Evaluating Knowledge Graph Based Retrieval Augmented Generation Methods under Knowledge Incompleteness

Dongzhuoran Zhou[1,2†], Yuqicheng Zhu[2,3], Xiaxia Wang[4], Hongkuan Zhou[2,3], Yuan He[4], Jiaoyan Chen[5], Steffen Staab[3,7], Evgeny Kharlamov[1,2]
[1]University of Oslo, [2]Bosch Center for AI, [3]University of Stuttgart,
[4]University of Oxford, [5]The University of Manchester, [7]University of Southampton
dongzhuoran.zhou@de.bosch.com

Abstract

Knowledge Graph based Retrieval-Augmented Generation (KG-RAG) is a technique that enhances Large Language Model (LLM) inference in tasks like Question Answering (QA) by retrieving relevant information from knowledge graphs (KGs). However, KGs are often incomplete, meaning that essential information for answering questions may be missing. Existing benchmarks do not adequately capture the impact of KG incompleteness on KG-RAG performance. In this paper, we systematically evaluate KG-RAG methods under incomplete KGs by removing triples using different methods and analyzing the resulting effects. We demonstrate that KG-RAG methods are sensitive to KG incompleteness, highlighting the need for more robust approaches in realistic settings.

1 Introduction

Large Language Models (LLMs) have achieved remarkable success across various natural language processing tasks such as Question Answering (QA) [Achiam et al., 2023, Touvron et al., 2023, Jiang et al., 2023]. However, they face several limitations, including outdated knowledge [Dhingra et al., 2022], insufficient domain-specific expertise [Li et al., 2023], and a tendency to generate plausible but incorrect information, known as hallucination [Ji et al., 2023].

Retrieval-Augmented Generation (RAG) mitigates these issues by integrating information retrieval mechanisms, allowing LLMs to access up-to-date and reliable information without requiring modifications to their architecture or parameters [Khandelwal et al., 2019, Ram et al., 2023, Borgeaud et al., 2022, Izacard and Grave, 2021]. While conventional RAG approaches rely on unstructured text corpora, recent studies [Chen et al., 2023, Han et al., 2024, Zhang et al., 2025] have started to explore knowledge graphs (KGs) as knowledge resources by exploring different retrieval methods, such as neighborhood extraction after entity linking and similarity-based entity search in the embedding space, as well as different prompting methods like knowledge re-writting [Wu et al., 2024]. Such KG-based RAG (KG-RAG for shortness) methods can reduce text redundancy [Li et al., 2024], enable flexible updates [Paulheim, 2016], and provide structured reasoning evidence [Luo et al., 2024]. Despite these benefits, KGs are often incomplete in real-world applications [Min et al., 2013, Ren et al., 2020, Pflueger et al., 2022], raising a crucial question: *Can KG-RAG methods remain effective when the given KG is incomplete*?

LLMs have demonstrated substantial knowledge retention and reasoning capabilities [Achiam et al., 2023, Guo et al., 2025]. Consequently, KG-RAG methods [Luo et al., 2024, Sun et al., 2024, He et al., 2024] are expected to leverage LLMs to infer relevant facts that extend beyond explicitly retrieved triples. For instance, from the retrieved triples ⟨*JustinBieber, has_parent, JeremyBieber*⟩ and ⟨*JeremyBieber, has_child, JaxonBieber*⟩, LLMs can infer that Jaxon Bieber is Justin Bieber's brother by recognizing the underlying semantic path. In this way, KG-RAG methods can effectively compensate for missing direct evidence and mitigate the impact of incomplete KGs.

However, existing KG-RAG studies directly adopt KGQA benchmarks [Yih et al., 2016, Talmor and Berant, 2018] for evaluation. In these benchmarks, the ground truth answer of each question can be

*Corresponding author. †Equal contribution.

IRIS-AI 2025: *International Research and Industry Symposium on Artificial Intelligence*, Montefiascone, Italy.

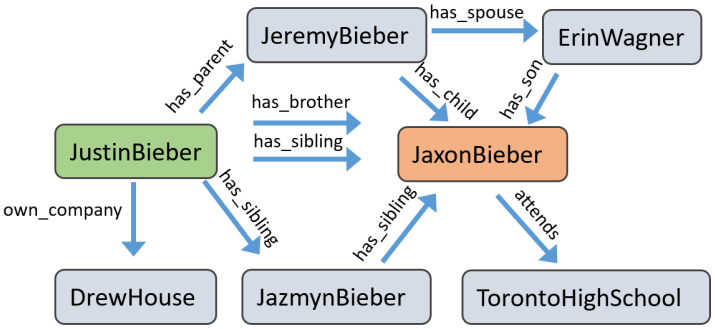

Figure 1: **Illustration of the running example.** The question "*Who is the brother of Justin Bieber?*" has the expected answer *JaxonBieber*. While the direct evidence path *JustinBieber* $\xrightarrow{has\_brother}$ *JaxonBieber* provides a straightforward answer, it is often missing in practice. A robust KG-RAG method should leverage alternative reasoning paths, such as *JustinBieber* $\xrightarrow{has\_parent}$ *JeremyBieber* $\xrightarrow{has\_child}$ *JaxonBieber*, to derive the correct answer.

directly inferred from the knowledge in the KG, failing to reflect real-world scenarios where KGs are incomplete for answering the question. Thus the current evaluation does not reflect KG-RAG methods' performance facing incomplete knowledge. In this paper, we systematically assess three popular KG-RAG methods under varying levels of KG incompleteness. Our findings reveal that **these methods still benefit significantly from incomplete KGs, but are sensitive to missing knowledge with performance drops**, emphasizing the need for more robust approaches for KG-RAG.

2 Methodology

2.1 Background and Notations

Knowledge Graph. Given two finite sets E and R, whose elements are called *entities* and *relations*, a KG G is a directed multi-relational graph represented as a subset of $E \times R \times E$, where each element $\langle h, r, t \rangle \in G$ is called a *triple*.

Reasoning Path. A reasoning path in a KG is a directed sequence of entities and relations that connect a source entity to a target entity, forming a logical pathway for inference:

$$e_0 \xrightarrow{r_1} e_1 \xrightarrow{r_2} \dots \xrightarrow{r_l} e_l,$$

where $e_i \in E$, $r_i \in R$, and l denotes the length of the reasoning path. For example, given the question "*Who is the brother of Justin Bieber?*", a possible reasoning path to the answer, as shown in Figure 1, is:

$$\textit{JustinBieber} \xrightarrow{has\_parent} \textit{JeremyBieber} \xrightarrow{has\_child} \textit{JaxonBieber}.$$

2.2 Benchmark Datasets

We build our experimental datasets based on two KGQA benchmarks that are mostly used by current KG-RAG methods: WebQuestionsSP (WebQSP) [Yih et al., 2016] and Complex WebQuestions (CWQ) [Talmor and Berant, 2018]. These datasets consist of natural language questions designed to be answered using a KG. The questions are open-domain and require retrieval of structured information from Freebase [Bollacker et al., 2008], a large-scale KG with 88 million entities and 126 million triples. Each question in these datasets is annotated with a *topic entity* and an *answer entity*.

- **Topic Entity**: This is the primary entity mentioned in the question, serving as the starting point for retrieving triples from the KG. In the running example, *"JustinBieber"* is the topic entity.
- **Answer Entity**: This is the specific entity within the KG that answers the given question. In our running example, the answer entity would be *"JaxonBieber"*, representing Justin Bieber's brother. Note that we might have multiple answer entities for the same question.

2.3 KG-RAG Methods

RoG [Luo et al., 2024] adopts a planning–retrieval–reasoning pipeline. The LLM first generates relation paths as faithful reasoning plans, which are then used to retrieve valid reasoning paths from the KG for answer generation.

Experimental Setting	WebQSP		CWQ	
	Accuracy (%)	Hits (%)	Accuracy (%)	Hits (%)
RoG - w/o deletion	76.75	86.60	57.49	61.79
RoG - 5% random deletion	75.55 (-1.56%)	85.99 (-0.71%)	57.33 (-0.28%)	61.54 (-0.40%)
RoG - 10% random deletion	74.66 (-2.72%)	85.68 (-1.06%)	55.88 (-2.80%)	60.43 (-2.20%)
RoG - 20% random deletion	72.15 (-5.99%)	84.15 (-2.83%)	53.23 (-7.41%)	58.12 (-5.94%)
RoG - reasoning path disruption	65.43 (-14.70%)	78.12 (-9.80%)	52.95 (-7.90%)	57.4 (-7.10%)
RoG - w/o retrieval	50.46 (-34.25%)	65.05 (-24.90%)	35.40 (-38.42%)	40.10 (-35.11%)
ToG - w/o deletion	44.19	67.40	48.94	51.80
ToG - 5% random deletion	43.74 (-1.02%)	67.17 (-0.33%)	48.78 (-0.32%)	51.34 (-0.89%)
ToG - 10% random deletion	43.09 (-1.64%)	66.81 (-0.88%)	47.94 (-2.05%)	50.44 (-2.63%)
ToG - 20% random deletion	41.61 (-3.42%)	66.18 (-1.81%)	46.14 (-5.73%)	49.18 (-5.05%)
ToG - reasoning path disruption	38.25 (-13.44%)	62.70 (-6.97%)	44.56 (-8.95%)	47.31 (-8.67%)
ToG - w/o retrieval	34.63 (-21.60%)	57.24 (-15.08%)	34.39 (-29.73%)	37.40 (-27.80%)
G-Retriever - w/o deletion	53.43	71.49	35.39	41.51
G-Retriever - 5% random deletion	52.33 (-2.06%)	70.91 (-0.81%)	34.63 (-2.15%)	40.55 (-2.31%)
G-Retriever - 10% random deletion	51.87 (-2.92%)	70.76 (-1.02%)	33.98 (-3.98%)	40.27 (-2.99%)
G-Retriever - 20% random deletion	50.88 (-4.78%)	69.70 (-2.51%)	33.78 (-4.55%)	39.98 (-3.69%)
G-Retriever - reasoning path disruption	49.36 (-7.61%)	69.41 (-2.91%)	33.56 (-5.17%)	39.73 (-4.29%)
G-Retriever - w/o retrieval	29.72 (-44.37%)	47.17 (-34.01%)	14.86 (-58.01%)	17.55 (-57.72%)

Table 1: Performance of KG-RAG methods under different levels of KG incompleteness. The numbers in parentheses indicate the relative performance drop compared to the no-deletion setting for each method.

ToG [Sun et al., 2024] treats the LLM as a graph reasoning agent. It performs iterative beam search over the KG, dynamically constructing top-ranked reasoning paths via relation/entity exploration and pruning.

G-Retriever [He et al., 2024] introduces a retrieval-augmented framework that extracts a compact subgraph via PCST over candidate nodes and encodes it as a soft prompt to guide LLM-based answer generation.

2.4 Deletion Strategies

To evaluate the robustness of three popular KG-RAG methods (ToG , RoG and G-Retriever) under KG incompleteness, we simulate missing knowledge by removing triples from the KG of WebQSP and CWQ. For ToG we choose LLama2-70B-Chat as backbone model. We employ two different deletion strategies: **random triple deletion** and **reasoning path disruption**, each designed to model different levels of KG incompleteness.

Random Triple Deletion. We randomly remove a specified percentage of triples from the KG. This approach provides a baseline for understanding the general impact of missing knowledge. By varying the deletion percentage, we analyze how KG-based RAG performance degrades as more information is lost.

Reasoning Path Disruption. To simulate more severe knowledge incompleteness, this strategy selectively disrupts reasoning paths that support LLMs' inference. For each question, we identify the shortest reasoning paths between the topic and answer entities using the breadth-first search algorithm [Moore, 1959]. We then randomly select **one** such path and remove a randomly chosen triple from it. This breaks a critical step in the reasoning chain, emulating real-world scenarios where essential intermediate knowledge may be missing.

2.5 Evaluation Metrics

The performance of KG-RAG is evaluated by *Accuracy* and *Hits*. Let Q denote the set of test questions. For each question $q \in Q$, the KG-RAG model generates a sequence of tokens as output, denoted by $S(q)$. Let $A(q) = \{a_1, a_2, \dots\}$ denote the corresponding set of ground truth token sequences for question q. Then, the evaluation metrics are defined as follows:

$$Accuracy := \frac{1}{|Q|} \sum_{q \in Q} \sum_{a \in A(q)} \frac{\mathbb{1}[a \preceq_{\text{subseq}} S(q)]}{|A(q)|};$$

$$Hits := \frac{1}{|Q|} \sum_{q \in Q} \max_{a \in A(q)} \mathbb{1}[a \preceq_{\text{subseq}} S(q)],$$

where $\mathbb{1}[\cdot]$ is the indicator function and $\preceq_{\text{subseq}}$ denotes the contiguous subsequence relation (i.e. $a \preceq_{\text{subseq}} S(q)$ means that the sequence a appears in order within $S(q)$ without interruption).

3 Experimental Results

Table 1 presents the performance of three KG-RAG methods under varying levels of KG incompleteness. We evaluate models in multiple settings: a fully intact KG (no deletion), random triple deletion at 5%, 10%, and 20% levels, reasoning path disruption (disabling a key step in the reasoning chain), and a baseline condition where retrieval from the KG is entirely disabled. Our findings highlight three key observations.

First, using KG as retrieval source significantly improves performance compared to the LLM without retrieval. For example, RoG's accuracy falls from 76.75% to 50.46% (-34.25%) in WebQSP. However, all evaluated KG-RAG methods exhibit **high sensitivity to missing knowledge in the KG**. Notably, even under random deletion, where test question-specific triples may remain unaffected, accuracy consistently declines as the proportion of deleted triples increases across both datasets. The relative accuracy drop is up to 5.99% in WebQSP and 7.41% in CWQ under 20% random triple deletion.

Second, **disrupting a single reasoning path leads to a substantial performance drop**, despite that multiple reasoning paths often exist for a given question. For instance, in WebQSP, RoG's accuracy drops from 76.75% to 65.43% (-14.70%), and its hits declines from 86.6% to 78.12% (-9.80%). This suggests that current KG-RAG methods often rely on specific paths for reasoning, rather than effectively utilizing alternative reasoning pathways when available. Models required fine-tuning on the KG such as RoG may be particularly dependent on a small set of learned retrieval patterns, making them vulnerable to missing knowledge in those specific paths.

Finally, our results indicate that **KGs remain valuable retrieval sources even when incomplete**. Although performance degrades under KG incompleteness, it still surpasses the baseline where no KG retrieval is used. This suggests that addressing KG incompleteness more effectively—rather than discarding KGs entirely—should be the focus when designing more resilient KG-RAG methods.

4 Case Study

We conduct a case study on the WebQTest-116 example: (*"which country was Justin Bieber born in?"*). The gold reasoning path is:

$$\textit{Justin Bieber} \xrightarrow{\textit{nationality}} \textit{Canada}$$

Although the question explicitly asks for a country, models often struggle when intermediate entities (e.g., cities or provinces) are involved. After removing the triple $\langle \textit{Justin Bieber, nationality, Canada} \rangle$, all evaluated KG-RAG methods fail to recover the correct answer, despite the existence of a reasonable alternative paths:

$$\textit{Justin Bieber} \xrightarrow{\textit{place of birth}} \textit{London} \xrightarrow{\textit{contained by}} \textit{Canada}$$

or

$$\textit{Justin Bieber} \xrightarrow{\textit{place lived}} \textit{Stratford} \xrightarrow{\textit{contained by}} \textit{Canada}$$

A closer analysis reveals that G-RETRIEVER incorrectly outputs a province (Ontario), while RoG retrieves a city (London), both misinterpreting the query's focus on a country.

To verify that this issue is not limited to a single example, we observe similar failures in more cases. In WebQTest-436 (*"What is the Nigeria time?"*), the gold reasoning path is:

$$\textit{Nigeria} \xrightarrow{\textit{time zones}} \textit{West Africa Time Zone}.$$

A valid alternative path exists via an administrative region:

$$\textit{Nigeria} \xrightarrow[\text{division}]{\text{administrative}} \textit{Bauchi} \xrightarrow{\textit{time zones}} \textit{West Africa Time Zone}.$$

Similarly, in WebQTest-1481 (*"What city is the University of Oregon state in?"*), the gold path is:

$$\textit{University of Oregon} \xrightarrow{\textit{contained by}} \textit{Eugene}$$

with a valid alternative route through the campus entity:

$$\textit{University of Oregon} \xrightarrow[\text{campus}]{\text{has}} \underset{\textit{Campus}}{\textit{Eugene}} \xrightarrow{\textit{contained by}} \textit{Eugene}.$$

In both cases, KG-RAG models fail to leverage the alternative paths once the gold triple is removed. This highlights a broader weakness: these models often rely on memorized or shallow retrieval patterns and struggle to generalize via multi-hop, semantically coherent reasoning chains—despite such paths being available in the KG.

5 Conclusion and Future Work

This work evaluated three KG-RAG methods under knowledge incompleteness, demonstrating their sensitivity to missing information. Even random triple deletions reduced performance, while reasoning path disruption led to substantial performance drops. Despite this, KGs remained valuable retrieval sources, consistently outperforming retrieval-free baselines.

Beyond incompleteness, real-world KGs also contain noise, such as incorrect triples generated during automatic KG construction process or errors in the original knowledge source [Heindorf et al., 2016, Paulheim, 2016]. Additionally, the lack of a standardized evaluation protocol for KG-RAG methods and the inherent factual instability of LLMs [Potyka et al., 2024, He et al., 2025] can lead to inconsistent assessments across studies.

Future work should focus on developing more robust KG-RAG methods that can handle both missing and noisy knowledge, ensuring reliable performance in real-world scenarios. This includes uncertainty-aware retrieval, noise-resistant reasoning mechanisms, and hybrid approaches that integrate structured and unstructured sources. Additionally, creating comprehensive benchmarks and standardized evaluation protocols will enable systematic assessment and foster progress toward more resilient KG-RAG systems.

6 Limitations

Our study evaluates the robustness of KG-RAG methods to knowledge incompleteness by systematically removing triples from the source KG. While this approach reveals meaningful failure modes, one limitation is that some removed triples may not be inferable from the remaining knowledge. In such cases, performance degradation is expected and does not necessarily reflect a model's reasoning capability. However, if such non-inferable deletions are common, this raises broader concerns about the adequacy of existing benchmarks for evaluating KG-RAG robustness.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. https://doi.org/10.48550/arXiv.2303.08774.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008. https://doi.org/10.1145/1376616.1376746.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR, 2022. https://doi.org/10.48550/arXiv.2112.04426.

Jiaoyan Chen, Yuan He, Yuxia Geng, Ernesto Jiménez-Ruiz, Hang Dong, and Ian Horrocks. Contextual semantic embeddings for ontology subsumption prediction. *World Wide Web*, 26(5):2569–2591, 2023. https://doi.org/10.1007/s11280-023-01169-9.

Bhuwan Dhingra, Jeremy R Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W Cohen. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273, 2022. https://doi.org/10.1162/tacl_a_00459.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. https://doi.org/10.48550/arXiv.2501.12948.

Haoyu Han, Yu Wang, Harry Shomer, Kai Guo, Jiayuan Ding, Yongjia Lei, Mahantesh Halappanavar, Ryan A Rossi, Subhabrata Mukherjee, Xianfeng Tang, et al. Retrieval-augmented generation with graphs (graphrag). *arXiv preprint arXiv:2501.00309*, 2024. https://doi.org/10.48550/arXiv.2501.00309.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *arXiv preprint arXiv:2402.07630*, 2024. https://doi.org/10.48550/arXiv.2402.07630.

Yuan He, Bailan He, Zifeng Ding, Alisia Lupidi, Yuqicheng Zhu, Shuo Chen, Caiqi Zhang, Jiaoyan Chen, Yunpu Ma, Volker Tresp, and Ian Horrocks. Supposedly equivalent facts that aren't? entity frequency in pre-training induces asymmetry in llms. *arXiv preprint arXiv:2503.22362*, 2025. https://doi.org/10.48550/arXiv.2503.22362.

Stefan Heindorf, Martin Potthast, Benno Stein, and Gregor Engels. Vandalism detection in wikidata. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 327–336, 2016. https://doi.org/10.1145/2983323.2983740.

Gautier Izacard and Édouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, 2021. https://doi.org/10.18653/v1/2021.eacl-main.74.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023. https://doi.org/10.1145/3571730.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. https://doi.org/10.48550/arXiv.2310.06825.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*, 2019. https://doi.org/10.48550/arXiv.1911.00172.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. https://doi.org/10.48550/arXiv.1312.6114.

Mufei Li, Siqi Miao, and Pan Li. Simple is effective: The roles of graphs and large language models in knowledge-graph-based retrieval-augmented generation. *arXiv preprint arXiv:2410.20724*, 2024. https://doi.org/10.48550/arXiv.2410.20724.

Xianzhi Li, Samuel Chan, Xiaodan Zhu, Yulong Pei, Zhiqiang Ma, Xiaomo Liu, and Sameena Shah. Are chatgpt and gpt-4 general-purpose solvers for financial text analytics? a study on several typical tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 408–422, 2023. https://doi.org/10.18653/v1/2023.emnlp-industry.39.

Linhao Luo, Yuan-Fang Li, Reza Haf, and Shirui Pan. Reasoning on graphs: Faithful and interpretable large language model reasoning. In *The Twelfth International Conference on Learning Representations*, 2024. https://doi.org/10.48550/arXiv.2310.01061.

Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. Distant supervision for relation extraction with an incomplete knowledge base. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 777–782, 2013.

E. F Moore. The shortest path through a maze. In *Proceedings of the International Symposium on the Theory of Switching*, pages 285–292, 1959.

Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2016. https://doi.org/10.3233/SW-160218.

Maximilian Pflueger, David J Tena Cucala, and Egor V Kostylev. Gnnq: A neuro-symbolic approach to query answering over incomplete knowledge graphs. In *International Semantic Web Conference*, pages 481–497. Springer, 2022. https://doi.org/10.1007/978-3-031-19433-7_28.

Nico Potyka, Yuqicheng Zhu, Yunjie He, Evgeny Kharlamov, and Steffen Staab. Robust knowledge extraction from large language models using social choice theory. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pages 1593–1601, 2024. https://doi.org/10.5555/3635637.3663020.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331, 2023. https://doi.org/10.1162/tacl_a_00605.

Hongyu Ren, Weihua Hu, and Jure Leskovec. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *International Conference on Learning Representations*, 2020. https://doi.org/10.48550/arXiv.2002.05969.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel Ni, Heung-Yeung Shum, and Jian Guo. Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph. In *The Twelfth International Conference on Learning Representations*, 2024. https://doi.org/10.48550/arXiv.2307.07697.

Alon Talmor and Jonathan Berant. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*, 2018. https://doi.org/10.18653/v1/N18-1059.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. https://doi.org/10.48550/arXiv.2302.13971.

Yike Wu, Yi Huang, Nan Hu, Yuncheng Hua, Guilin Qi, Jiaoyan Chen, and Jeff Pan. Cotkr: Chain-of-thought enhanced knowledge rewriting for complex knowledge graph question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3501–3520, 2024. https://doi.org/10.18653/v1/2024.emnlp-main.205.

Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–206, 2016. https://doi.org/10.18653/v1/P16-2033.

Qinggang Zhang, Shengyuan Chen, Yuanchen Bei, Zheng Yuan, Huachi Zhou, Zijin Hong, Junnan Dong, Hao Chen, Yi Chang, and Xiao Huang. A survey of graph retrieval-augmented generation for customized large language models. *arXiv preprint arXiv:2501.13958*, 2025. https://doi.org/10.48550/arXiv.2501.13958.

A KG-RAG Method Details

RoG (Reasoning on Graphs). RoG implements a faithful reasoning pipeline by decomposing KGQA into three stages: (1) *planning*: prompting the LLM to generate relation paths grounded in the KG (e.g., `marry_to → father_of`); (2) *retrieval*: matching these relation paths with actual reasoning paths in the KG (e.g., `Alice → Bob → Charlie`); (3) *reasoning*: using the retrieved paths as evidence to answer the question.

Formally, RoG models the answer probability as:

$$P_\theta(a \mid q, G) = \sum_{z \in \mathcal{Z}} P_\theta(a \mid q, z, G) \cdot P_\theta(z \mid q) \tag{1}$$

where z denotes a relation path (plan), and θ are the LLM parameters.

Training is guided by the ELBO [Kingma and Welling, 2014]:

$$\log P(a \mid q, G) \geq \mathbb{E}_{z \sim Q(z)} \big[\log P_\theta(a \mid q, z, G) \big] - \mathrm{KL}\left(Q(z) \,\|\, P_\theta(z \mid q)\right) \tag{2}$$

Note that this ELBO formulation serves as a theoretical framework; in practice, RoG adopts supervised instruction tuning rather than explicitly optimizing this bound.

ToG (Think-on-Graph). ToG follows a tight integration paradigm (denoted as "LLM⊗KG" in the original paper), where LLMs interact directly with the KG during reasoning via iterative exploration and pruning. It consists of three phases:

- **Initialization**: The LLM identifies top-N topic entities from the question.
- **Exploration**: At each iteration, the LLM conducts relation and entity exploration from current frontiers and prunes candidates to retain top-N reasoning paths.
- **Reasoning**: The LLM evaluates whether the current paths suffice to answer the question. If not, exploration continues until a satisfactory path is found or a max depth is reached.

G-Retriever. G-Retriever is a retrieval-augmented generation framework designed for question answering over graphs. It is applicable to both structured symbolic graphs (e.g., Freebase) and text-attributed graphs (e.g., scene graphs, commonsense graphs).

The method integrates subgraph retrieval with prompt-based conditioning as follows:

- **Subgraph Retrieval via PCST**: Given a question and topic entity, G-Retriever first selects candidate nodes by exploring the local KG neighborhood (e.g., 1–2 hops). These nodes are treated as rewards in a Prize-Collecting Steiner Tree (PCST) formulation, which extracts a compact subgraph containing informative multi-hop paths.
- **Graph Prompt Construction**: The retrieved subgraph is linearized into token sequences using fixed templates (e.g., `[head] -[relation]-> [tail]`), converting structural triples into flattened sequences.
- **Soft Prompt Encoding**: Rather than appending text directly, the serialized subgraph is embedded as a trainable soft prompt. These embeddings are prepended to the LLM input, conditioning generation in a parameter-efficient manner.

At inference time, the LLM processes the soft prompt and question jointly to generate the answer. This design enables faithful reasoning over retrieved evidence while adapting to different graph modalities through prompt tuning.