Paths-over-Graph (PoG) : Knowledge Graph Empowered Large Language Model Reasoning

Xingyu Tan[1,2], Xiaoyang Wang[1,*], Qing Liu[2], Xiwei Xu[2], Xin Yuan[2], Wenjie Zhang[1]

[1] University of New South Wales, [2] Data61, CSIRO Email: xingyu.tan@unsw.edu.au

Introduction & Motivation

❑ **LLM Challenges:** LLMs struggle with complex reasoning, staying updated with current knowledge, and hallucination. Existing plan-retrieval-answering methods rely heavily on LLM reasoning rather than knowledge faithfulness.

❑ **KG Integration & Motivation** : KGs provide structured facts to enhance LLMs. Current KG-LLM methods face challenges:

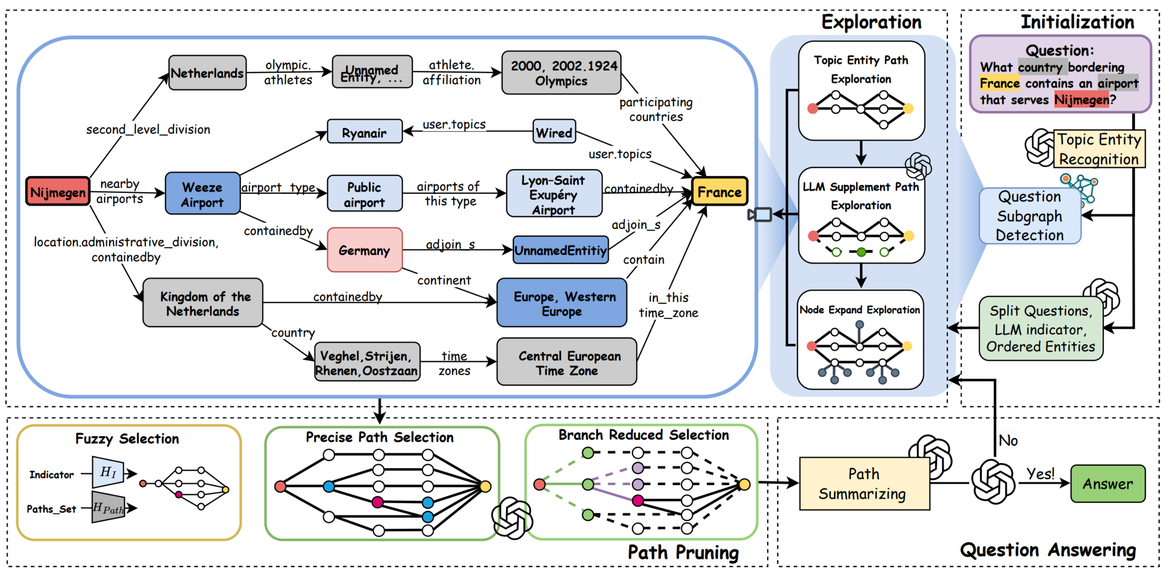
- **Multi-hop Reasoning:** Difficulty reasoning over multiple steps in the KG, often leading to **incorrect answers that is only local optimal instead of global optimal.**
- **Multi-entity Questions:** Existing methods often explore KG for each entity separately, ignoring interconnections and retrieving irrelevant information.
- **Utilizing Graph Structure:** Many methods overlook graph structure, converting KG info to text, which can overwhelm LLMs and lose structural insights.

❑ **Our Contribution:**

- **Dynamic deep search:** Guided by LLMs, PoG dynamically **extracts multi-hop reasoning paths** from KGs, enhancing LLM capabilities in complex knowledge-intensive tasks.
- **Interpretable and faithful reasoning:** By utilizing highly question-relevant knowledge paths, PoG improves the interpretability of LLM reasoning, enhancing the faithfulness and question-relatedness of generated content.
- **Efficient pruning with graph structure integration:** PoG incorporates efficient pruning techniques in both the KG and reasoning paths to reduce computational costs, mitigate LLM hallucinations caused by irrelevant noise.
- **Flexibility and effectiveness:**
  a) **PoG is a plug-and-play framework** that can be seamlessly applied to various LLMs and KGs.
  b) **PoG allows frequent knowledge updates** via the KG, avoiding the expensive and slow updates for LLMs.
  c) **PoG reduces the LLMs token usage by over 50% with only a ±2% difference in accuracy.**
  d) **PoG achieves SOTA results on all the tested KGQA datasets,** outperforming the strong baseline ToG by an average of 18.9% accuracy using both GPT-3.5 and GPT-4. **Notably, PoG with GPT-3.5 can outperform ToG with GPT-4 by up to 23.9%**

Method

An example workflow of PoG:

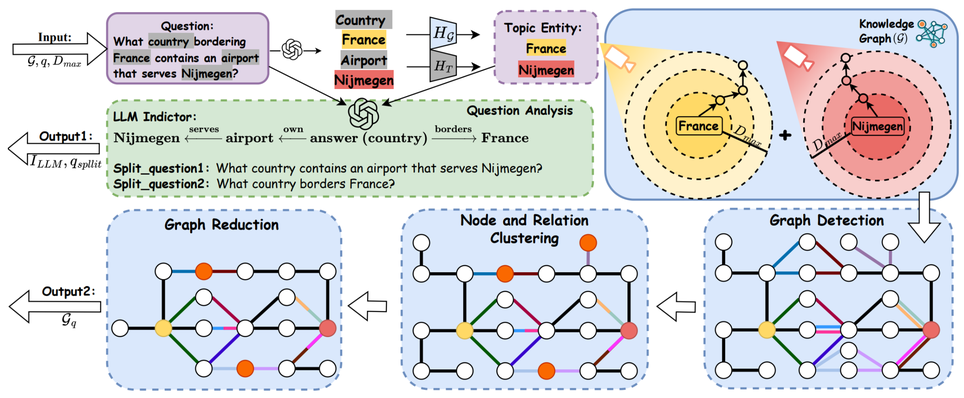


❖ Initialization

❑ Build multi-keywords KG subgraph for **facts evidence implements (LLM Lack Knowledge)**

❑ Skyline LLM indictor for **long reasoning guiding and length prediction (Multi-hops Problem)**

❑ Graph reduction for **early step noise removing (Graph structure utilization)**



❖ Exploration

❑ **Topic Entity Path Exploration**:

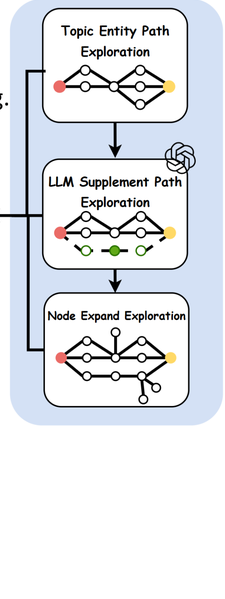
- **Aim: provide multi-hops and multi-entities faithful and interpretable facts** for reasoning.
- **Dynamic exploration**: begins exploration from a predicted depth $D_{Predict}$
- **Topic entity path :** All the paths contained all the topic entity and meet the length's limitation considered.

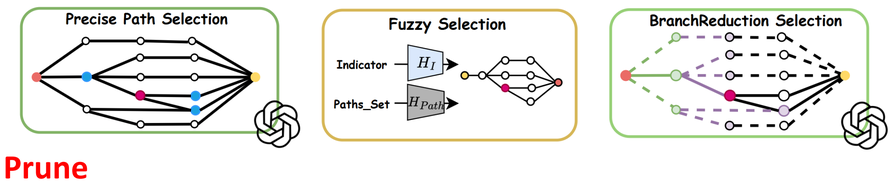
❑ **LLM Supplement Path Exploration:**

- **Aim:** Leverage the **inherent knowledge of the LLM** to generate predictive insights
- **generate one predict insight**, then employ a **verification process** using KGs to **evaluate its faithfulness**

❑ **Node Expand Exploration:**

- **Aim:** Utilizing the **neighborhood information** around the path





❖Path Prune

❑ **Considering the LLM costs and algorithm complexity for different user used purposes.**

❑ **Precise Path Selection:** → For **most accurate** selection

❑ **Fuzzy Selection:** → For **minimal cost.**

❑ **BranchReduction Selection:** → For the **consider of both cost and accuracy**. **Incorporate the graph structure**.

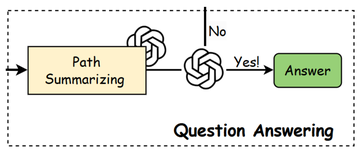
❖Question Answering

❑ **Path Summarizing:**

- Creating a **concise and focused** path.
- To **decrease hallucinations caused by paths with excessive or incorrect text.**

❑ **Question answering:** encouraging deep reasoning

- **Deep reasoning:** prompt the LLM to answer individual split questions and then the overall question.
- **Slow thinking:** use Chain-of-Thought to promote thorough thinking.



Experiments & Results

❖ Experimental Result on Knowledge-Intensive Datasets

Method	Class	LLM	Multi-Hop KGQA			Single-Hop KGQA	Open-Domain QA
			CWQ	WebQSP	GrailQA	Simple Questions	WebQuestions
Without external knowledge							
IO prompt[37]	-	GPT-3.5-Turbo	37.6	63.3	29.4	20.0	48.7
CoT[37]	-	GPT-3.5-Turbo	38.8	62.2	28.1	20.3	48.5
SC[37]	-	GPT-3.5-Turbo	45.4	61.1	29.6	18.9	50.3
With external knowledge							
Prior FT SOTA	SL	-	70.4[9]	85.7[27]	75.4[11]	85.8[1]	56.3[18]
KB-BINDER[24]	ICL	Codex	-	74.4	58.5	-	-
ToG/ToG-R[37]	ICL	GPT-3.5-Turbo	58.9	76.2	68.7	53.6	54.5
ToG-2.0[28]	ICL	GPT-3.5-Turbo	-	81.1	-	-	-
ToG/ToG-R[37]	ICL	GPT-4	69.5	82.6	81.4	66.7	57.9
PoG-E	ICL	GPT-3.5-Turbo	71.9	90.9	87.6	78.3	76.9
PoG	ICL	GPT-3.5-Turbo	74.7	93.9	91.6	80.8	81.8
PoG-E	ICL	GPT-4	78.5	95.4	91.4	81.2	82.0
PoG	ICL	GPT-4	**81.4**	**96.7**	**94.4**	**84.0**	**84.6**

❑ PoG achieves **SOTA results on all the tested KGQA datasets.**

❑ Outperforming ToG by an **average of 18.9%** accuracy using both GPT-3.5 and GPT-4.

❑ Notably, **PoG with GPT-3.5 can outperform ToG with GPT-4 by up to 23.9%.**

Full Paper

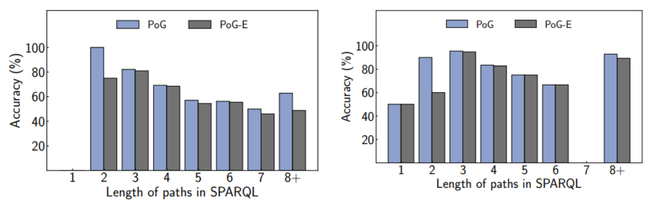
❖ Effective evaluation on multi-entity questions

Question Set	CWQ	WebQSP	GrailQA	WebQuestions	Simple Questions
PoG with GPT-3.5-Turbo					
Single-entity	70.3	93.9	92.1	81.7	78.3
Multi-entity	80.2	93.1	70.7	82.8	-
PoG-E with GPT-3.5-Turbo					
Single-entity	67.5	91	88.2	76.8	80.8
Multi-entity	77.5	82.8	76.0	82.8	-

❑ **PoG maintains the exllecent result with multi entities question.**

❖ Effective evaluation on multi-hops problems

❑ **result shows PoG still maintained the exllecent result although the multi Hops prolems is much more complex.**



The accuracy of PoG and PoG-E, categorized by the different lengths of the ground-truth answers for each question.

❖ Effective evaluation on graph structure utilization

❑ **evaluation on graph structure pruning**

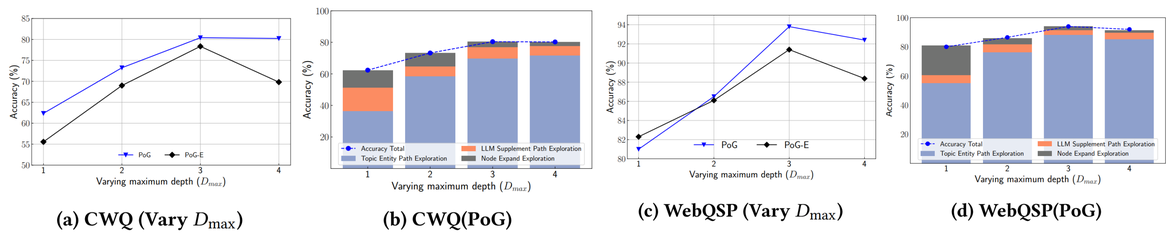
	CWQ	WebQSP	GrailQA	WebQuestions
Ave Entity Number	3,540,267	243,826	62,524	240,863
Ave Entity Number After Pruned	1,621,055	182,673	30,267	177,822
Ave Entitiy Reduction Proportion (%)	**54%**	**25%**	**52%**	**26%**

❑ **Compare the effect of different beam searches**

PoG	Evaluation	CWQ	WebQSP
w/ Fuzzy Selection	Accuracy	57.1	86.4
	Token Input	-	-
	LLM Calls	6.8	6.5
w/ Fuzzy and BranchReduced Selection	Accuracy	79.3	93.0
	Token Input	101,455	328,742
	LLM Calls	9.7	9.3
w/ Fuzzy and Precise Path Selection	Accuracy	**81.4**	**93.9**
	Token Input	216,884	617,448
	LLM Calls	9.1	7.5
w/ 3-Steps Beam Search	Accuracy	79.8	91.9
	Token Input	102,036	369,175
	LLM Calls	8.8	9.0

Effective of graph struct in path prune

❖ Ablation Study: Does search depth matter?



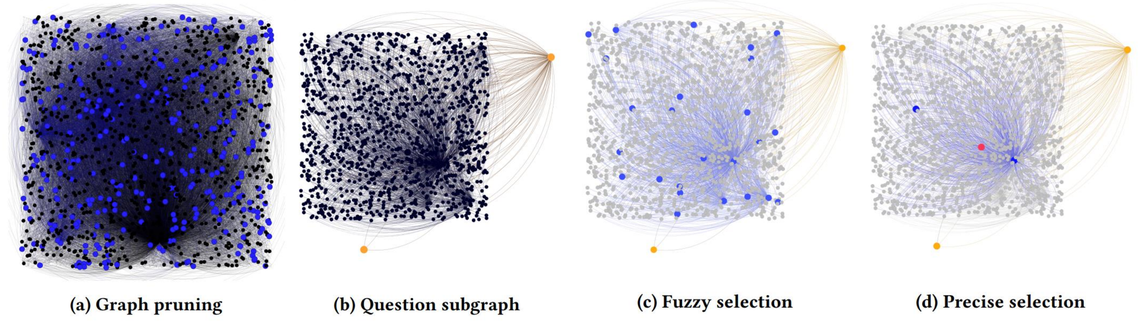
(a) CWQ (Vary $D_{\max}$) (b) CWQ(PoG) (c) WebQSP (Vary $D_{\max}$) (d) WebQSP(PoG)

❑ **PoG performance improves with increased depth, but the benefits diminish beyond a depth-3.**

❑ **The higher depths reduce the effectiveness of both LLM-based path supplementation and node exploration.**

❖ Case study: graph reduction and path pruning

We conducted a case study using the example question presented to illustrate the effects of graph pruning and path pruning on the graph structure.



(a) Graph pruning (b) Question subgraph (c) Fuzzy selection (d) Precise selection

❑ **In these figures, vertices in blue represent the selected entity after each pruning, vertices in yellow represent the topic entities, and the vertex in red denotes the final answer entity.**

❑ From these graphs, we observe that **utilizing the graph structure allows for the rapid pruning of irrelevant vertices, ensuring that the reasoning paths remain faithful and highly relevant to the question,** thereby maintaining the integrity and relevance of the reasoning process.