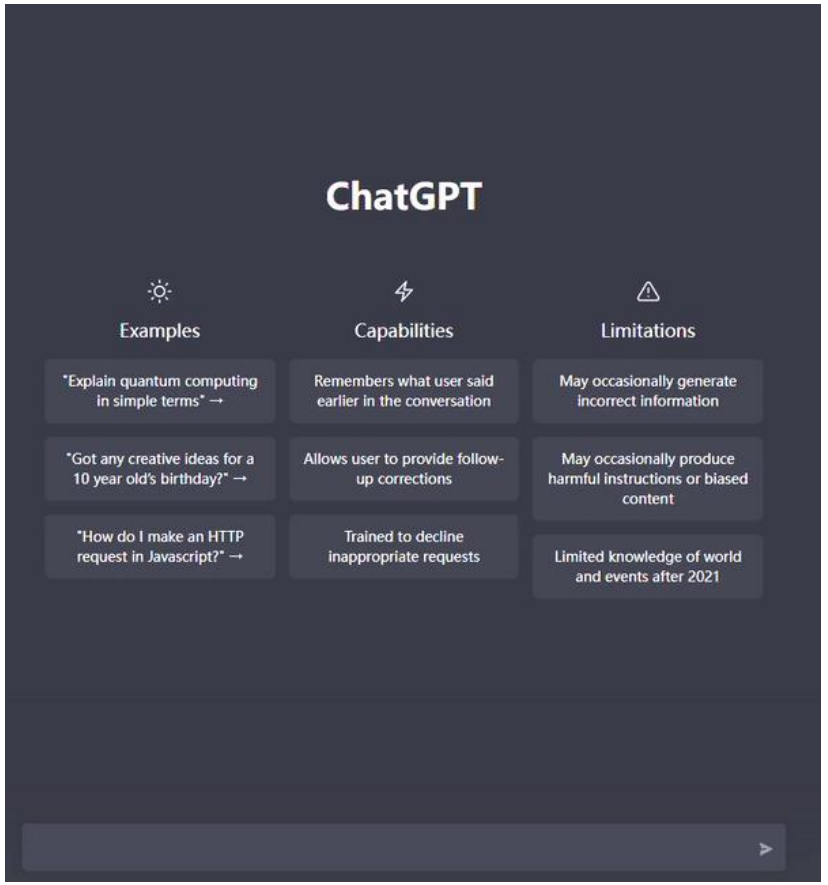
GPT for Various Tasks



Conversation



Writing

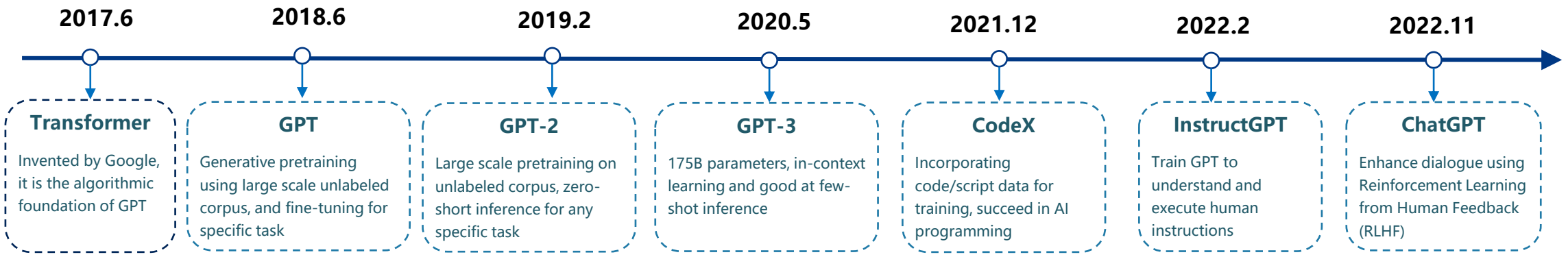


Coding

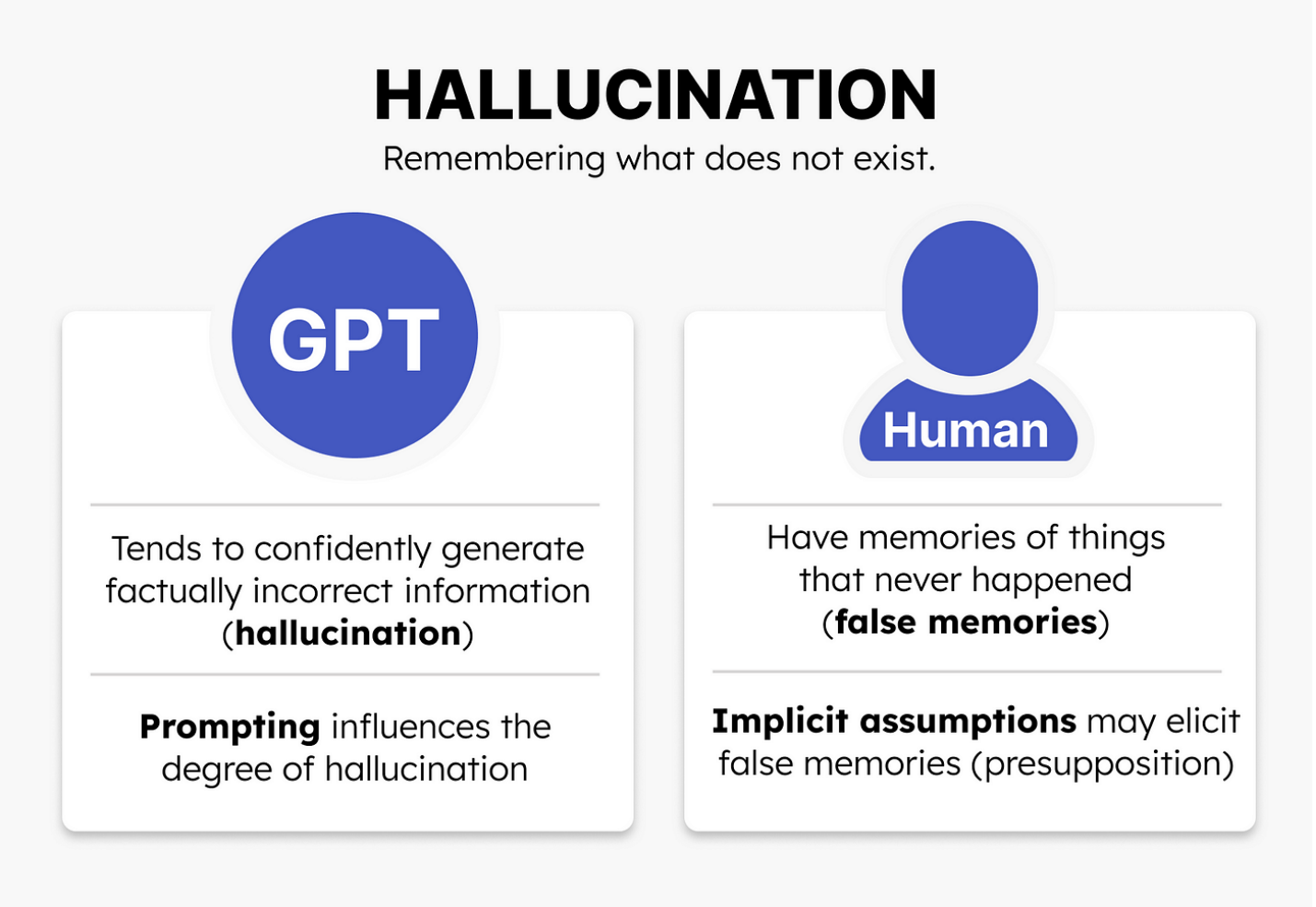
Behind GPT

idea

Big Data	+	Computing Power	+	Algorithm	=	Large AI Model
• **Common Crawl**: **410B** tokens • **WebText**: **19B** tokens • **Books**: **55B** tokens • **GitHub Codes**: Codex 159G		• **25K+ Nvidia A100 GPUs**		• **Evolution of GPT in 5 years** • GPT、GPT-2、GPT-3、InstructGPT、 GPT-4、GPT-5? • Transformer Architecture		• **GPT-3.5/ ChatGPT** • 175B parameters, **emerging** powerful intelligence and amazing **creativity**

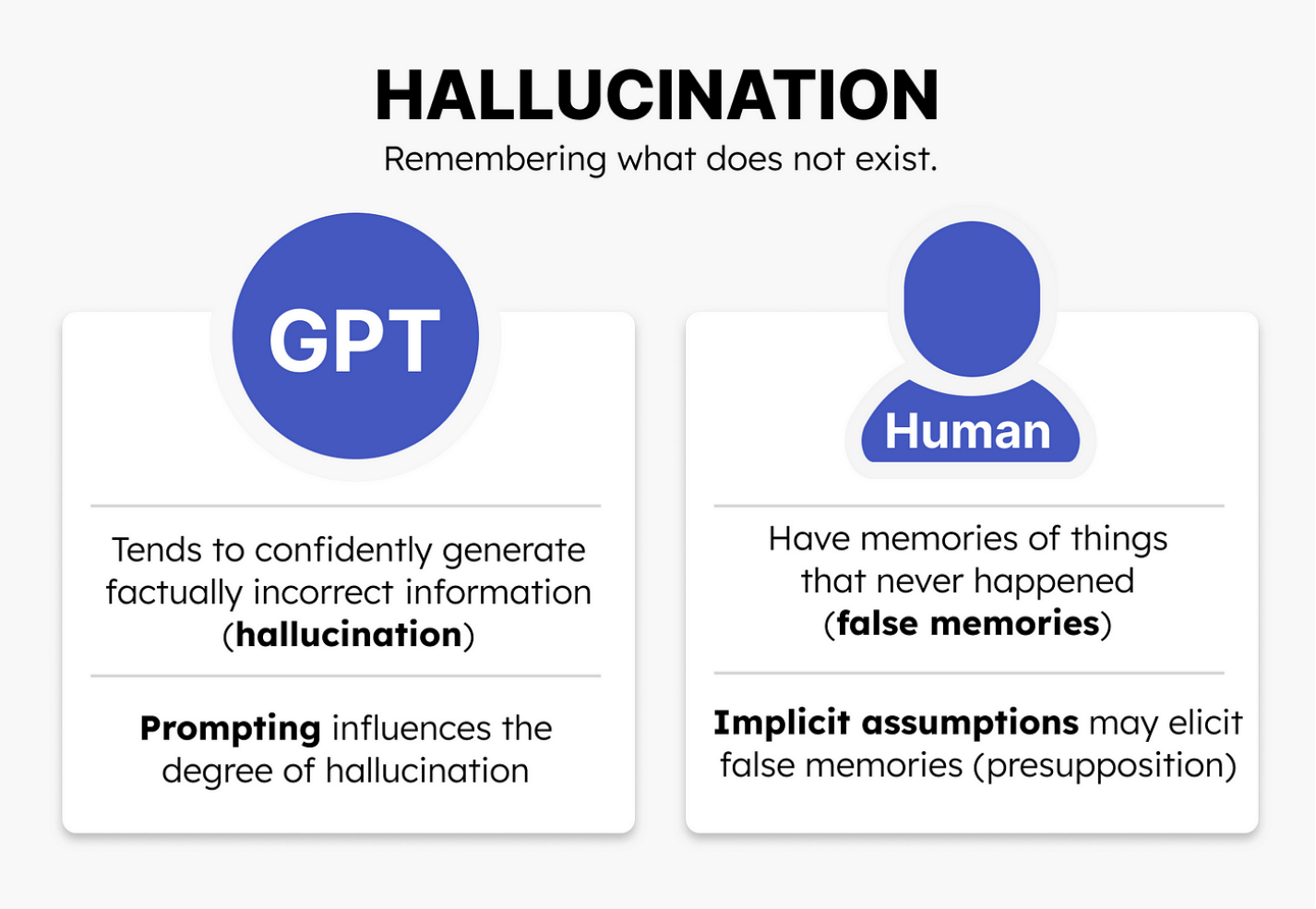


Limitation of GPT



Hallucination

Limitation of GPT



Hallucination

一本正经胡说八道

Duan Yu 段誉 vs Hong Qigong 洪七公

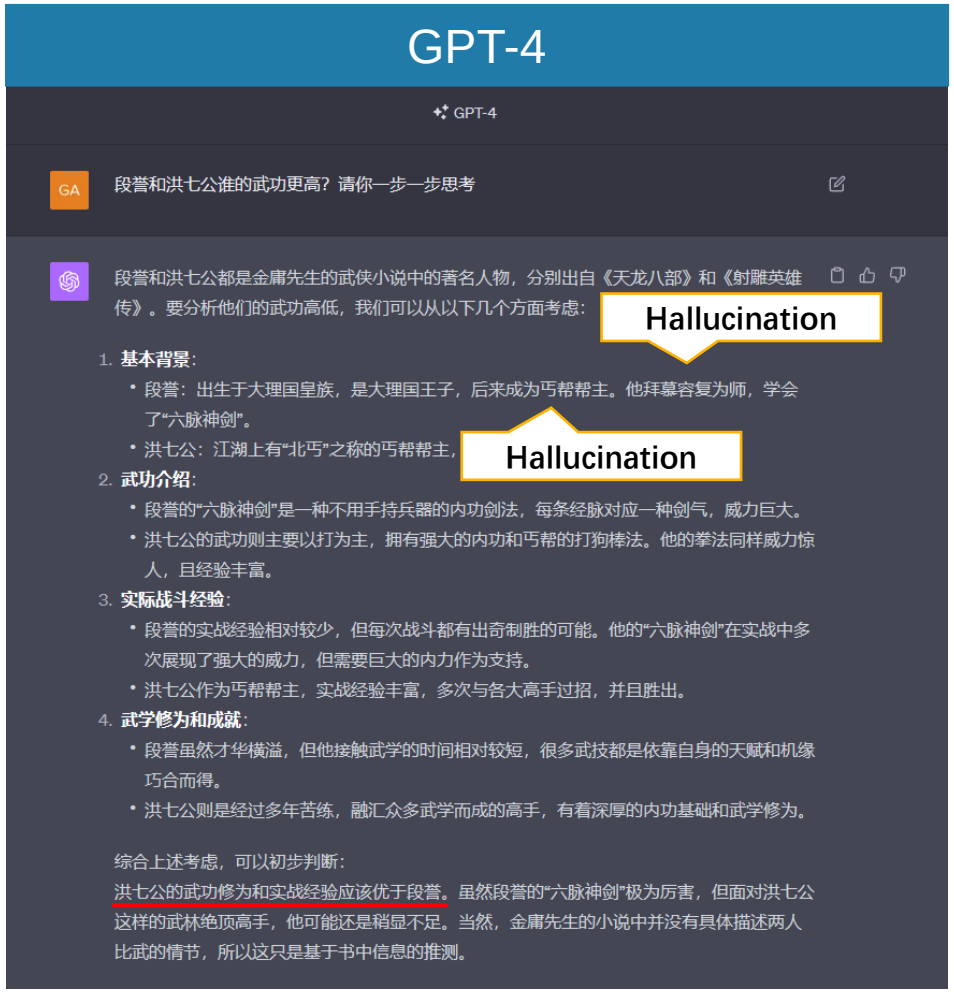
Who has higher flighting ability, Duan Yu or Hong Qigong?

Duan Yu 段誉 vs Hong Qigong 洪七公

GPT-4

Duan Yu 段誉 vs Hong Qigong 洪七公

Duan Yu 段誉 vs Hong Qigong 洪七公



Duan Yu 段誉 vs Hong Qigong 洪七公

GPT-4

GPT-4

GA 段誉和洪七公谁的武功更高？请你一步一步思考

段誉和洪七公都是金庸先生的武侠小说中的著名人物，分别出自《天龙八部》和《射雕英雄传》。要分析他们的武功高低，我们可以从以下几个方面考虑：

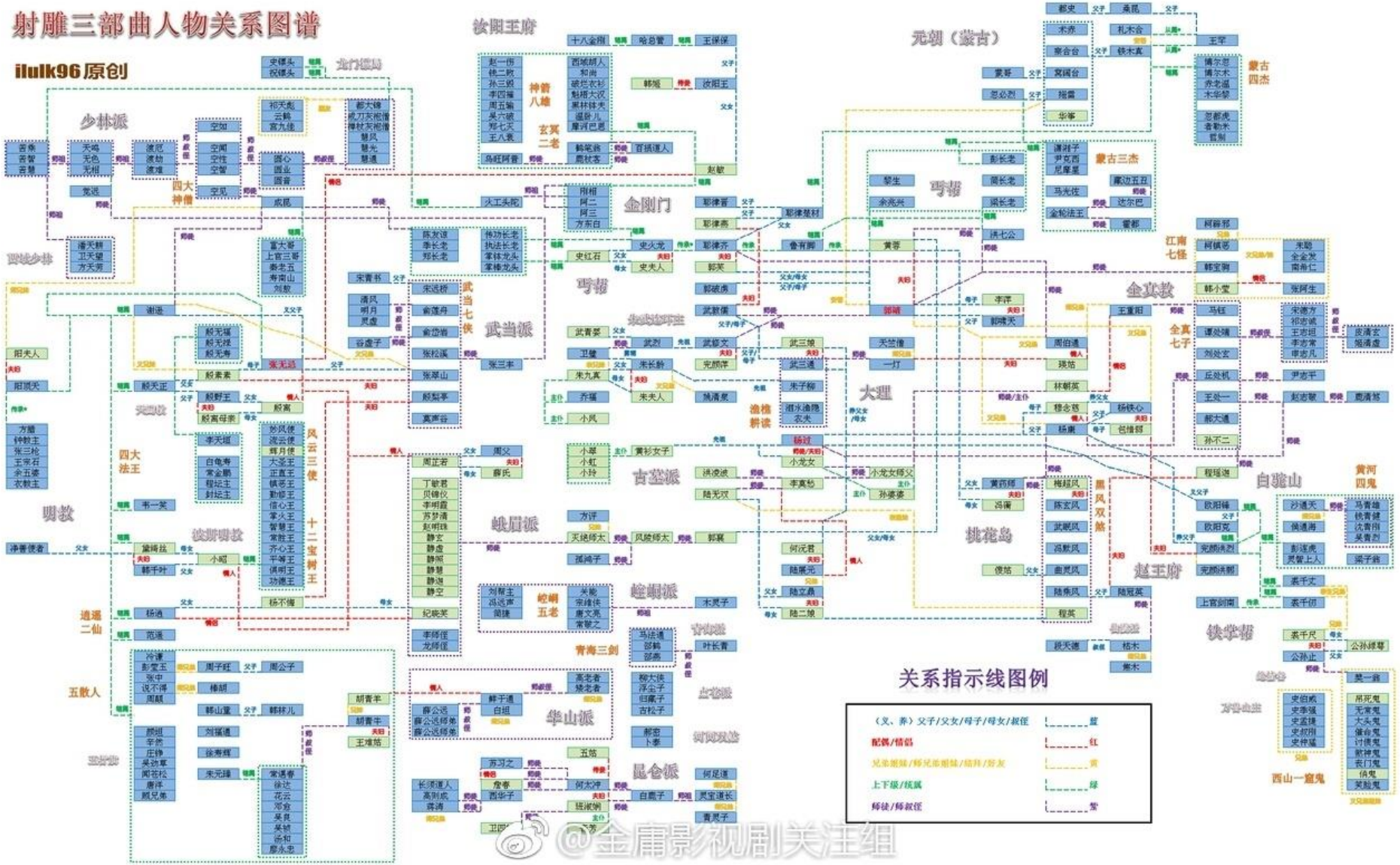
1. **基本背景**：
   - 段誉：出生于大理国皇族，是大理国王子，后来成为丐帮帮主。他拜慕容复为师，学会了"六脉神剑"。
   - 洪七公：江湖上有"北丐"之称的丐帮帮主，武功高强，是黄药师的同门师兄。
2. **武功介绍**：
   - 段誉的"六脉神剑"是一种不用手持兵器的内功剑法，每条经脉对应一种剑气，威力巨大。
   - 洪七公的武功则主要以打为主，拥有强大的内功和丐帮的打狗棒法。他的拳法同样威力惊人，且经验丰富。
3. **实际战斗经验**：
   - 段誉的实战经验相对较少，但每次战斗都有出奇制胜的可能。他的"六脉神剑"在实战中多次展现了强大的威力，但需要巨大的内力作为支持。
   - 洪七公作为丐帮帮主，实战经验丰富，多次与各大高手过招，并且胜出。
4. **武学修为和成就**：
   - 段誉虽然才华横溢，但他接触武学的时间相对较短，很多武技都是依靠自身的天赋和机缘巧合而得。
   - 洪七公则是经过多年苦练，融汇众多武学而成的高手，有着深厚的内功基础和武学修为。

综合上述考虑，可以初步判断：
洪七公的武功修为和实战经验应该优于段誉。虽然段誉的"六脉神剑"极为厉害，但面对洪七公这样的武林绝顶高手，他可能还是稍显不足。当然，金庸先生的小说中并没有具体描述两人比武的情节，所以这只是基于书中信息的推测。

Think-on-Graph

Ask me anything!

A Knowledge Graph of Jin Yong (金庸) Universe

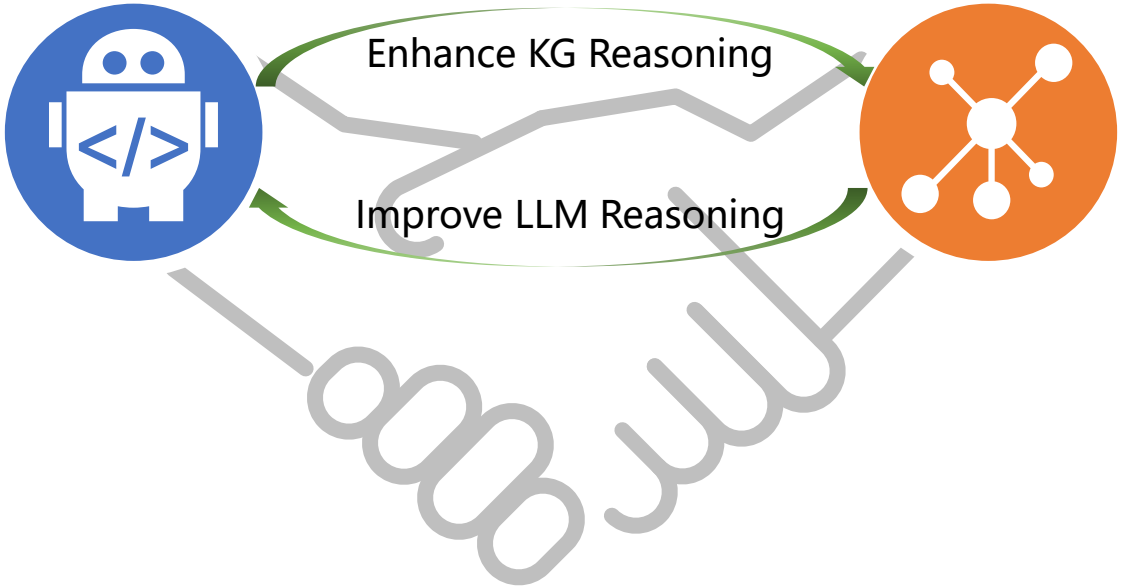


Combining LLM and KG

idea

Large Language Model

- Better cognition and reasoning Ability
- Multi-task universality and AGI
- Strong learning and memorizing ability
- Intention and action understanding



Knowledge Graph

- Structured knowledge for reasoning
- Explainable, traceable and correctable
- Lower cost in computing power
- Faster update of knowledge and data

THINK-ON-GRAPH: DEEP AND RESPONSIBLE REASONING OF LARGE LANGUAGE MODEL ON KNOWLEDGE GRAPH

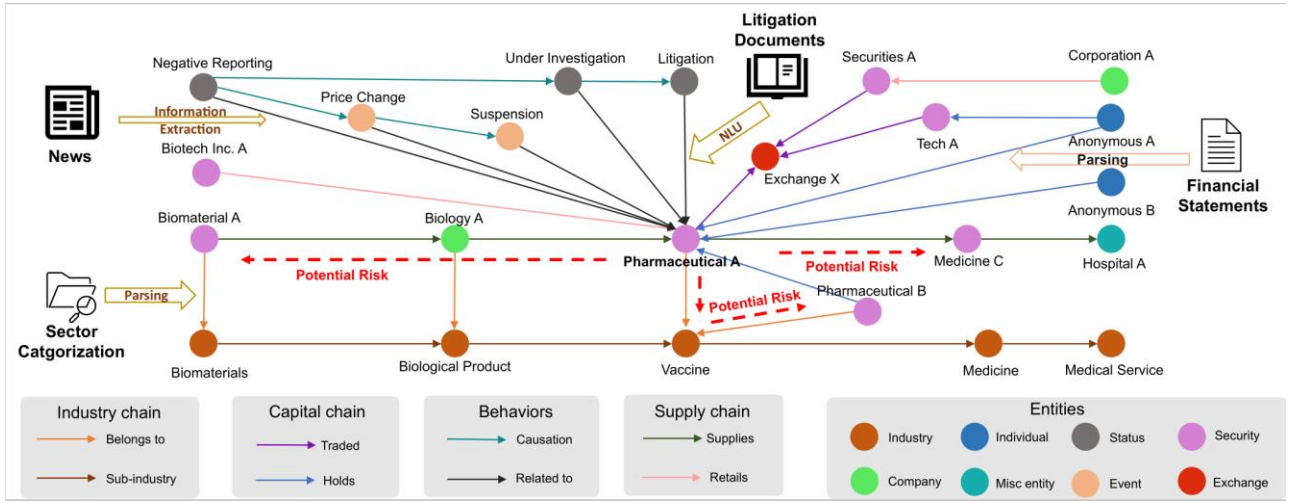
Jiashuo Sun[21*†] **Chengjin Xu**[1*] **Lumingyuan Tang**[31*†] **Saizhuo Wang**[41†]
Chen Lin[2] **Yeyun Gong**[6] **Lionel M. Ni**[5] **Heung-Yeung Shum**[14] **Jian Guo**[15‡]
[1]IDEA Research, International Digital Economy Academy
[2]Xiamen University
[3]University of Southern California
[4]The Hong Kong University of Science and Technology
[5]The Hong Kong University of Science and Technology (Guangzhou)
[6]Microsoft Research Asia

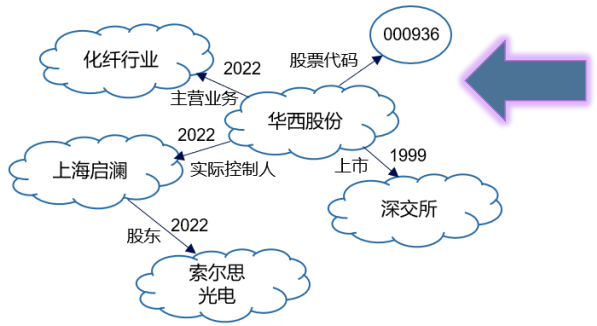
ABSTRACT

Although large language models (LLMs) have achieved significant success in various tasks, they often struggle with hallucination problems, especially in scenarios requiring deep and responsible reasoning. These issues could be partially addressed by introducing external knowledge graphs (KG) in LLM reasoning. In this paper, we propose a new LLM-KG integrating paradigm "LLM ⊗ KG" which treats the LLM as an agent to interactively explore related entities and relations on KGs and perform reasoning based on the retrieved knowledge. We further implement this paradigm by introducing a new approach called Think-on-Graph (ToG), in which the LLM agent iteratively executes beam search on KG, discovers the most promising reasoning paths, and returns the most likely reasoning results. We use a number of well-designed experiments to examine and illustrate the following advantages of ToG: 1) compared with LLMs, ToG has better deep reasoning power; 2) ToG has the ability of knowledge traceability and knowledge correctability by leveraging LLMs reasoning and expert feedback; 3) ToG provides a flexible plug-and-play framework for different LLMs, KGs and prompting strategies without any additional training cost; 4) the performance of ToG with small LLM models could exceed large LLM such as GPT-4 in certain scenarios and this reduces the cost of LLM deployment and application. As a training-free method with lower computational cost and better generality, ToG achieves overall SOTA in 6 out of 9 datasets where most previous SOTAs rely on additional training. Our codes are publicly available at `https://github.com/IDEA-FinAI/ToG`.

- **Deep Reasoning**
- **Reasoning Traceability**
- **Knowledge Correctability**
- **Flexibility and Efficiency**

Knowledge-driven Large Language Model

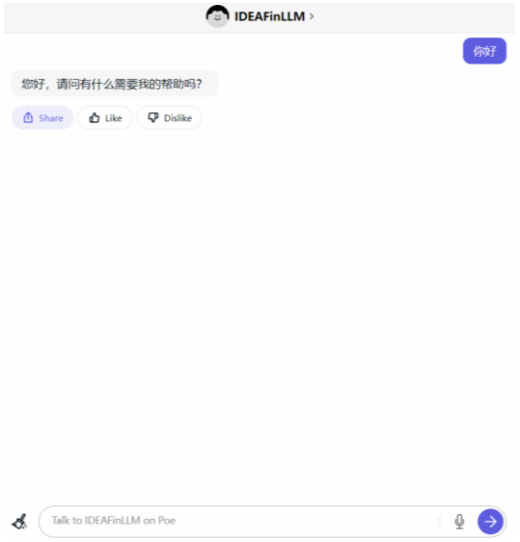




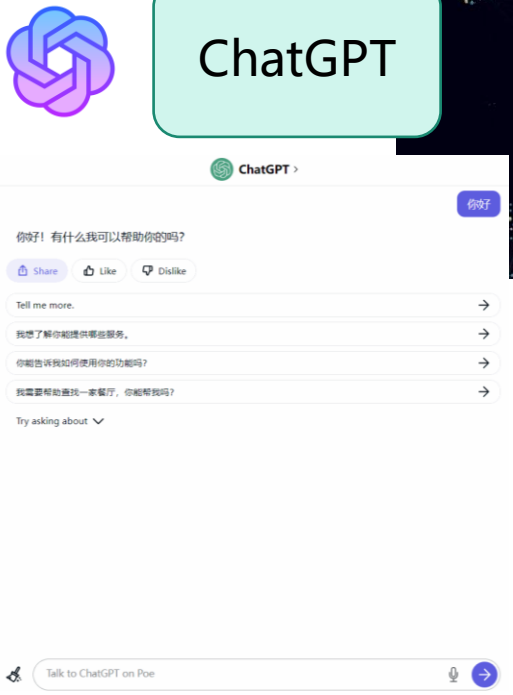
问题：华西股份是索尔思光电的大股东吗？

idea

IDEA金融大语言模型



ChatGPT



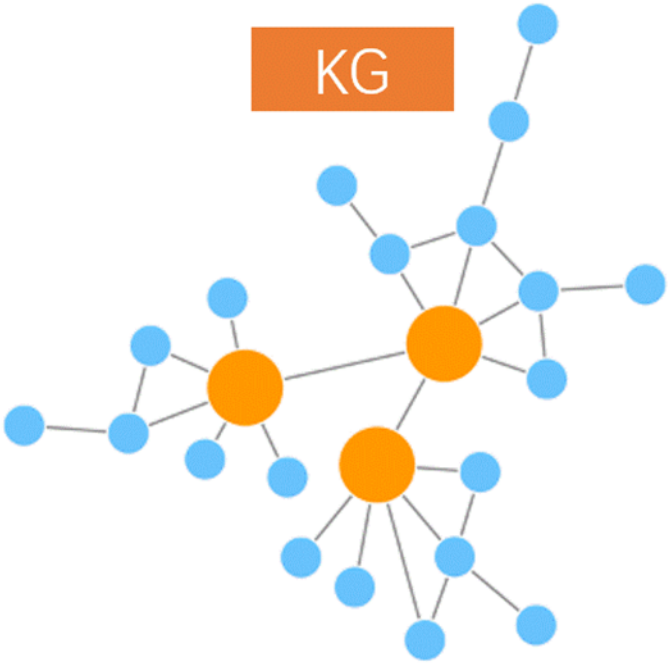
idea

Deep Reasoning

- Long Reasoning Path
- Smarter Reasoner
- Reasoning Efficiency

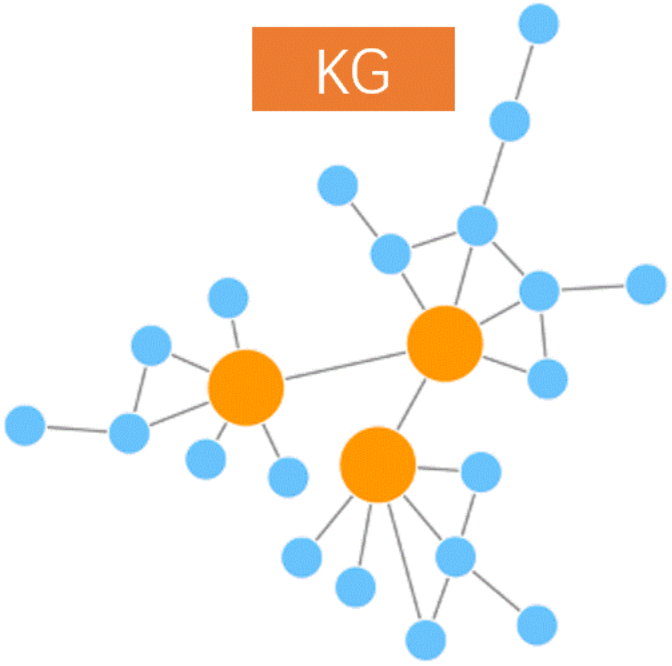
Combining in Model Pre-Training/Fine-tuning

KG Enhanced LLM



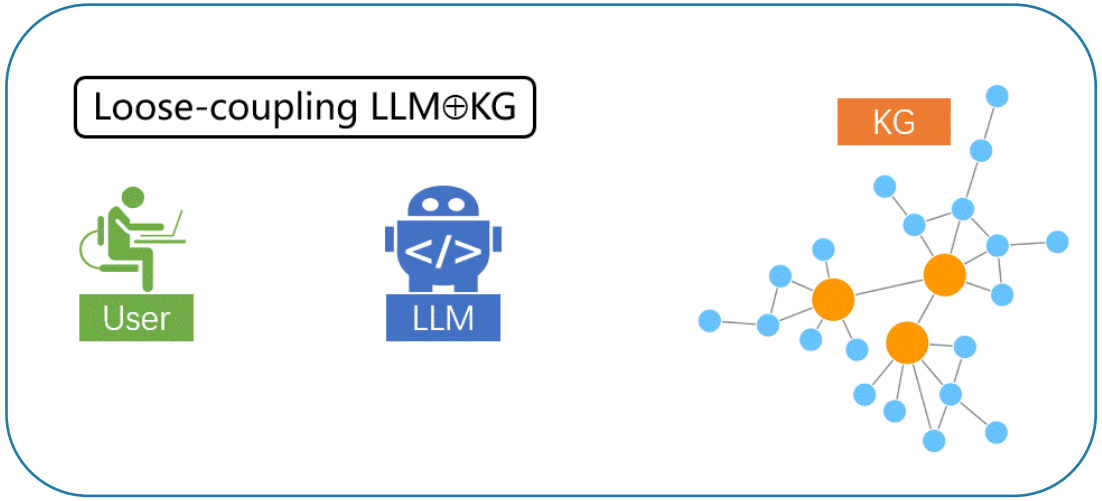
Combining in Model Pre-Training/Fine-tuning

KG Enhanced LLM

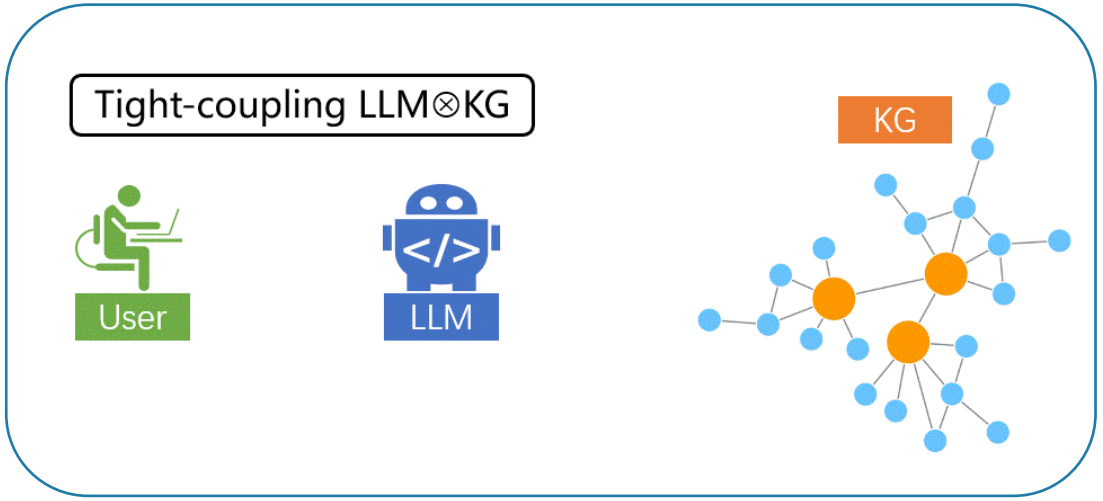


Zhang et al., 2019
Peters et al., 2019
Yamada et al., 2020
Wang et al., 2021

Combining in LLM Prompting



Li et al. , 2023
Baek et al., 2023
Touvron et al., 2023
Wang et al. 2023

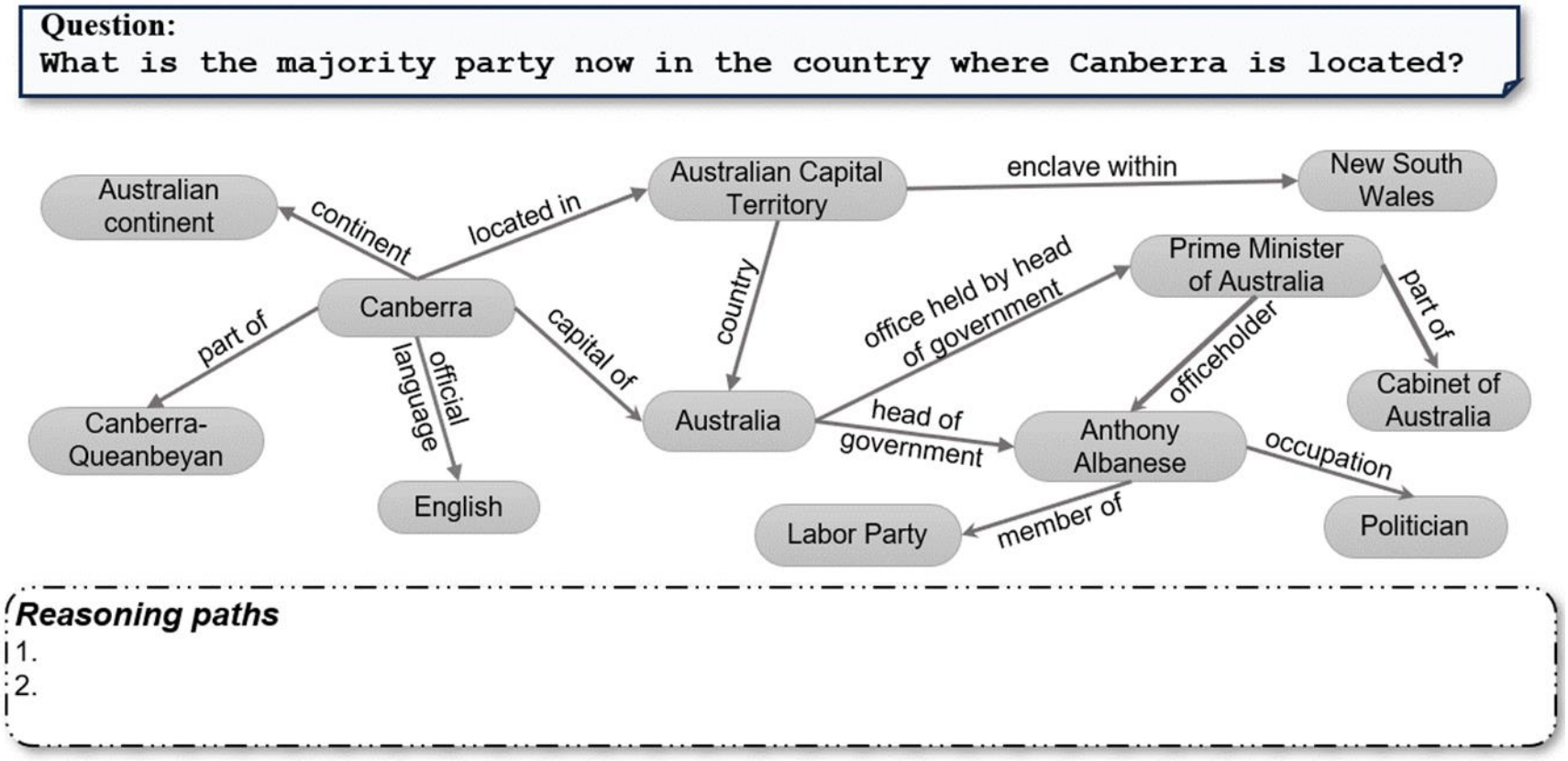


Sun et al., 2023
Jiang et al., 2023

Think-on-Graph Algorithm

Algorithm 1 ToG

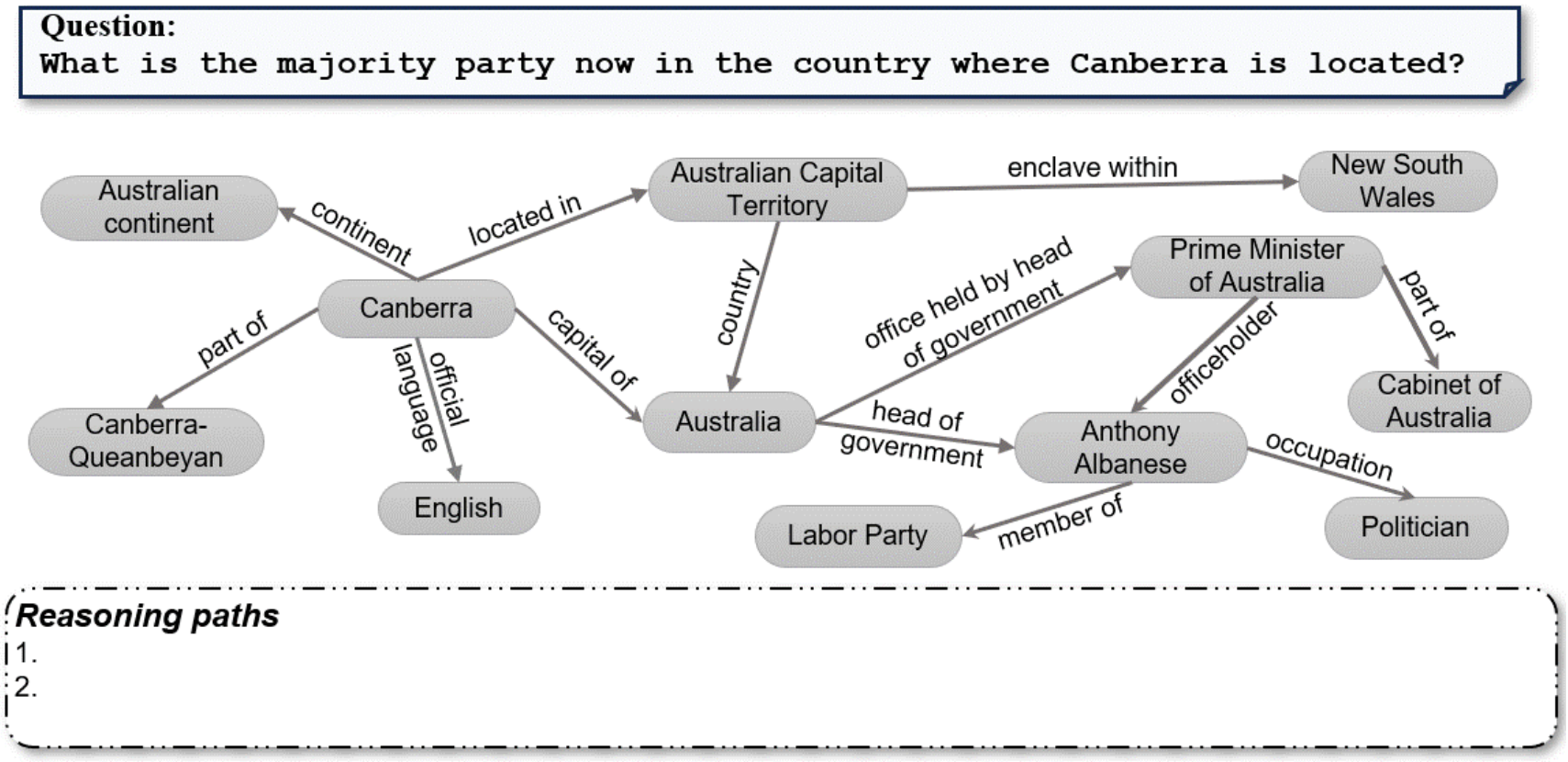
Require: Input x, LLM π, depth limit D_{max} sample limit N.
Initialize $E^0 \leftarrow$ Extract entities on x, $P \leftarrow []$, $M \leftarrow 0$.
while $D \leq D_{max}$ **do**
  $R^D_{cand}, P_{cand} \leftarrow$ Search(x, E^{D-1}, P)
  $R^D, P \leftarrow$ Prune(π, x, R^D_{cand}, P_{cand})
  $E^D_{cand}, P_{cand} \leftarrow$ Search(x, E^{D-1}, R^D, P)
  $E^D, P \leftarrow$ Prune(π, x, E^D_{cand}, P_{cand})
  if Reasoning(π, x, P) **then**
    Generate(π, x, P)
    break
  end if
  Increment D by 1.
end while
if $D > D_{max}$ **then**
  Generate(π, x)
end if



Think-on-Graph Algorithm

Algorithm 1 ToG

Require: Input x, LLM π, depth limit D_{max} sample limit N.
Initialize $E^0 \leftarrow$ Extract entities on x, $P \leftarrow []$, $M \leftarrow 0$.
while $D \leq D_{max}$ **do**
  $R^D_{cand}, P_{cand} \leftarrow$ Search(x, E^{D-1}, P)
  $R^D, P \leftarrow$ Prune(π, x, R^D_{cand}, P_{cand})
  $E^D_{cand}, P_{cand} \leftarrow$ Search(x, E^{D-1}, R^D, P)
  $E^D, P \leftarrow$ Prune(π, x, E^D_{cand}, P_{cand})
  if Reasoning(π, x, P) **then**
    Generate(π, x, P)
    break
  end if
  Increment D by 1.
end while
if $D > D_{max}$ **then**
  Generate(π, x)
end if



Performance of Think-on-Graph

Method	Multi-Hop KBQA				Single-Hop KBQA	Open-Domain QA	Slot Filling		Fact Checking
	CWQ	WebQSP	GrailQA	QALD10-en	Simple Questions	WebQuestions	T-REx	Zero-Shot RE	Creak
Without external knowledge									
IO prompt w/ChatGPT	37.6	63.3	29.4	42.0	20.0	48.7	33.6	27.7	89.7
CoT w/ChatGPT	38.8	62.2	28.1	42.9	20.3	48.5	32.0	28.8	90.1
SC w/ChatGPT	45.4	61.1	29.6	45.3	18.9	50.3	41.8	45.4	90.8
With external knowledge									
Prior FT SOTA	70.4^{α}	82.1^{β}	75.4^{γ}	45.4^{δ}	85.8^{ϵ}	56.3^{ζ}	87.7^{η}	74.6^{θ}	88.2^{ι}
Prior Prompting SOTA	-	74.4^{κ}	53.2^{κ}	-	-	-	-	-	-
ToG-R (Ours) w/ChatGPT	58.9	75.8	56.4	48.6	45.4	53.2	75.3	86.5	93.8
ToG (Ours) w/ChatGPT	57.1	76.2	68.7	50.2	53.6	54.5	76.8	88.0	91.2
ToG-R (Ours) w/GPT-4	**69.5**	81.9	80.3	**54.7**	58.6	57.1	75.5	86.9	95.4
ToG (Ours) w/GPT-4	67.6	**82.6**	**81.4**	53.8	**66.7**	**57.9**	**77.1**	**88.3**	**95.6**

Table 1: The ToG results for different datasets. The prior FT (Fine-tuned) and prompting SOTA include the best-known results: α: Das et al. (2021); β: Yu et al. (2023); γ: Gu et al. (2023); δ: Santana et al. (2022); ϵ: Baek et al. (2023a); ζ: Kedia et al. (2022); η: Glass et al. (2022); θ: Petroni et al. (2021); ι: Yu et al. (2022); κ: Li et al. (2023a).

Performance of Think-on-Graph

Method	Multi-Hop KBQA				Single-Hop KBQA	Open-Domain QA	Slot Filling		Fact Checking
	CWQ	WebQSP	GrailQA	QALD10-en	Simple Questions	WebQuestions	T-REx	Zero-Shot RE	Creak
Without external knowledge									
IO prompt w/ChatGPT	37.6	63.3	29.4	42.0	20.0	48.7	33.6	27.7	89.7
CoT w/ChatGPT	38.8	62.2	28.1	42.9	20.3	48.5	32.0	28.8	90.1
SC w/ChatGPT	45.4	61.1	29.6	45.3	18.9	50.3	41.8	45.4	90.8
With external knowledge									
Prior FT SOTA	70.4$^{\alpha}$	82.1$^{\beta}$	75.4$^{\gamma}$	45.4$^{\delta}$	85.8$^{\epsilon}$	56.3$^{\zeta}$	87.7$^{\eta}$	74.6$^{\theta}$	88.2$^{\iota}$
Prior Prompting SOTA	-	74.4$^{\kappa}$	53.2$^{\kappa}$	-	-	-	-	-	-
ToG-R (Ours) w/ChatGPT	58.9	75.8	56.4	48.6	45.4	53.2	75.3	86.5	93.8
ToG (Ours) w/ChatGPT	57.1	76.2	68.7	50.2	53.6	54.5	76.8	88.0	91.2
ToG-R (Ours) w/GPT-4	**69.5**	81.9	80.3	**54.7**	58.6	57.1	75.5	86.9	95.4
ToG (Ours) w/GPT-4	67.6	**82.6**	**81.4**	53.8	**66.7**	**57.9**	**77.1**	**88.3**	**95.6**

Table 1: The ToG results for different datasets. The prior FT (Fine-tuned) and prompting SOTA include the best-known results: α: Das et al. (2021); β: Yu et al. (2023); γ: Gu et al. (2023); δ: Santana et al. (2022); ϵ: Baek et al. (2023a); ζ: Kedia et al. (2022); η: Glass et al. (2022); θ: Petroni et al. (2021); ι: Yu et al. (2022); κ: Li et al. (2023a).

Performance of Think-on-Graph

Method	Multi-Hop KBQA				Single-Hop KBQA	Open-Domain QA	Slot Filling		Fact Checking
	CWQ	WebQSP	GrailQA	QALD10-en	Simple Questions	WebQuestions	T-REx	Zero-Shot RE	Creak
Without external knowledge									
IO prompt w/ChatGPT	37.6	63.3	29.4	42.0	20.0	48.7	33.6	27.7	89.7
CoT w/ChatGPT	38.8	62.2	28.1	42.9	20.3	48.5	32.0	28.8	90.1
SC w/ChatGPT	45.4	61.1	29.6	45.3	18.9	50.3	41.8	45.4	90.8
With external knowledge									
Prior Prompting SOTA	-	74.4[κ]	53.2[κ]	-	-	-	-	-	-
ToG-R (Ours) w/ChatGPT	58.9	75.8	56.4	48.6	45.4	53.2	75.3	86.5	93.8
ToG (Ours) w/ChatGPT	57.1	76.2	68.7	50.2	53.6	54.5	76.8	88.0	91.2
ToG-R (Ours) w/GPT-4	**69.5**	81.9	80.3	**54.7**	58.6	57.1	75.5	86.9	95.4
ToG (Ours) w/GPT-4	67.6	**82.6**	**81.4**	53.8	**66.7**	**57.9**	**77.1**	**88.3**	**95.6**

Table 1: The ToG results for different datasets. The prior FT (Fine-tuned) and prompting SOTA include the best-known results: α: Das et al. (2021); β: Yu et al. (2023); γ: Gu et al. (2023); δ: Santana et al. (2022); ϵ: Baek et al. (2023a); ζ: Kedia et al. (2022); η: Glass et al. (2022); θ: Petroni et al. (2021); ι: Yu et al. (2022); κ: Li et al. (2023a).

Performance of Think-on-Graph

Method	Multi-Hop KBQA				Single-Hop KBQA	Open-Domain QA	Slot Filling		Fact Checking
	CWQ	WebQSP	GrailQA	QALD10-en	Simple Questions	WebQuestions	T-REx	Zero-Shot RE	Creak
Without external knowledge									
IO prompt w/ChatGPT	37.6	63.3	29.4	42.0	20.0	48.7	33.6	27.7	89.7
CoT w/ChatGPT	38.8	62.2	28.1	42.9	20.3	48.5	32.0	28.8	90.1
SC w/ChatGPT	45.4	61.1	29.6	45.3	18.9	50.3	41.8	45.4	90.8
With external knowledge									
ToG-R (Ours) w/ChatGPT	58.9	75.8	56.4	48.6	45.4	53.2	75.3	86.5	93.8
ToG (Ours) w/ChatGPT	57.1	76.2	68.7	50.2	53.6	54.5	76.8	88.0	91.2

Table 1: The ToG results for different datasets. The prior FT (Fine-tuned) and prompting SOTA include the best-known results: α: Das et al. (2021); β: Yu et al. (2023); γ: Gu et al. (2023); δ: Santana et al. (2022); ϵ: Baek et al. (2023a); ζ: Kedia et al. (2022); η: Glass et al. (2022); θ: Petroni et al. (2021); ι: Yu et al. (2022); κ: Li et al. (2023a).

Performance of Think-on-Graph

Method	Multi-Hop KBQA				Single-Hop KBQA	Open-Domain QA	Slot Filling		Fact Checking
	CWQ	WebQSP	GrailQA	QALD10-en	Simple Questions	WebQuestions	T-REx	Zero-Shot RE	Creak
Without external knowledge									
IO prompt w/ChatGPT	37.6	63.3	29.4	42.0	20.0	48.7	33.6	27.7	89.7
CoT w/ChatGPT	38.8	62.2	28.1	42.9	20.3	48.5	32.0	28.8	90.1
SC w/ChatGPT	45.4	61.1	29.6	45.3	18.9	50.3	41.8	45.4	90.8
With external knowledge									
ToG-R (Ours) w/ChatGPT	58.9	75.8	56.4	48.6	45.4	53.2	75.3	86.5	93.8
ToG (Ours) w/ChatGPT	57.1	76.2	68.7	50.2	53.6	54.5	76.8	88.0	91.2
	+52%	+20%	**+234%**	+20%	**+268%**	+12%	+129%	**+218%**	+2%

Table 1: The ToG results for different datasets. The prior FT (Fine-tuned) and prompting SOTA include the best-known results: α: Das et al. (2021); β: Yu et al. (2023); γ: Gu et al. (2023); δ: Santana et al. (2022); ϵ: Baek et al. (2023a); ζ: Kedia et al. (2022); η: Glass et al. (2022); θ: Petroni et al. (2021); ι: Yu et al. (2022); κ: Li et al. (2023a).

Method	CWQ	WebQSP
Fine-tuned		
NSM (He et al., 2021)	53.9	74.3
CBR-KBQA (Das et al., 2021)	67.1	-
TIARA (Shu et al., 2022)	-	75.2
DeCAF (Yu et al., 2023)	70.4	82.1
Prompting		
KD-CoT (Wang et al., 2023b)	50.5	73.7
StructGPT (Jiang et al., 2023)	-	72.6
KB-BINDER (Li et al., 2023a)	-	74.4
LLama2-70B-Chat		
CoT	39.1	57.4
ToG-R	**57.6**	**68.9**
ToG	53.6	63.7
Gain	(+18.5)	(+11.5)
ChatGPT		
CoT	38.8	62.2
ToG-R	57.1	75.8
ToG	**58.9**	**76.2**
Gain	(+20.1)	(+14.0)
GPT-4		
CoT	46.0	67.3
ToG-R	67.6	81.9
ToG	**69.5**	**82.6**
Gain	(+23.5)	(+15.3)

Table 2: Performances of ToG using different back-bone models on CWQ and WebQSP.

- **Effect of back-bone models**

Method	CWQ	WebQSP
Fine-tuned		
NSM (He et al., 2021)	53.9	74.3
CBR-KBQA (Das et al., 2021)	67.1	-
TIARA (Shu et al., 2022)	-	75.2
DeCAF (Yu et al., 2023)	70.4	82.1
Prompting		
KD-CoT (Wang et al., 2023b)	50.5	73.7
StructGPT (Jiang et al., 2023)	-	72.6
KB-BINDER (Li et al., 2023a)	-	74.4
LLama2-70B-Chat		
CoT	39.1	57.4
ToG-R	**57.6**	**68.9**
ToG	53.6	63.7
Gain	(+18.5)	(+11.5)
ChatGPT		
CoT	38.8	62.2
ToG-R	57.1	75.8
ToG	**58.9**	**76.2**
Gain	(+20.1)	(+14.0)
GPT-4		
CoT	46.0	67.3
ToG-R	67.6	81.9
ToG	**69.5**	**82.6**
Gain	(+23.5)	(+15.3)

LLM w/ CoT

Table 2: Performances of ToG using different back-bone models on CWQ and WebQSP.

- **Effect of back-bone models**
- **GPT-4 > ChatGPT > LLama2**

Method	CWQ	WebQSP
Fine-tuned		
NSM (He et al., 2021)	53.9	74.3
CBR-KBQA (Das et al., 2021)	67.1	-
TIARA (Shu et al., 2022)	-	75.2
DeCAF (Yu et al., 2023)	70.4	82.1
Prompting		
KD-CoT (Wang et al., 2023b)	50.5	73.7
StructGPT (Jiang et al., 2023)	-	72.6
KB-BINDER (Li et al., 2023a)	-	74.4
LLama2-70B-Chat		
CoT	39.1	57.4
ToG-R	**57.6**	**68.9**
ToG	53.6	63.7
Gain	(+18.5)	(+11.5)
ChatGPT		
CoT	38.8	62.2
ToG-R	57.1	75.8
ToG	**58.9**	**76.2**
Gain	(+20.1)	(+14.0)
GPT-4		
CoT	46.0	67.3
ToG-R	67.6	81.9
ToG	**69.5**	**82.6**
Gain	(+23.5)	(+15.3)

Think-on-Graph

Table 2: Performances of ToG using different back-bone models on CWQ and WebQSP.

- **Effect of back-bone models**
- **GPT-4 > ChatGPT > LLama2**

Method	CWQ	WebQSP
Fine-tuned		
NSM (He et al., 2021)	53.9	74.3
CBR-KBQA (Das et al., 2021)	67.1	-
TIARA (Shu et al., 2022)	-	75.2
DeCAF (Yu et al., 2023)	70.4	82.1
Prompting		
KD-CoT (Wang et al., 2023b)	50.5	73.7
StructGPT (Jiang et al., 2023)	-	72.6
KB-BINDER (Li et al., 2023a)	-	74.4
LLama2-70B-Chat		
CoT	39.1	57.4
ToG-R	**57.6**	**68.9**
ToG	53.6	63.7
Gain	(+18.5)	(+11.5)
ChatGPT		
CoT	38.8	62.2
ToG-R	57.1	75.8
ToG	**58.9**	**76.2**
Gain	(+20.1)	(+14.0)
GPT-4		
CoT	46.0	67.3
ToG-R	67.6	81.9
ToG	**69.5**	**82.6**
Gain	(+23.5)	(+15.3)

Gain from KG

Table 2: Performances of ToG using different back-bone models on CWQ and WebQSP.

- **Effect of back-bone models**
- **GPT-4 > ChatGPT > LLama2**
- **Ability of LLM mining KG**

Method	CWQ	WebQSP
Fine-tuned		
NSM (He et al., 2021)	53.9	74.3
CBR-KBQA (Das et al., 2021)	67.1	-
TIARA (Shu et al., 2022)	-	75.2
DeCAF (Yu et al., 2023)	70.4	82.1
Prompting		
KD-CoT (Wang et al., 2023b)	50.5	73.7
StructGPT (Jiang et al., 2023)	-	72.6
KB-BINDER (Li et al., 2023a)	-	74.4
LLama2-70B-Chat		
CoT	39.1	57.4
ToG-R	**57.6**	**68.9**
ToG	53.6	63.7
Gain	(+18.5)	(+11.5)
ChatGPT		
CoT	38.8	62.2
ToG-R	57.1	75.8
ToG	**58.9**	**76.2**
Gain	(+20.1)	(+14.0)
GPT-4		
CoT	46.0	67.3
ToG-R	67.6	81.9
ToG	**69.5**	**82.6**
Gain	(+23.5)	(+15.3)

Small LLM > Big LLM

Table 2: Performances of ToG using different back-bone models on CWQ and WebQSP.

- **Effect of back-bone models**
- **Small LLM + KG > Big LLM**

The Effect of Search Depth and Width

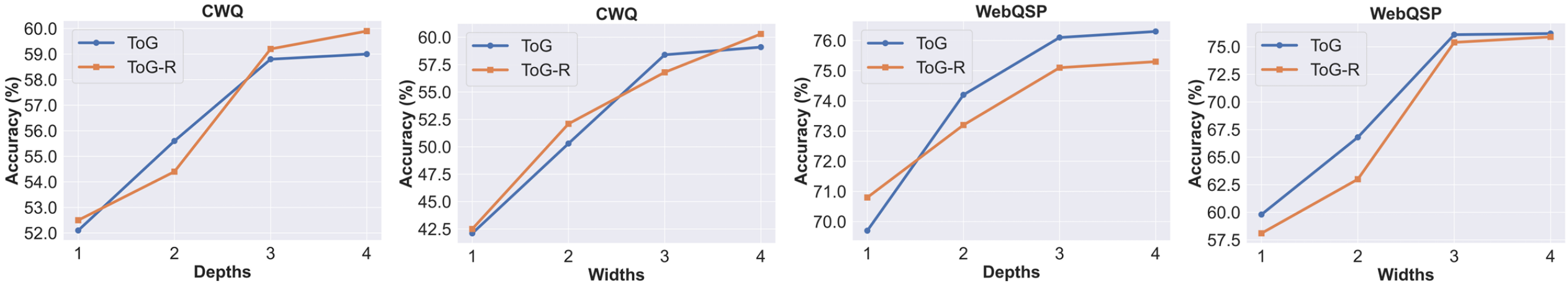
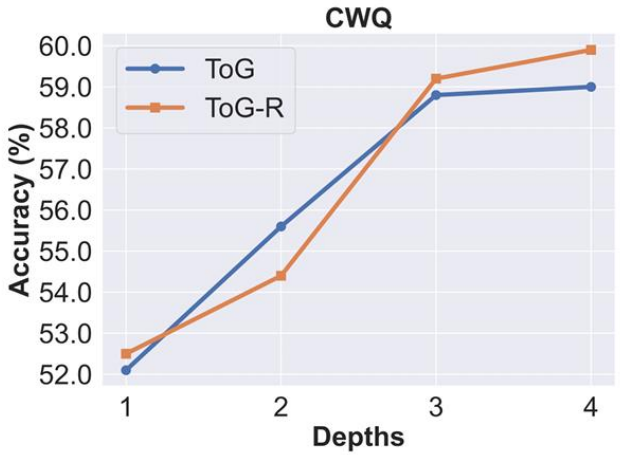
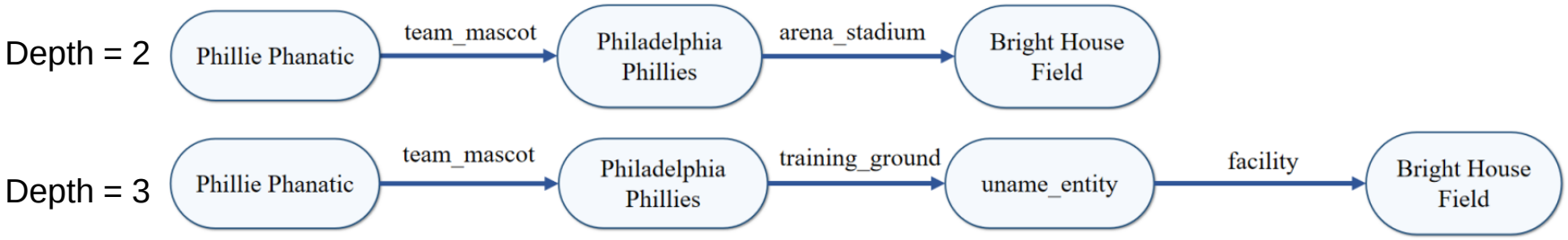


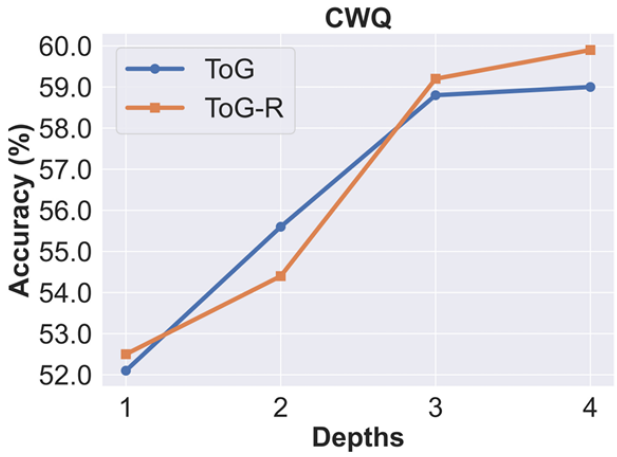
Figure 3: Performance of ToG on different search depth and width.

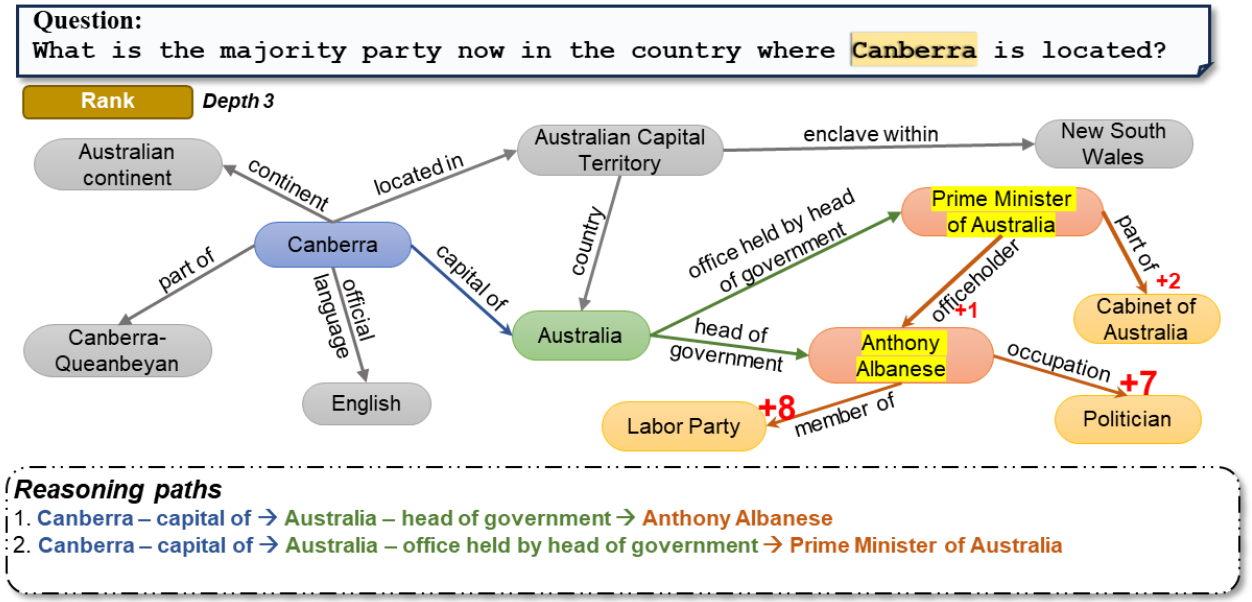
The Effect of Search Depth and Width





The Effect of Search Depth and Width





Width = 2

The Effect of Search Depth and Width

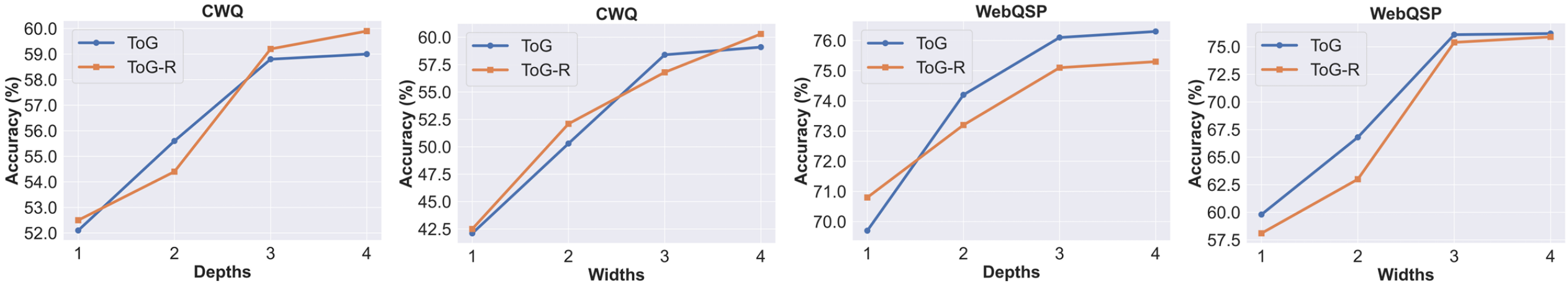


Figure 3: Performance of ToG on different search depth and width.

Where does the knowledge come from?

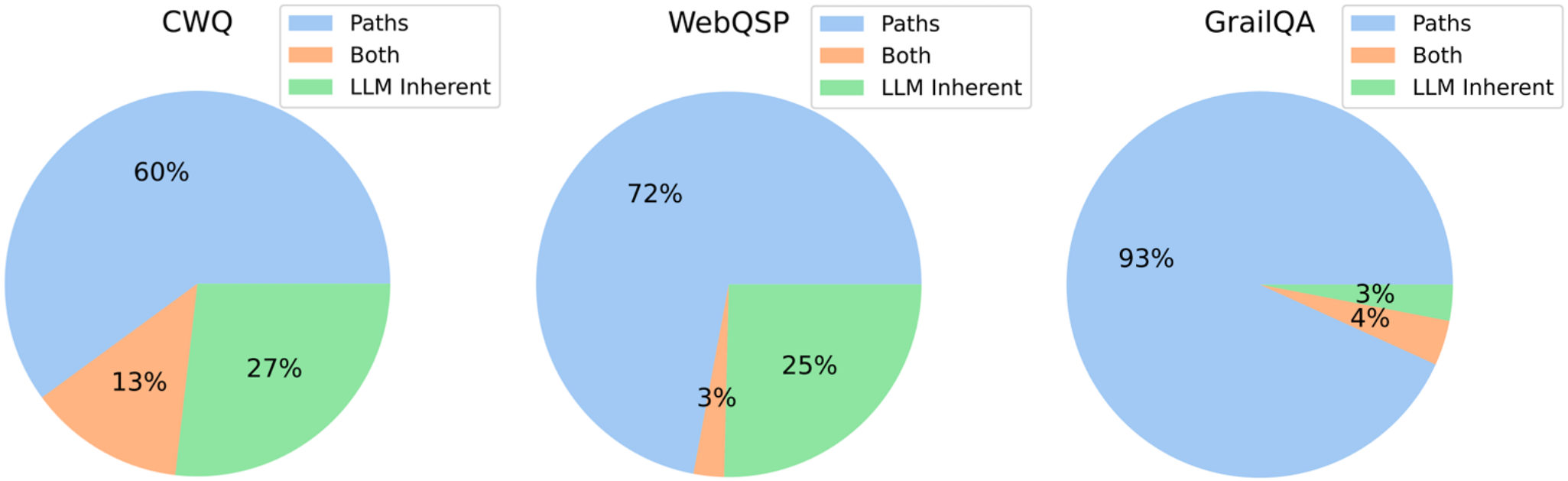


Figure 7: The proportions of ToG's evidence of answers on CWQ, WebQSP, and GrailQA datasets.

Where does the knowledge come from?

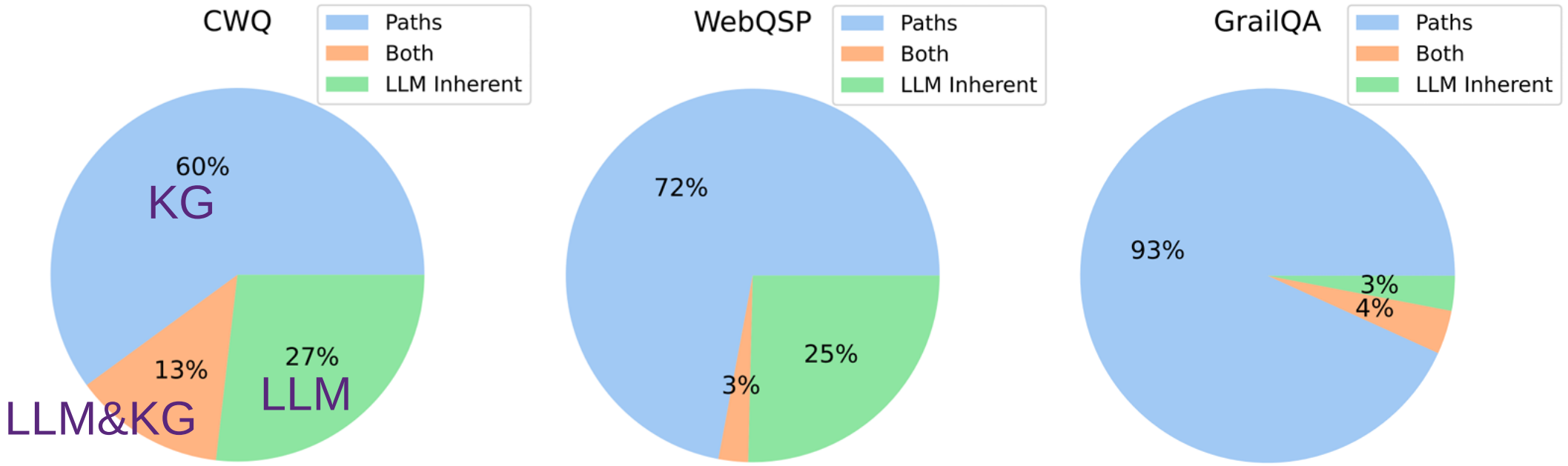
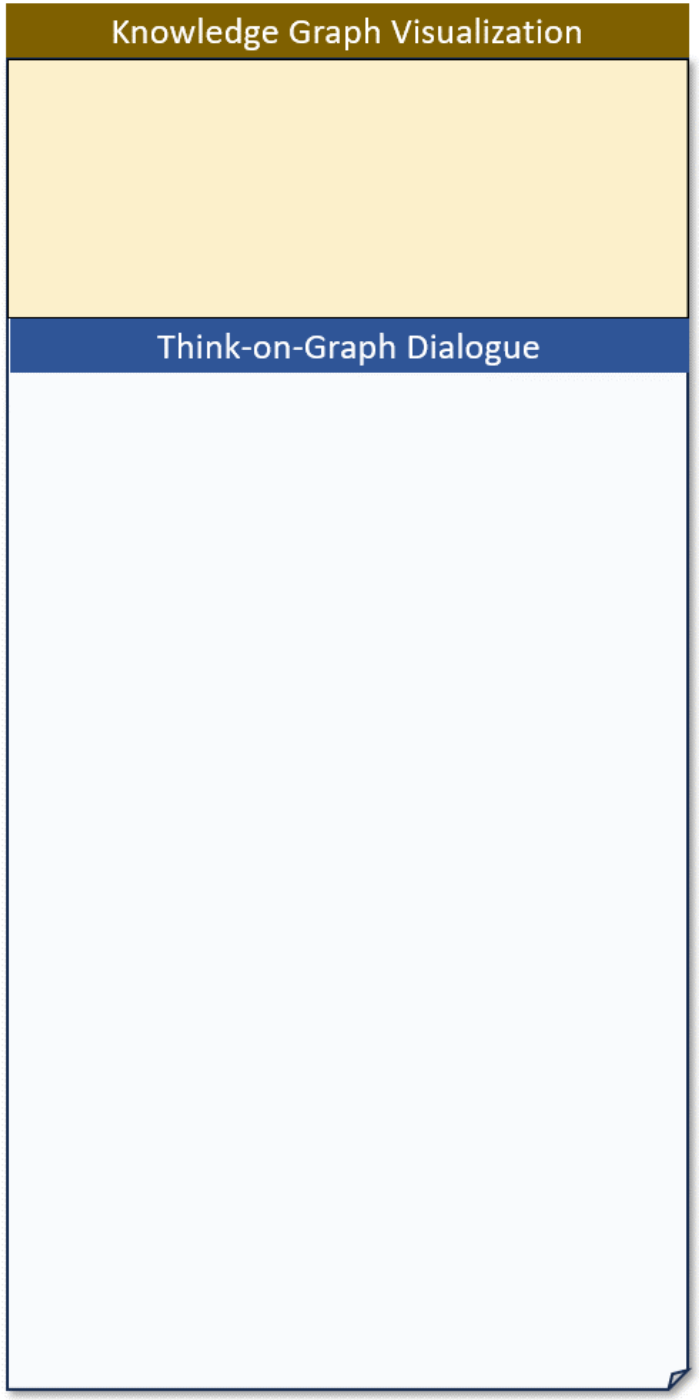


Figure 7: The proportions of ToG's evidence of answers on CWQ, WebQSP, and GrailQA datasets.

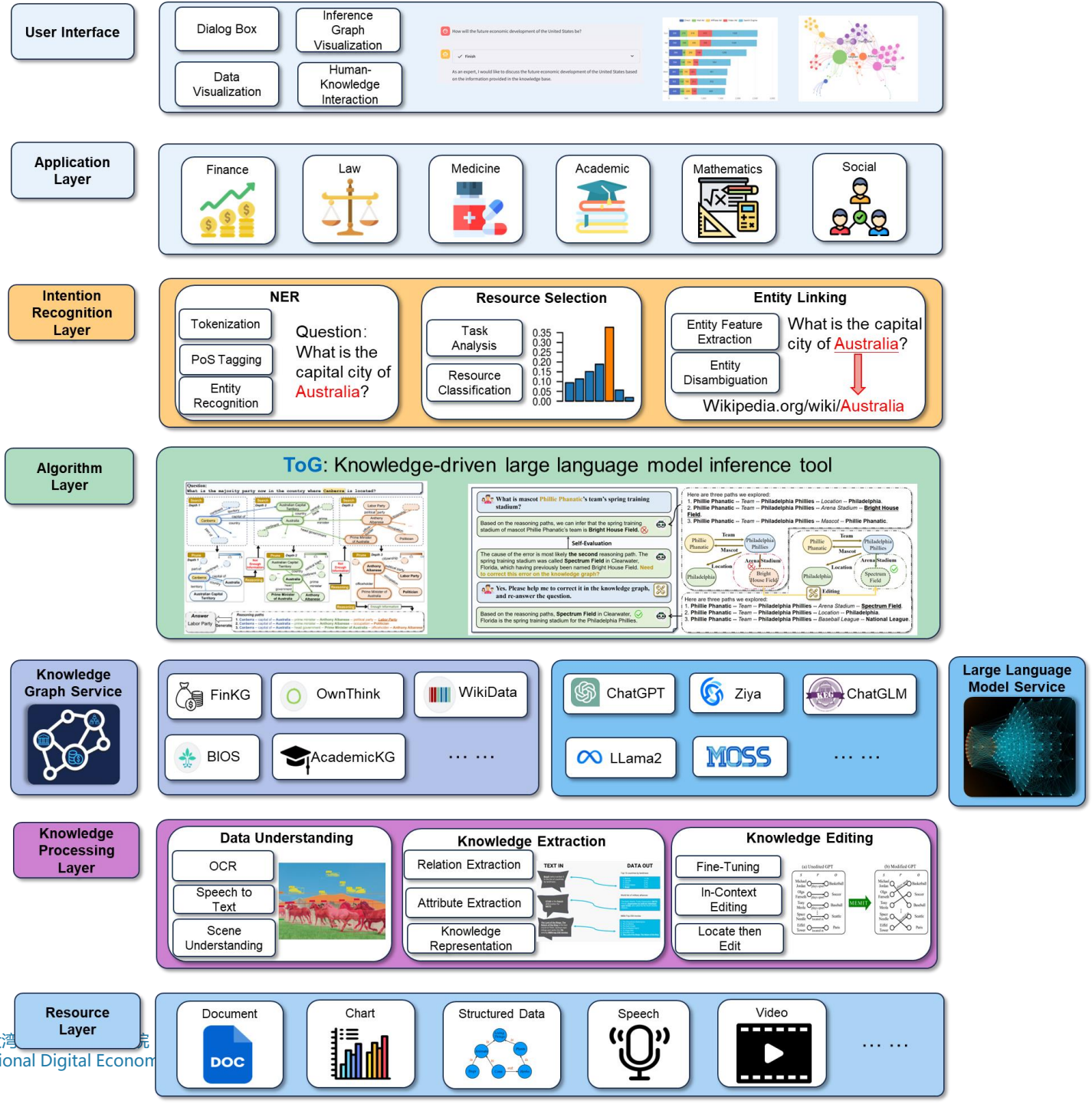
Responsible Reasoning

- Traceability
- Correctability
- Credibility



- **Traceability**
- **Correctability**
- **Credibility**

System Architecture of Think-on-Graph



Let's Think-on-Graph

Let's Think-on-Graph

- Deep Reasoning Path
- Reliability and Credibility
- Plug-and-play

THANK YOU

The world needs a few good IDEAs

中国深圳市福田区市花路5号
长富金茂大厦1号楼3901&57层，邮编518045

Room 3901 & 57/F & 3301 & 3305 & 20/F , Building 1,
Chang FuJin Mao Tower, 5 Shihua Road,
Futian District, ShenZhen, China, 518045

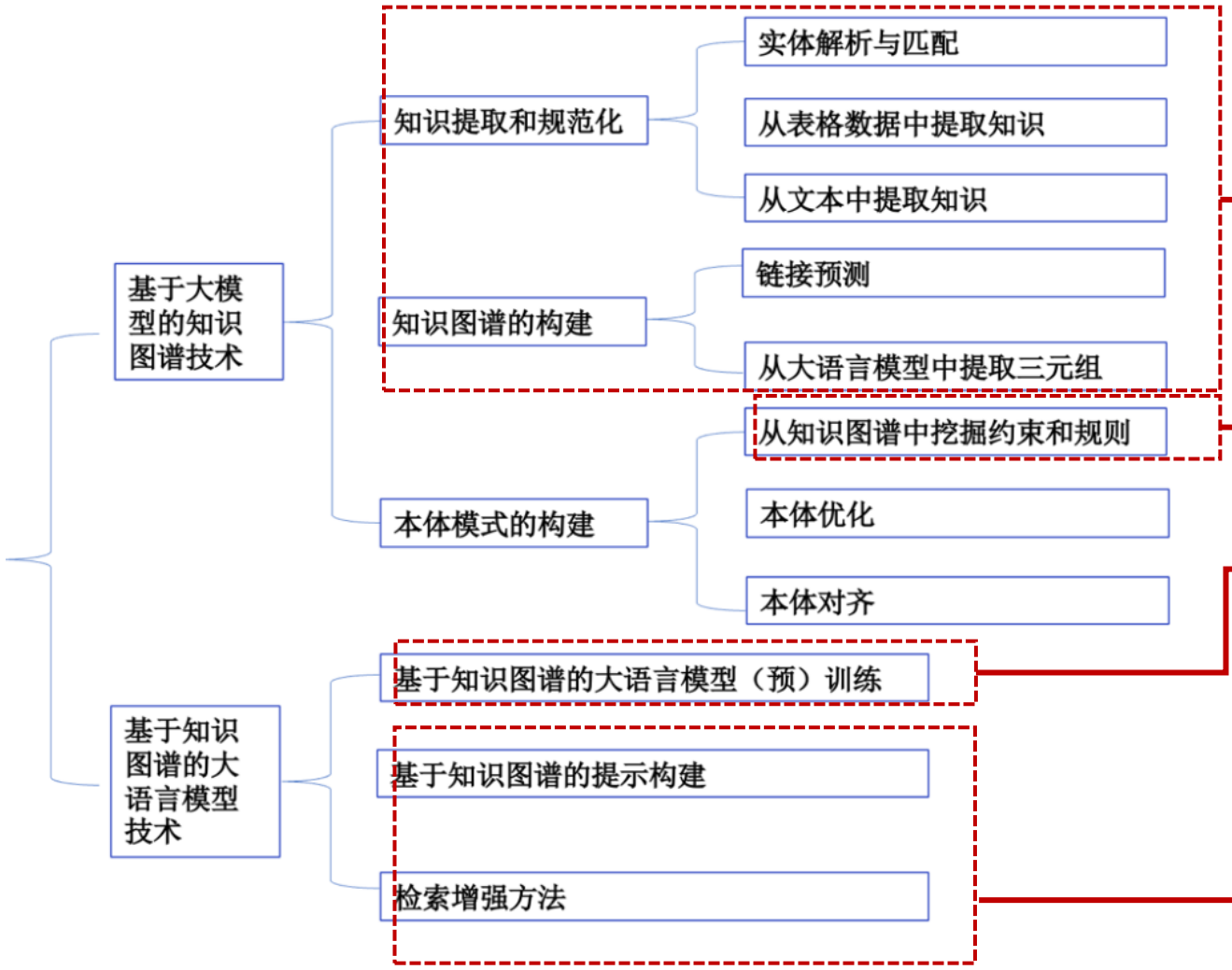
86-755-61610106（3901）
86-755-83219017（57楼）

粤港澳大湾区数字经济研究院
International Digital Economy Academy

www.idea.edu.cn

IDEA WeChat Official

KG+LLM的结合方式



1. LLM-assisted Knowledge Graph Engineering: Experiments with ChatGPT
2. DeepKE-KnowLM
3. ChatIE
4. CodeKGC: Code Language Model for Generative Knowledge Graph Construction
5. TEXT2KGBENCH

ChatRule: Mining Logical Rules with Large Language Models for Knowledge Graph Reasoning

HuaTuo: Tuning LLaMA Model with Chinese Medical Knowledge

1. Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering
2. Few-shot In-context Learning for Knowledge Base Question Answering
3. Chain of Knowledge: A Framework for Grounding Large Language Models with Structured Knowledge Bases
4. StructGPT: A General Framework for Large Language Model to Reason over Structured Data

LLM用于KG构建

LLM用于知识工程的相关实验分析.

研究动机：

缺乏系统性地研究分析LLM是否能够用于Knowledge Engineering；

研究方案：

1. 从知识图谱使用的角度出发，进行了NL2SPARQL的实验；
2. 从知识图谱构建的角度出发，进行了知识获取的实验；

	ChatGPT-3	ChatGPT-4
syntactically correct	5/5	5/5
plausible query structure	4/5	3/5
producing correct result	3/5	2/5
using only defined classes and properties	3/5	4/5
correct usage of classes and properties	5/5	5/5
correct prefix for the graph	5/5	4/5

	ChatGPT-3	ChatGPT-4
syntactically correct	5/5	5/5
plausible query structure	2/5	4/5
producing correct result	0/5	0/5
using only defined classes and properties	1/5	3/5
correct usage of classes and properties	0/5	3/5
correct prefix for mondial graph	0/5	1/5

4.4 Knowledge Extraction from Fact Sheets

As an experiment to evaluate knowledge extraction capabilities, we used PDF fact sheets of 3D printer specifications from different additive manufacturing (AM) vendor websites. The goal is to build a KG about existing 3D printers and their type as well as capabilities. We fed plaintext excerpts (extracted via pdfplumber) from these PDFs into ChatGPT-3 and prompted it to:

Prompt 5: Convert the following $$vendor$$ 3d printer specification into a JSON_LD formatted Knowledge Graph. The node for this KG should be Printer as a main node, Type of 3d printer such as FDM, SLA, and SLS, Manufacturer, Material, Applications, and Technique.

研究结果：

1. 基于LLM做知识工程具有实践意义；
2. 本文的实验过于有限，参考价值业有限，相关文献还需要进一步调研，比如DeepKE-LLM/KnowLM、TechGPT；

LLM-assisted Knowledge Graph Engineering: Experiments with ChatGPT

Lars-Peter Meyer[1,2,3][0000−0001−5260−5181], Claus Stadler[1,2,3][0000−0001−9948−6458], Johannes Frey[1,2,3][0000−0003−3127−0815], Norman Radtke[1,2][0000−0001−9155−8920], Kurt Junghanns[1,2][0000−0003−1337−2770], Roy Meissner[2,3][0000−0003−4193−8209], Gordian Dziwis[1][0000−0002−9592−418X], Kirill Bulert[1,2][0000−0002−1459−3754], and Michael Martin[1,2][0000−0003−0762−8688]

[1] InfAI e.V. Leipzig, Germany, lpmeyer@infai.org, https://www.infai.org
[2] AKSW research group, https://aksw.org
[3] Leipzig University, Germany, https://www.uni-leipzig.de

- Assistance in knowledge graph usage:
  - Generate SPARQL queries from natural language questions (related experiment in Section 4.1 and Section 4.3)
  - Exploration and summarization of existing knowledge graphs (related experiment in Section 4.5)
  - Conversion of competency questions to SPARQL queries
  - Code generation or configuration of tool(chain)s for data pipelines
- Assistance in knowledge graph construction
  - Populating knowledge graphs (related experiment in Section 4.4) and vice versa
  - Creation or enrichment of knowledge graph schemas / ontologies
  - Get hints for problematic graph design by analysing ChatGPT usages problems with a knowledge graph
  - Semantic search for concepts or properties defined in other already existing knowledge graphs
  - Creation and adjustment of knowledge graphs based on competency questions

LLM用于KG构建

LLM用于知识抽取的相关开源项目（基于指令微调）.

InstructUIE: Multi-task Instruction Tuning for Unified Information Extraction:

InstructUIE: Multi-task Instruction Tuning for Unified Information Extraction

Xiao Wang★*, Weikang Zhou★*, Can Zu★, Han Xia★, Tianze Chen★,
Yuansen Zhang★, Rui Zheng★, Junjie Ye★, Qi Zhang★†, Tao Gui♦ †,
Jihua Kang♣, Jingsheng Yang♣, Siyuan Li♣, Chunsai Du♣,
★ School of Computer Science, Fudan University, Shanghai, China
♦ Institute of Modern Languages and Linguistics, Fudan University, Shanghai, China
♣ ByteDance Inc.
{xiao_wang20,qz,tgui}@fudan.edu.cn

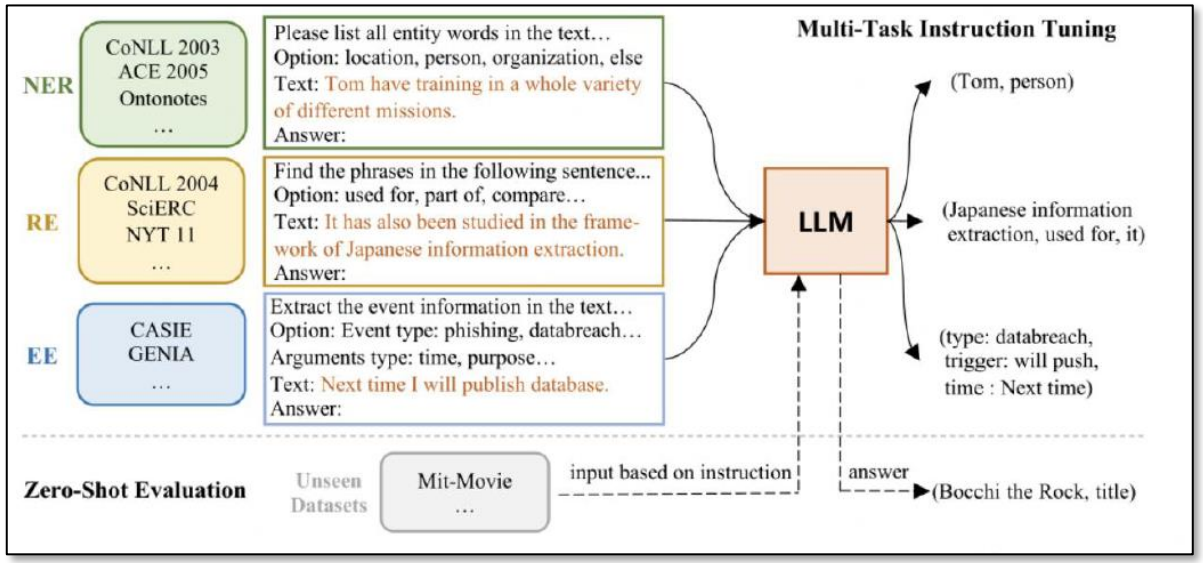
https://github.com/BeyonderXX/InstructUIE

引入一个基于多任务指令调优的统一信息提取框架，名为InstructUIE，将IE任务重新表述为一个自然语言生成问题。对于源语句，设计了描述性指令，使模型能够理解不同的任务，并采用包括所有候选类别的选项机制作为输出空间的约束。

然后，需要预先训练的语言模型以自然语言的形式生成目标结构和相应的类型

模型基座

基于**Flan T5**进行指令微调；




```
"""
Instruction example:
{
  "RE": [
    {"instruction_type": "zero-shot", "instruction": "Given a phrase that describes the
    two words, extract the words and the lexical relationship between them.
    The output format should be :[(word1, relation, word2)]. \n"},
  ],
  "NER": [
    {"instruction_type": "zero-shot", "instruction": "Please list all entity words in t
    fit the category.Output format is [(word1, type1), (word2, type2))]. \n"},
  ],
  "EE": [
    {"instruction_type": "zero-shot", "instruction": "Extract the event information in
    and return them in the event list. \n"}
  ]
}
"""
```

LLM用于KG构建

LLM用于知识抽取的相关开源项目（基于Prompt）.

ChatIE: Zero-Shot Information Extraction via Chatting with ChatGPT

Zero-Shot Information Extraction via Chatting with ChatGPT

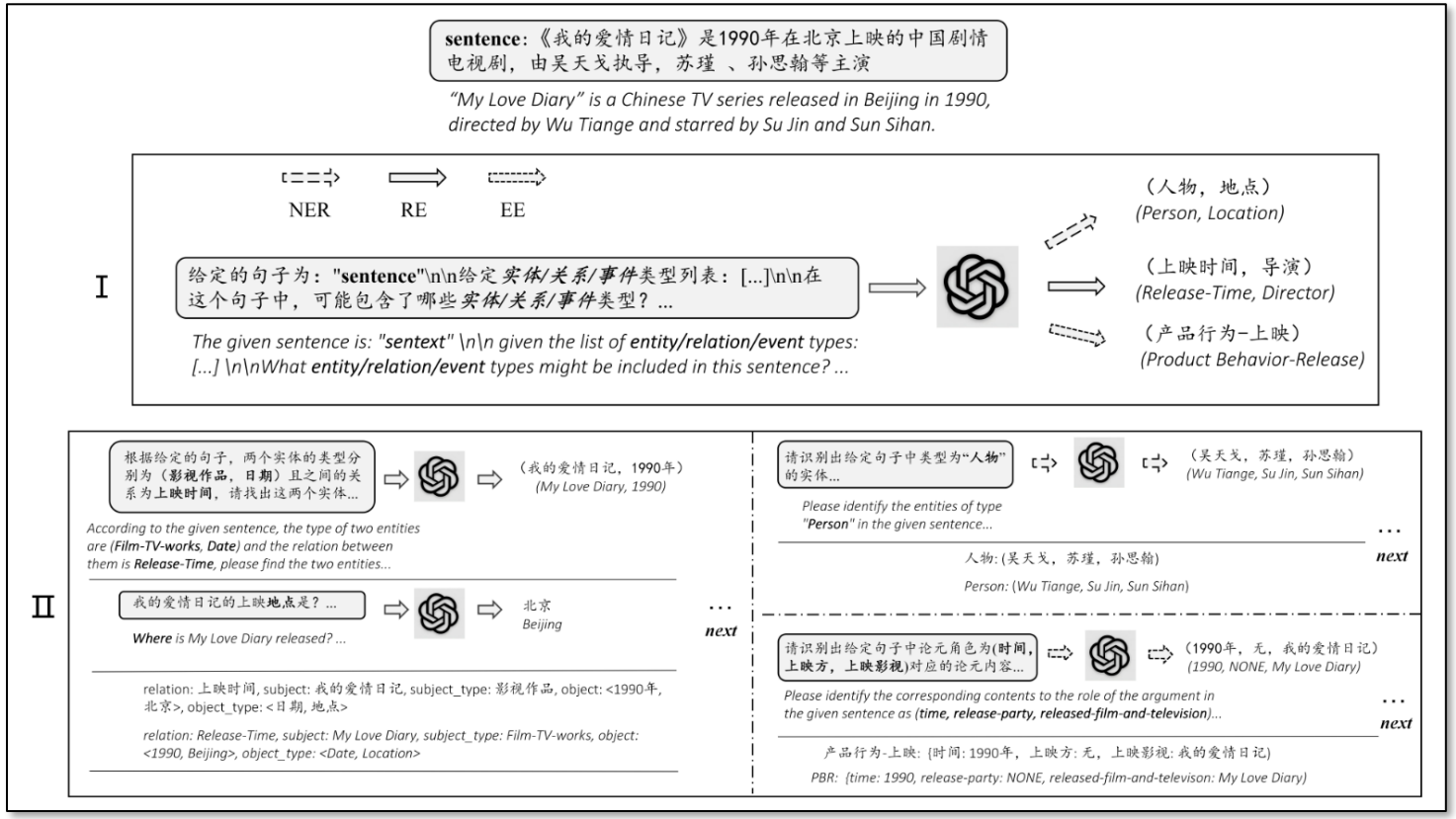
Xiang Wei[1], Xingyu Cui[1], Ning Cheng[1], Xiaobin Wang[2], Xin Zhang, Shen Huang[2], Pengjun Xie[2], Jinan Xu[1], Yufeng Chen[1], Meishan Zhang, Yong Jiang[2], and Wenjuan Han[1]

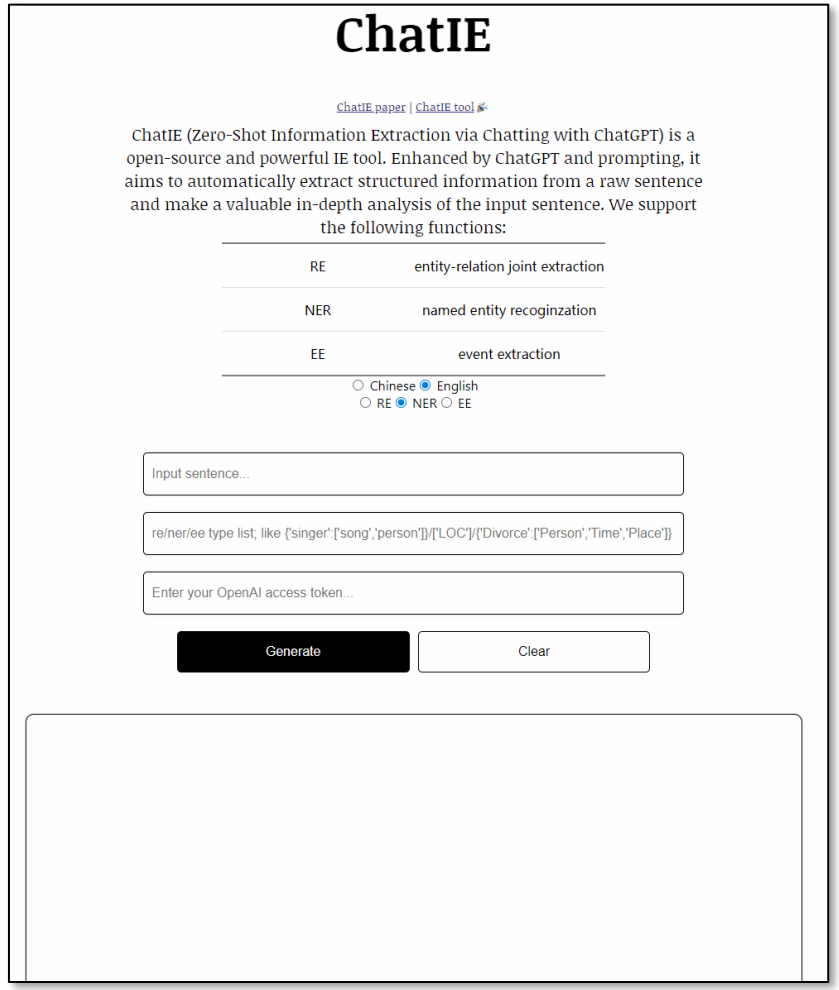
[1] Beijing Jiaotong University, Beijing, China

[2] DAMO Academy, Alibaba Group, China

https://github.com/zjunlp/DeepKE/tree/main/example/llm

将零样本IE任务转变为一个两阶段框架的多轮问答问题（Chat IE）,通过制定实体关系三元组抽取、命名实体识别和事件抽取任务，并为每个任务设计2个步骤的prompt-pattern，多轮抽取





LLM用于KG构建

LLM结合Code范式用于知识图谱构建的论文.

CodeKGC: Code Language Model for Generative Knowledge Graph Construction:

https://github.com/zjunlp/DeepKE/tree/main/example/llm

本文提出一种基于Code形式用于构建KG的范式**CodeKGC**，用结构化地代码去表示知识图谱，将Text2KG转化为Text2Code问题，利用LLM对于Code的感知能力，使得LLM能够捕获文本中的知识以及结构信息用于知识抽取。

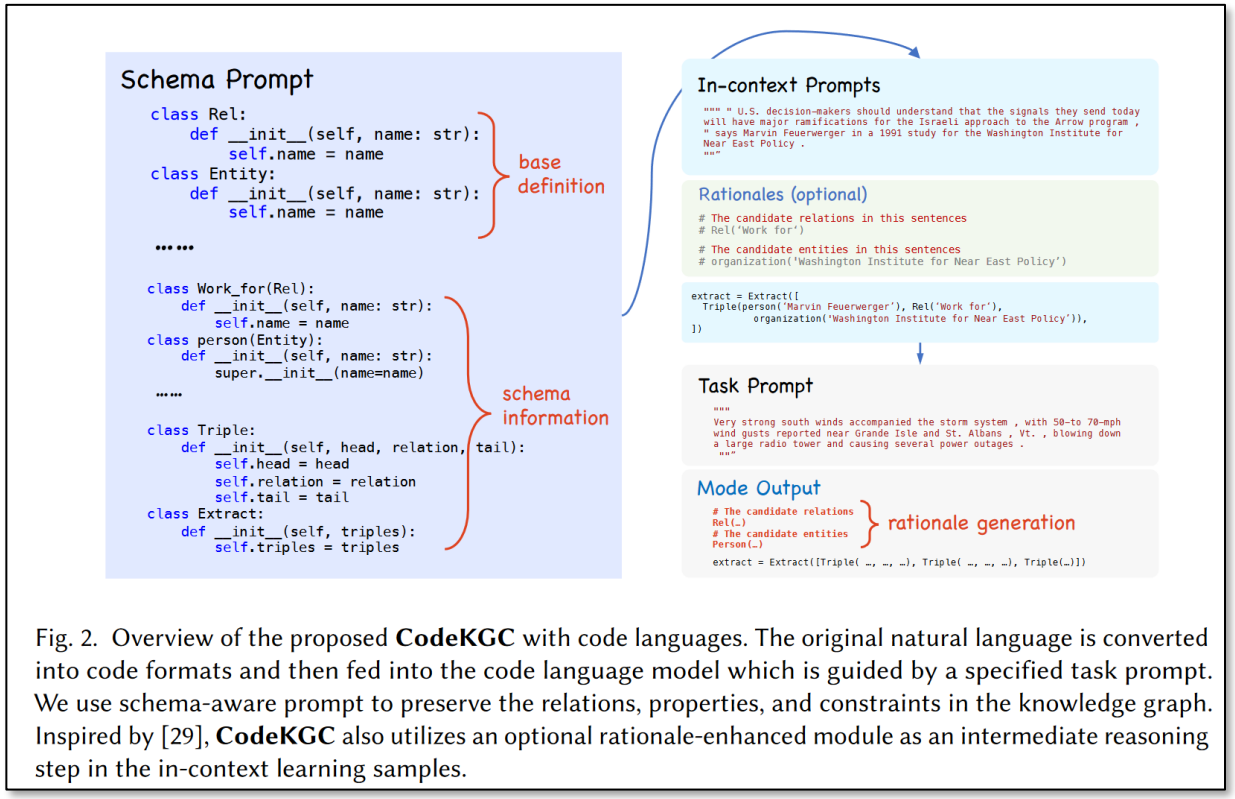


Fig. 2. Overview of the proposed **CodeKGC** with code languages. The original natural language is converted into code formats and then fed into the code language model which is guided by a specified task prompt. We use schema-aware prompt to preserve the relations, properties, and constraints in the knowledge graph. Inspired by [29], **CodeKGC** also utilizes an optional rationale-enhanced module as an intermediate reasoning step in the in-context learning samples.

CodeKGC: Code Language Model for Generative Knowledge Graph Construction

ZHEN BI* and JING CHEN*, Zhejiang University, China
YINUO JIANG, Zhejiang University, China
FEIYU XIONG, Alibaba Group, China
WEI GUO, Alibaba Group, China
HUAJUN CHEN, Zhejiang University, China
NINGYU ZHANG†, Zhejiang University, China

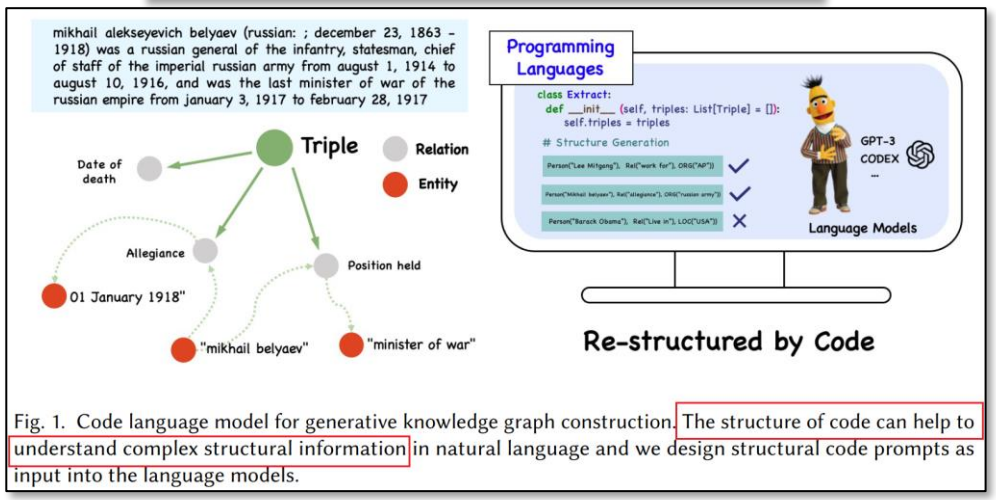


Fig. 1. Code language model for generative knowledge graph construction. The structure of code can help to understand complex structural information in natural language and we design structural code prompts as input into the language models.

```
"""
U.S. decision-makers should understand that the signals they
send today will have major ramifications for the Israeli
approach to the Arrow program , " says Marvin Feuerwerger in
a 1991 study for the Washington Institute for Near East
Policy.
"""
```

Rationale as intermediate steps (optional)

Step 1 relation identification from text

```
# The candidate relations in this sentences
# person('Marvin Feuerwerger')
# Rel('Work for')
```

Step 2 entitiy extraction with relations

```
# The candidate entities in this sentences
# person('Marvin Feuerwerger')
# organization('Washington Institute for Near East Policy')
```

Step 3 final result generation

```
extract = Extract([Triple(person('Marvin Feuerwerger'),
Rel('Work for'), organization('Washington Institute
for Near East Policy'))])
```

KG用于LLM推理

LLM使用实体在KG中的一阶知识辅助推理

KAPING：

https://arxiv.org/pdf/2306.04136.pdf

识别句子中的实体

在KG中搜索包含句子中实体的三元组

通过三元组和句子的相似度，筛选一定数量的三元组

将筛选后的三元组作为上下文，构成新的prompt，输入到LLM中，用以辅助推理

Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering

Jinheon Baek[1*] **Alham Fikri Aji**[2] **Amir Saffari**[3]
KAIST[1] MBZUAI[2] Amazon[3]
jinheon.baek@kaist.ac.kr alham.fikri@mbzuai.ac.ae amsafari@amazon.com

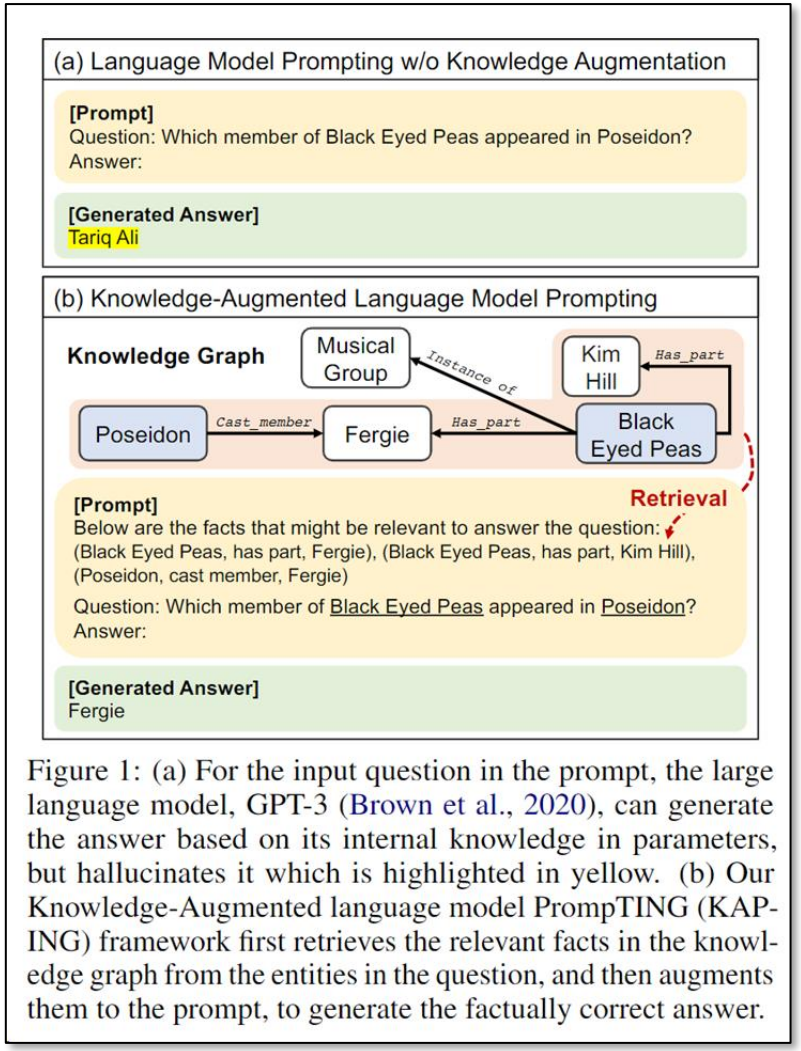


Figure 1: (a) For the input question in the prompt, the large language model, GPT-3 (Brown et al., 2020), can generate the answer based on its internal knowledge in parameters, but hallucinates it which is highlighted in yellow. (b) Our Knowledge-Augmented language model PrompTING (KAPING) framework first retrieves the relevant facts in the knowledge graph from the entities in the question, and then augments them to the prompt, to generate the factually correct answer.

KG用于LLM推理

使用LLM生成准确的KG查询语句

KB-Binder:
https://github.com/ltl3A87/KB-BINDER

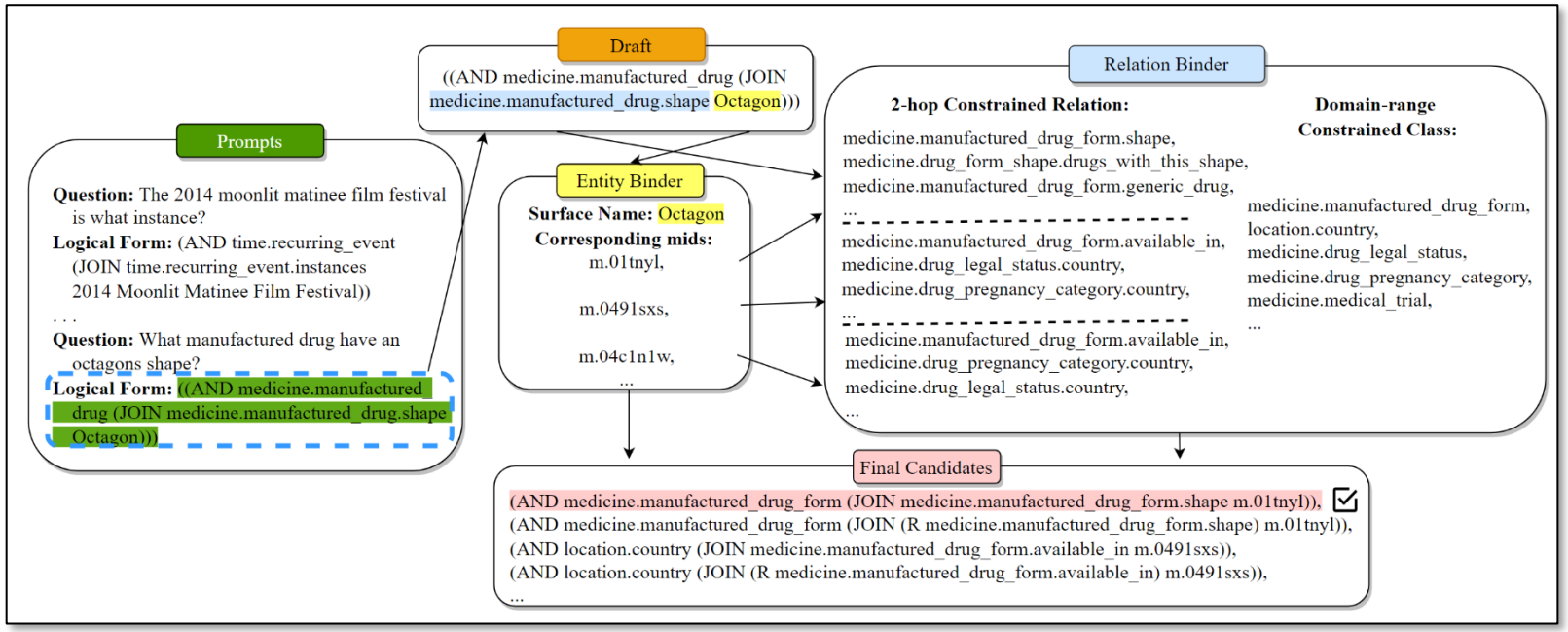
使用LLM将问题转化为SPARQL查询语句

利用语义相似度根据KG中的实体和关系对查询语句中的实体关系进行修改

执行修改后的SPARQL语句

Few-shot In-context Learning for Knowledge Base Question Answering

♠Tianle Li, ♠Xueguang Ma, ♠Alex Zhuang, ♥Yu Gu, ♥Yu Su, ♠♥Wenhu Chen
♠University of Waterloo
♥The Ohio State University
♠Vector Institute, Toronto
{t29li,x93ma,a5zhuang,wenhuchen}@uwaterloo.ca, {gu.826,su.809}@osu.edu



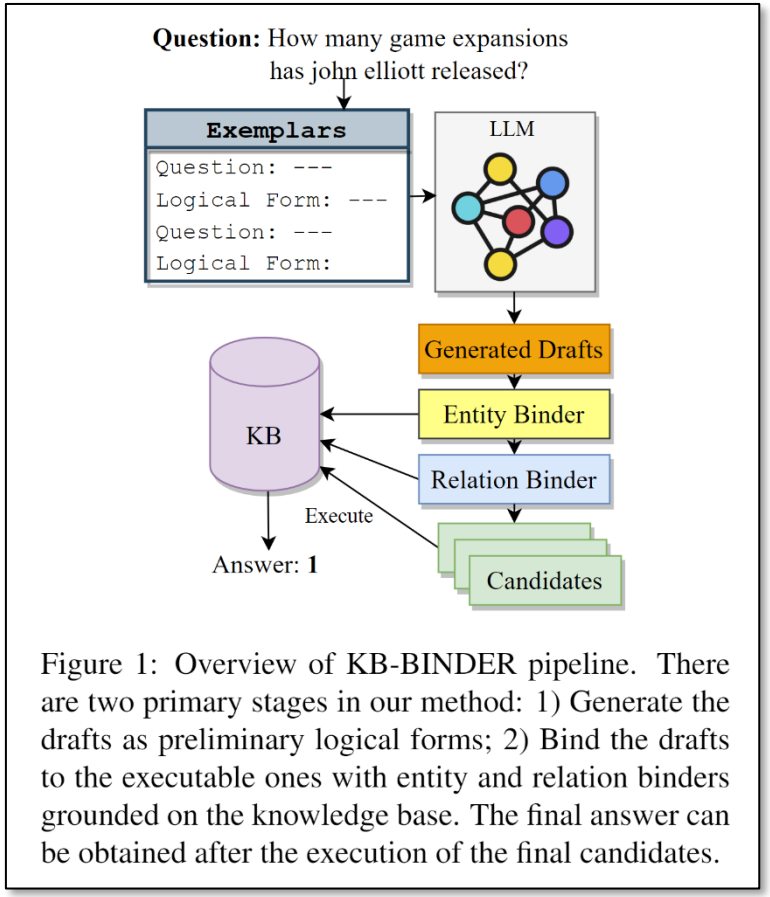


Figure 1: Overview of KB-BINDER pipeline. There are two primary stages in our method: 1) Generate the drafts as preliminary logical forms; 2) Bind the drafts to the executable ones with entity and relation binders grounded on the knowledge base. The final answer can be obtained after the execution of the final candidates.

KG用于LLM推理

使用LLM生成准确的KG查询语句

Chain-of-Knowledge：

https://arxiv.org/pdf/2305.13269v1.pdf

使用LLM将问题分解成若干个子问题

使用一个Query Generator将子问题转化为简单的SPARQL查询

将查询得到的三元组作为上下文，构成新的prompt输入到LLM中，以辅助LLM推理

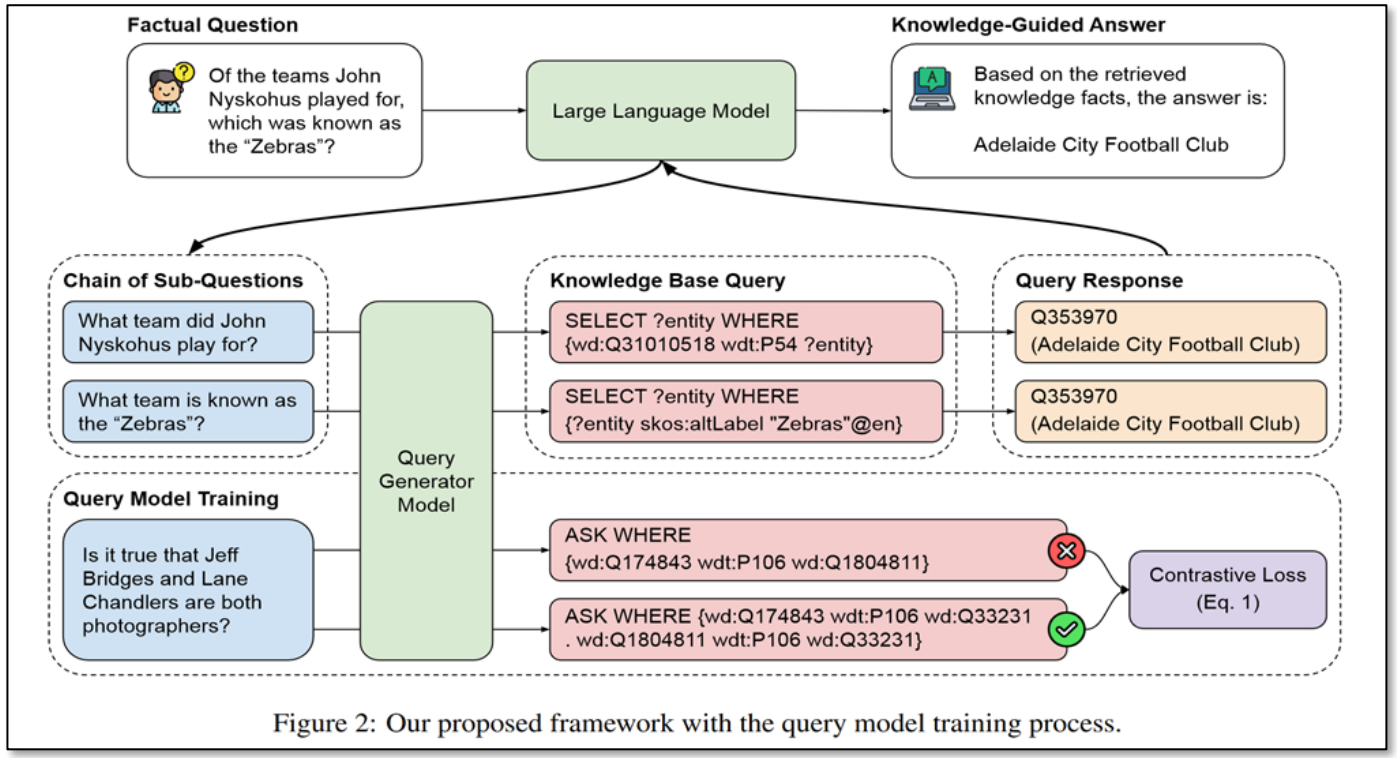
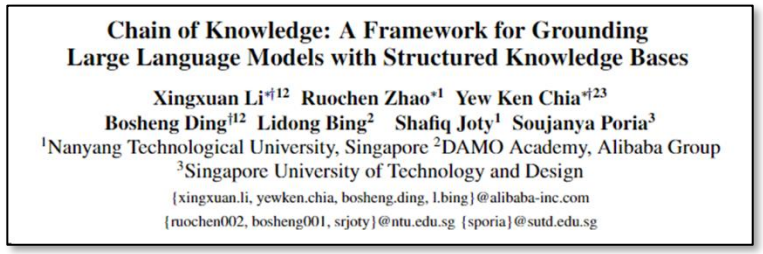


Figure 2: Our proposed framework with the query model training process.



Chain of Knowledge: A Framework for Grounding Large Language Models with Structured Knowledge Bases

Xingxuan Li*†12 Ruochen Zhao*1 Yew Ken Chia*†23
Bosheng Ding†12 Lidong Bing2 Shafiq Joty1 Soujanya Poria3
[1]Nanyang Technological University, Singapore [2]DAMO Academy, Alibaba Group
[3]Singapore University of Technology and Design
{xingxuan.li, yewken.chia, bosheng.ding, l.bing}@alibaba-inc.com
{ruochen002, bosheng001, srjoty}@ntu.edu.sg {sporia}@sutd.edu.sg

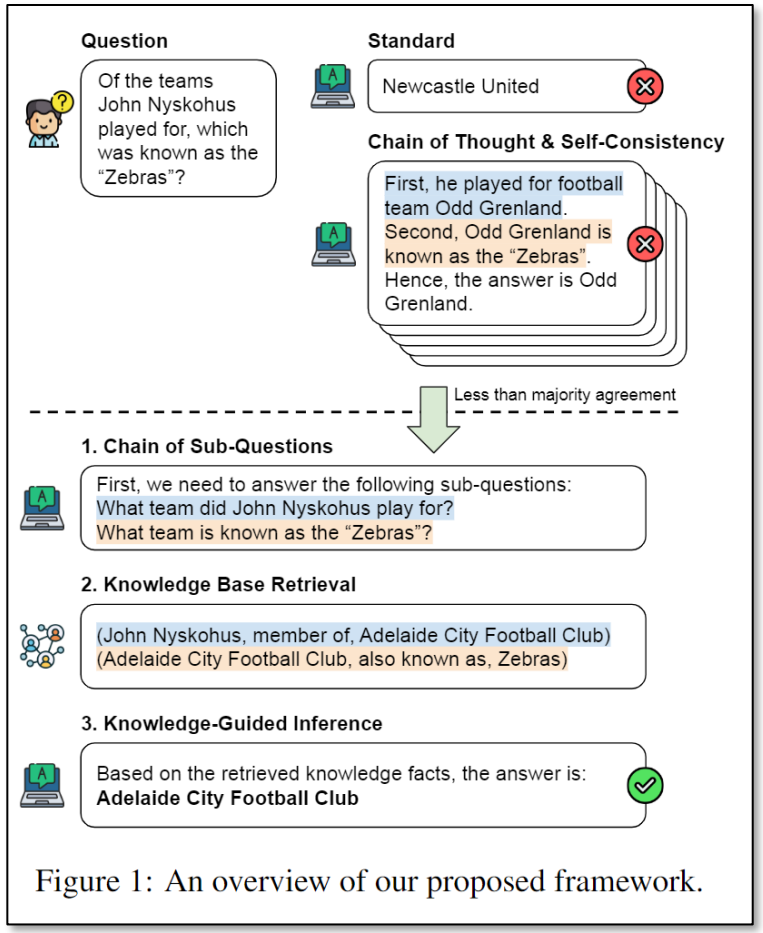


Figure 1: An overview of our proposed framework.

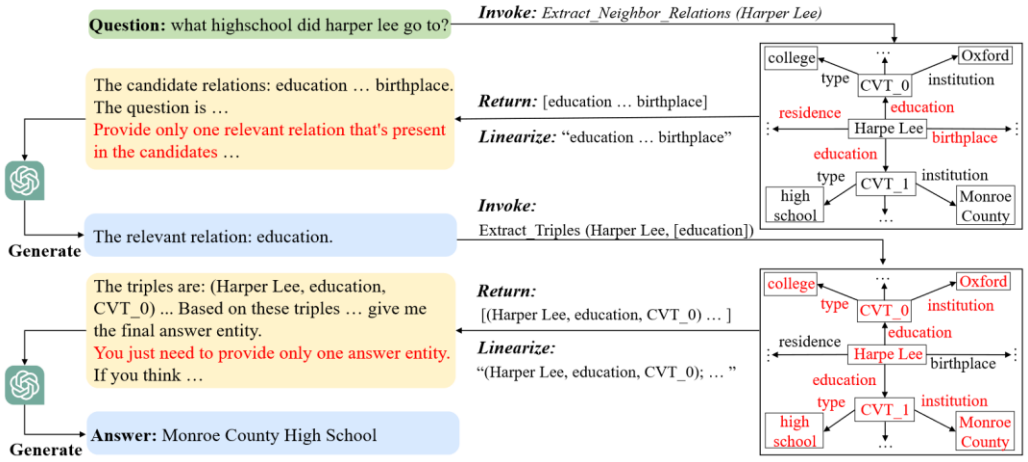
KG用于LLM推理

LLM在KG上做迭代式知识推理

StructGPT：

https: //github.com/RUCAIBox/StructGPT

对于知识图谱、表格、数据库三类结构信息较为重要的数据结构，本文提出了**StructGPT**架构，其采用Iterative Reading then-Reasoning (IRR)即**迭代阅读-推理方法。**



(a) Case of WebQSP (KGQA)

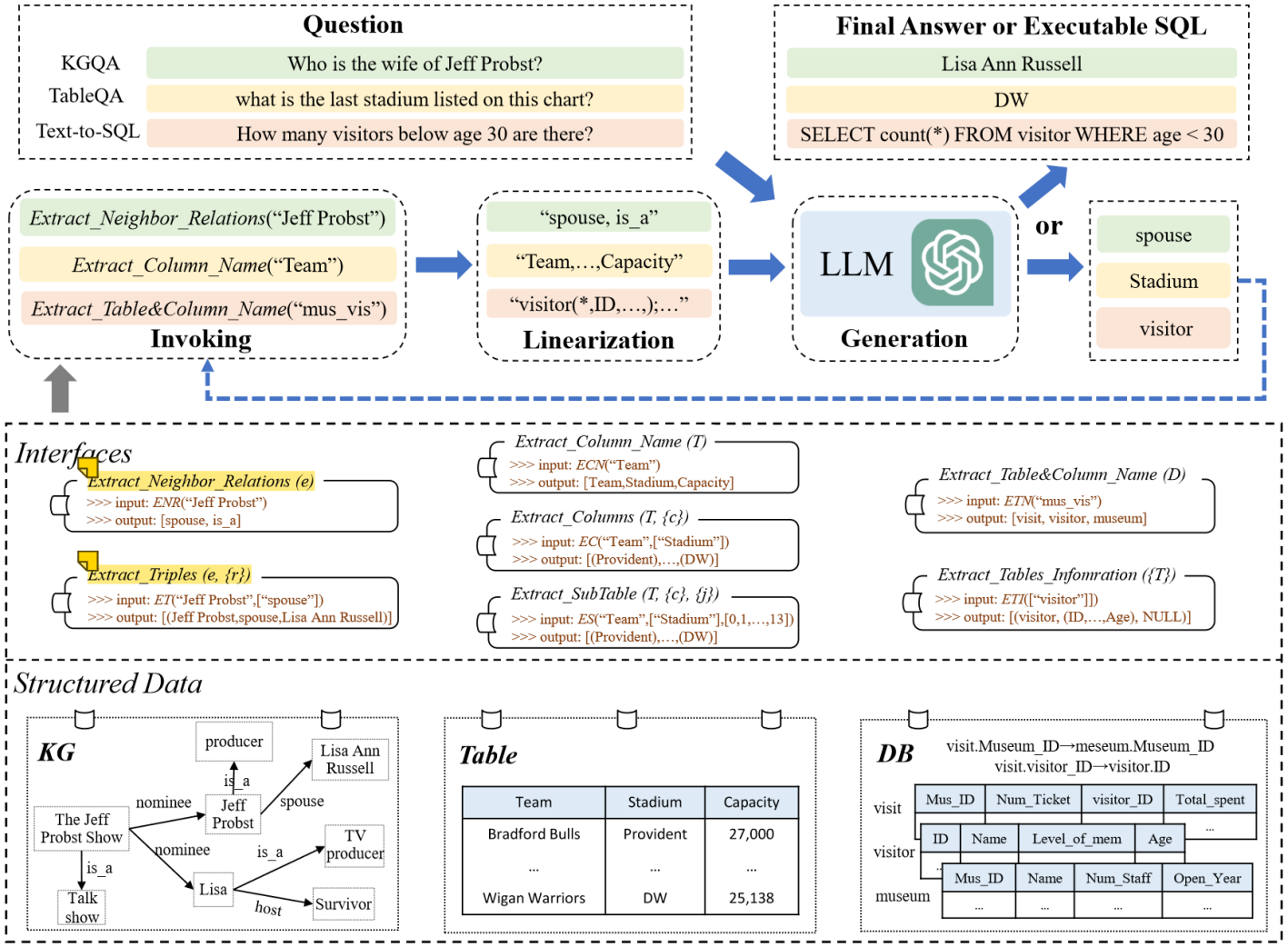
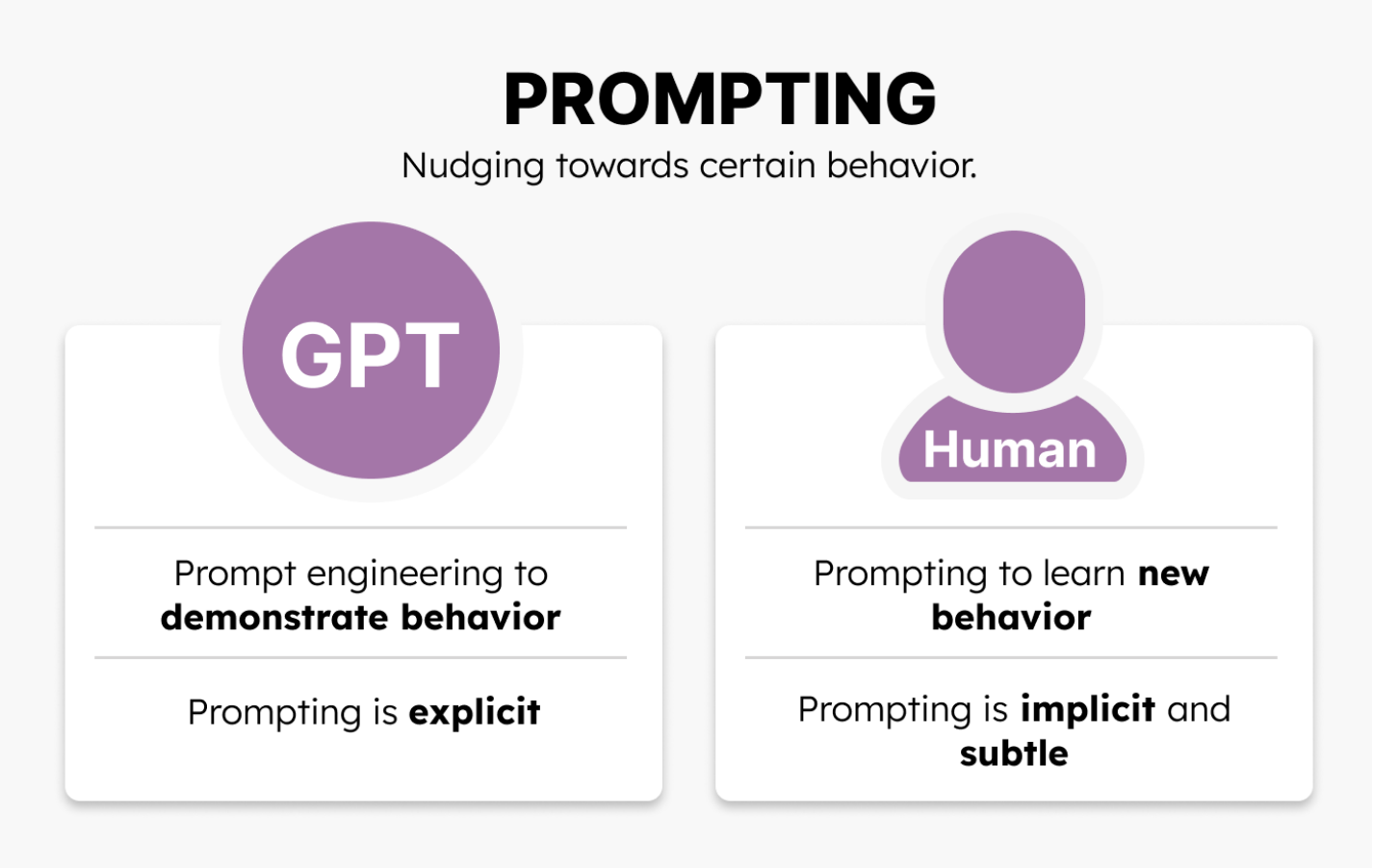


Figure 1: The overview of the proposed iterative reading-then-reasoning approach. We design specialized interfaces for reading structured data, and iterate the invoking-linearization-generation procedure to utilize LLMs for performing reasoning on the interfaces, until deriving the final answer or executable SQL.

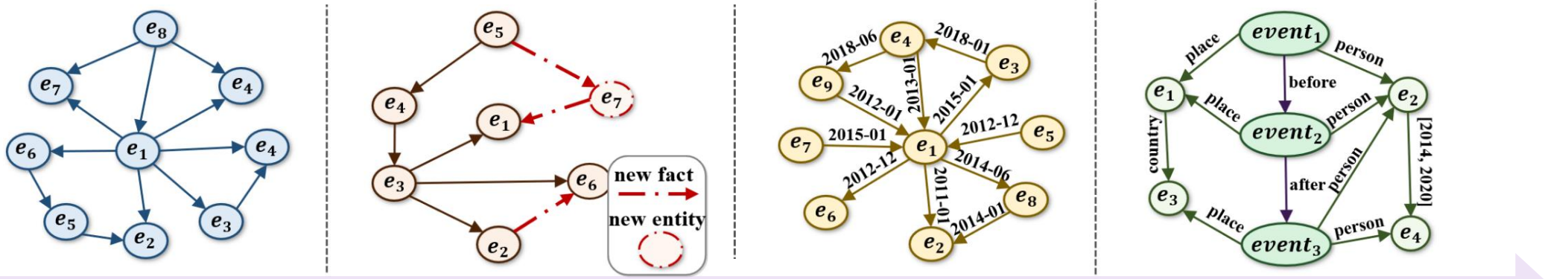
Incorporating KG via Prompting



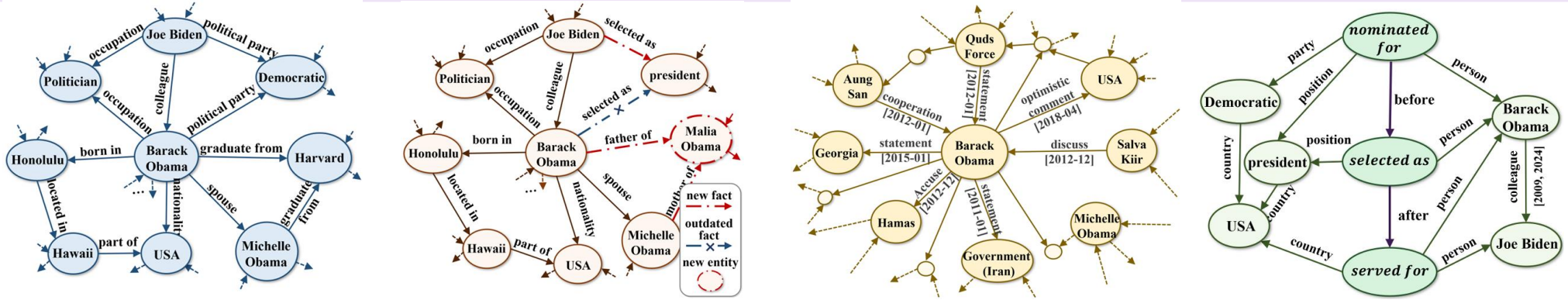
Prompting

Evolution of Knowledge Graph

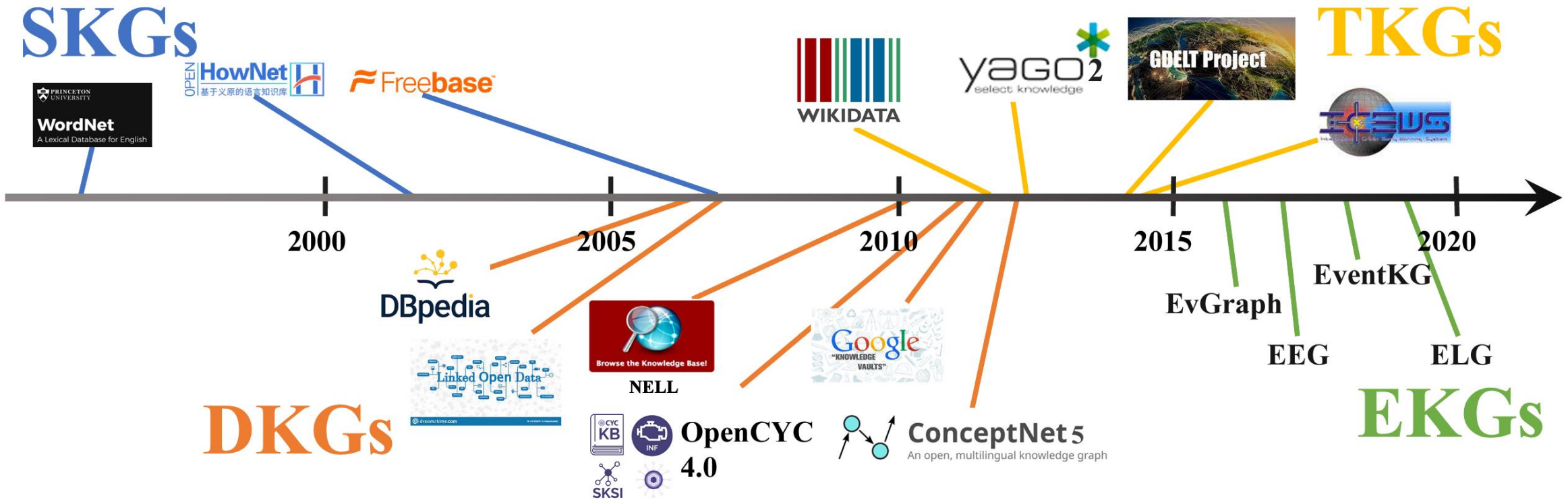
Concept



Example



History of Knowledge Graphs



Neuron connections in Human Brain ~ 100T

2012
AlexNet
60M Parameters

2018
BERT
100M Parameters

2020
GPT-3
175B Parameters

2023
GPT-4
1.7T Parameters

Performance against Explainability

Deep Learning

Residual Network

LSTM

Transformer

Mixture-of-Expert

Vanilla RNN

CRF

HMM

Sequence Learning

Kernel SVM

Gaussian Process

Kernel k-NN

Kernel Learning

Ensemble Learning

Boosting

Random Forest

Rule-based Learning

Symbolic Regression

Bayesian Network

Decision Tree

Linear SVM

Generalized Additive Model

Generalized Linear Model

Linear Model

High

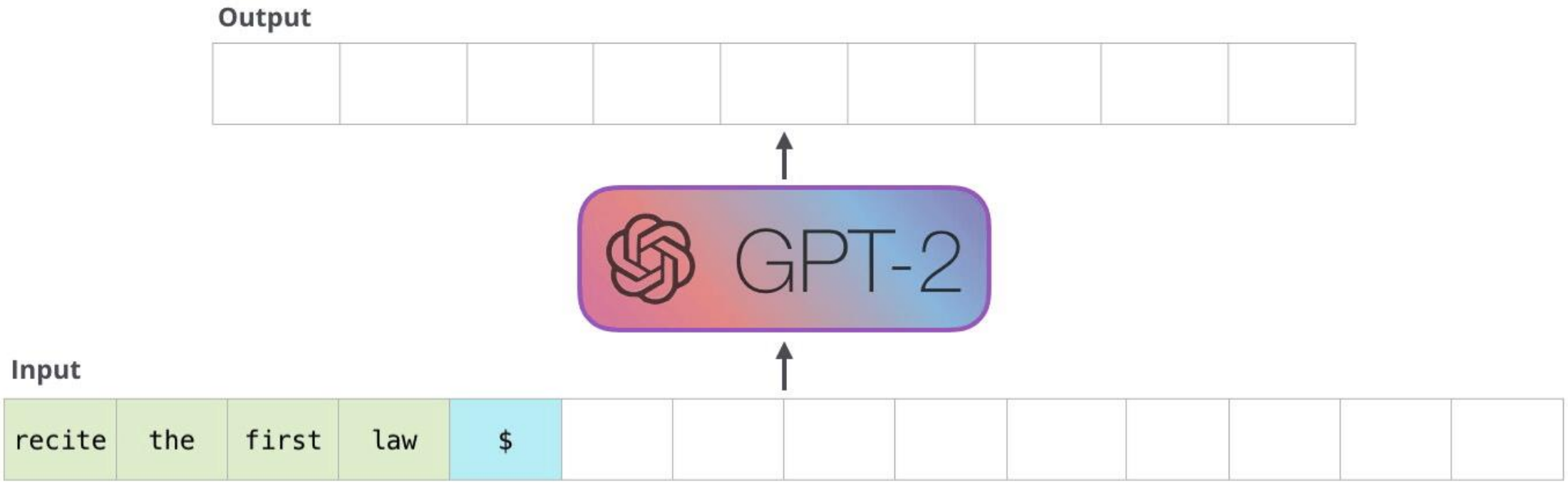
Prediction Performance

Low

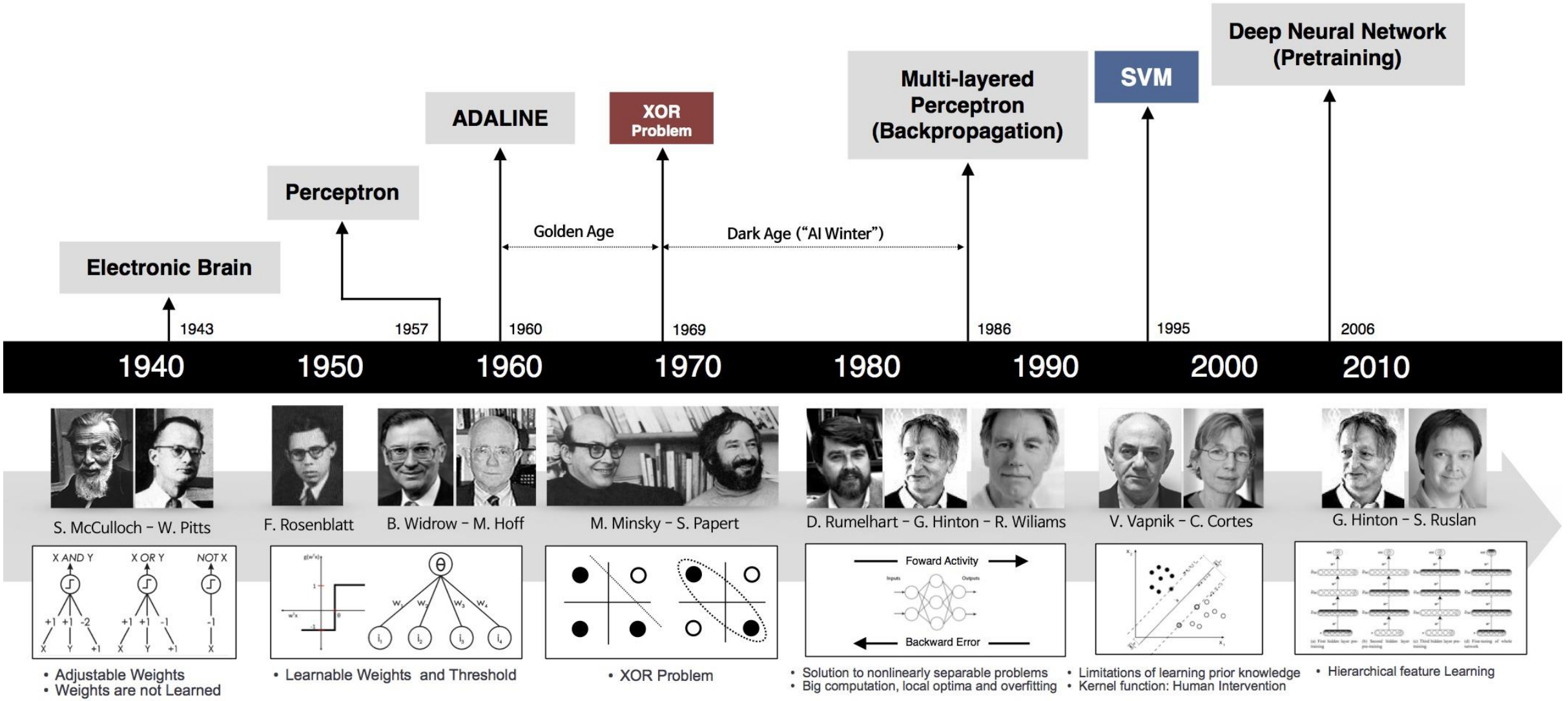
Model Explainability

High

Principle of GPT



History of Machine Learning



自动化构建金融行为知识图谱