Debate on Graph: A Flexible and Reliable Reasoning Framework for Large Language Models

Jie Ma[1*], Zhitao Gao[1], Qi Chai[2], Wangchun Sun[1], Pinghui Wang[1*], Hongbin Pei[1], Jing Tao[1], Lingyun Song[3], Jun Liu[1], Chen Zhang[4], Lizhen Cui[5]

[1] MOE KLINNS Lab, Xi'an Jiaotong University
[2] The Hong Kong University of Science and Technology (Guangzhou)
[3] Northwestern Polytechnical University
[4] Zhejiang Createlink Technology
[5] Shandong University
{jiema, phwang}@xjtu.edu.cn

Abstract

Large Language Models (LLMs) may suffer from hallucinations in real-world applications due to the lack of relevant knowledge. In contrast, knowledge graphs encompass extensive, multi-relational structures that store a vast array of symbolic facts. Consequently, integrating LLMs with knowledge graphs has been extensively explored, with Knowledge Graph Question Answering (KGQA) serving as a critical touchstone for the integration. This task requires LLMs to answer natural language questions by retrieving relevant triples from knowledge graphs. However, existing methods face two significant challenges: *excessively long reasoning paths distracting from the answer generation*, and *false-positive relations hindering the path refinement*. In this paper, we propose an iterative interactive KGQA framework that leverages the interactive learning capabilities of LLMs to perform reasoning and Debating over Graphs (DoG). Specifically, DoG employs a subgraph-focusing mechanism, allowing LLMs to perform answer trying after each reasoning step, thereby mitigating the impact of lengthy reasoning paths. On the other hand, DoG utilizes a multi-role debate team to gradually simplify complex questions, reducing the influence of false-positive relations. This debate mechanism ensures the reliability of the reasoning process. Experimental results on five public datasets demonstrate the effectiveness and superiority of our architecture. Notably, DoG outperforms the state-of-the-art method ToG by 23.7% and 9.1% in accuracy on WebQuestions and GrailQA, respectively. Furthermore, the integration experiments with various LLMs on the mentioned datasets highlight the flexibility of DoG.

Code — https://github.com/reml-group/DoG

Introduction

Large Language Models (LLMs), characterized by their substantial parameter amount (Zhang et al. 2023) and training on extensive, diverse, and unlabeled data (Rawte, Sheth, and Das 2023), exhibit remarkable proficiency in a wide range of natural language understanding and generation tasks (Lin et al. 2023; Liu et al. 2024). For example, GPT-4 (Achiam et al. 2023) demonstrates human-level performance across a majority of professional and academic exams originally intended for humans. However, recent studies (Guan et al. 2024; Waldendorf, Haddow, and Birch 2024; Gunjal, Yin, and Bas 2024; Ma et al. 2023) have revealed that they may suffer from hallucinations in real-world applications due to a deficiency in relevant knowledge.

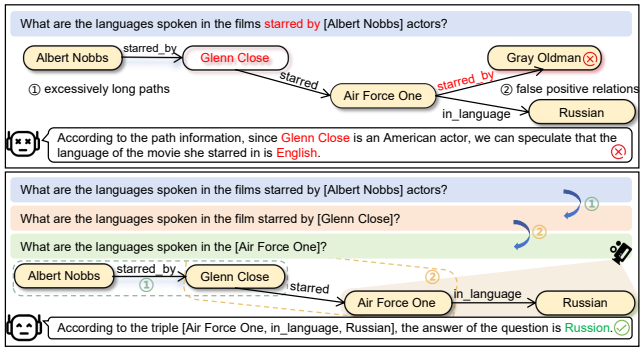

Figure 1: Illustration of challenges and our solutions.

Knowledge graphs (Wang et al. 2024) are large-scale, multi-relational structures housing a plethora of symbolic facts, such as the triple `<The Eiffel Tower, locatedIn, Paris>`. The incorporation of these structured facts may tackle the aforementioned issue of hallucinations in LLMs (Guan et al. 2024; Quintero-Narvaez and Monroy 2024; Shi et al. 2023). One approach to evaluating the integration of knowledge graphs with LLMs is through Knowledge Graph Question Answering (KGQA) (Ji et al. 2022), which requires machines to answer natural language questions by retrieving relevant facts from knowledge graphs. Recent works (Li et al. 2024; Toroghi et al. 2024; Nie et al. 2024) primarily follow an iterative inference paradigm, consisting of two steps: (1) identifying the initial entity in the question, and (2) retrieving and refining the inference path iteratively until reaching the answer or obtaining sufficient evidence to answer the question. Although they have achieved significant success, they still suffer from *excessively long paths* and *false-positive relations*.

Challenge 1: excessively long paths distracting from

*Corresponding author.

the answer generation. Existing methods (Ye et al. 2022; Guo et al. 2023a; Kim et al. 2023) usually feed a lengthy evidence path like {<Albert Nobbs, starred_by, Glenn Close>, ···, <Air Force One, starred_by, Gray Oldman>, ···} at the top of Fig. 1 into LLMs to perform answer generation in a single step, which may make it challenging for LLMs to discern the key points in the path. For instance, LLMs may focus on the tail entity Glenn Close and employ their internal prior knowledge to generate answers. This will result in answers that appear reasonable but are incorrect.

Challenge 2: false-positive relations hindering the path refinement. Current methods (Bai et al. 2023; Hu et al. 2024; Li et al. 2024) typically focus on identifying relations within graphs that closely match or have the same meaning as those in the questions, even if the relations have already been identified in previous reasoning steps. For example, at the top of Fig. 1, these methods may select starred_by, which was used in the previous reasoning step and is mentioned in the question, to expand paths rather than choosing in_language when dealing with the entity Air Force One. This will lead to incomplete evidence paths.

To address these challenges, we propose an iterative interactive KGQA framework that leverages the interactive learning capabilities of LLMs to perform reasoning and **D**ebating **o**ver **G**raphs, dubbed **DoG**. Unlike existing approaches (Jiang et al. 2023a; Luo et al. 2023; Sun et al. 2024) that seek to construct a complete evidence chain before answering questions, our architecture employs a subgraph-focusing mechanism that allows LLMs to perform answer trying after each reasoning step. For each filtered triple, DoG uses LLMs to assess whether sufficient information is available to answer the current question. In this way, the triple in each reasoning step, such as <Glenn Close, starred, AirForce One> in the bottom of Fig. 1, can be deeply pondered by LLMs. If the triple does not support answering the current question, DoG employs a multi-role LLM team to debate and simplify the question based on the triple. The iterative process allows complex multi-hop questions to be gradually transformed into single-hop questions, which enables LLMs not to be disturbed by the relation that is retrieved in the previous reasoning step. For example, the relation starred_by that is linked with Air Force One will not disturb reasoning after the simplification procedure ②. This is inspired by the human brain in tackling complex questions, which guides LLMs to reason on graphs through chain-of-thought (Wei et al. 2022). The simplification process can also enhance the transparency of the reasoning process.

To verify the effectiveness and superiority of our architecture, we conduct thorough experiments on five public KGQA datasets: MetaQA (Zhang et al. 2018), WebQSP (Yih et al. 2016), CWQ (Talmor and Berant 2018), WebQuestions (Berant et al. 2013), and GrailQA (Gu et al. 2021). Our findings show that DoG achieves state-of-the-art results on all datasets, except for the 2-hop and 3-hop questions within MetaQA. Notably, DoG outperforms the strong baseline ToG (Sun et al. 2024) by 23.7% and 9.1% in accuracy on WebQuestions and GrailQA, respectively. In summary, our contributions are threefold.

- We propose a flexible and reliable reasoning framework, DoG, which enables LLMs to reason and debate over knowledge graphs and answer questions after thorough deliberation.
- We introduce a strategy, which transforms questions from complex to easy through the interactive learning of a multi-role LLM team, for handling complex reasoning on knowledge graphs. This guides LLMs to engage in step-by-step reasoning, thereby enhancing the reliability of the reasoning process.
- Extensive experiments and ablation studies are carried out on five public datasets to demonstrate the effectiveness and superiority of our architecture. Furthermore, we also conduct integration experiments with various LLMs to verify the flexibility of DoG.

Related Work

The methods of LLM reasoning over knowledge graphs can be classified into *batch triple recalling*, and *reasoning path refining* from the perspective of evidence gathering.

Batch triple recalling. Knowledge graphs typically store an extensive amount of facts (Cui et al. 2023). For instance, Freebase (Bollacker et al. 2008) contains over 1.9 billion triples, and even the smaller non-open-domain MetaQA (Zhang et al. 2018) includes over 130,000 triples. The number of relevant triples can be substantial even when constrained by the entities in a given question. Injecting all these triples into the context window of LLMs to perform reasoning not only incurs a high encoding cost but also introduces significant noise (Wei et al. 2023). To address this issue, previous studies (Shu et al. 2022; Ye et al. 2022; Guo et al. 2023a) focus on how to filter suitable facts. For instance, KAPING (Baek, Aji, and Saffari 2023) projects questions and triples into the same space to obtain relevant knowledge by semantic similarity. KG-GPT (Kim et al. 2023) further focuses on fine-grained question representations, decomposing multi-hop questions into sub-questions and matching the relations associated with entities in those sub-questions, then selecting the top-k relevant relations to form evidence triples. Similarly, KGR (Guan et al. 2024) splits the retrieved triples into several chunks and utilizes LLM to distinguish the critical triple relevant with questions.

Reasoning path refining. The paradigm of this kind of method (Gu, Deng, and Su 2023; Jiang et al. 2023a; Liu et al. 2023; Luo et al. 2023; Sun et al. 2024; Guo et al. 2023b) is first to identify the initial entity in the question, then to iteratively retrieve and refine the reasoning path until reaching the answer or obtaining sufficient evidence to answer the question, and finally to employ LLMs to generate answers based on the refined path. For example, Jiang et al. (2023a) proposed an iterative reading-reasoning approach, which iterates an invoking-linearization-generation procedure. It utilizes LLMs to perform reasoning on the interface that is specifically designed for reading structured data until deriving the final answer. Similarly, Sun et al. (2024) introduced a deep and responsible reasoning framework, which first conducts a beam search on a graph from the entity within ques-

tions and then acquires multiple reasoning paths as evidence for answer generation. It is noteworthy that these methods all treat the LLM as a tool for accomplishing specific tasks, conceptualizing it as function executors, and relying on in-context learning (Dong et al. 2022) or fine-tuning to refine its outputs (Jiang et al. 2024). However, some studies (Zhao et al. 2024; Zhang, Xu, and Deng 2023; Schumann et al. 2024) have demonstrated that LLMs can be induced to exhibit human personality traits and role distinctions to undertake complex reasoning tasks.

Communicative Agents. The primary objective of agents is to collaboratively address complex tasks in a productive and efficient manner through autonomous communication and negotiation (Chan et al. 2023; Liang et al. 2023; Yang et al. 2023; Kirk et al. 2024). LLMs such as ChatGPT and Vicuna (Chiang et al. 2023) are frequently employed as these communicative agents. Recently, numerous studies have investigated the application of these agents in various domains, including AI societies (Li et al. 2023a), software development (Qian et al. 2023), translation (Liang et al. 2023), arithmetic problem-solving (Du et al. 2023), dialogue response generation (Chan et al. 2023), and strategic planning among robots (Mandi, Jain, and Song 2023). Specifically, Wang et al. (2023) guided ChatGPT to emulate expert system reviewers, thereby improving the quality of its literature retrieval queries. Kong et al. (2023) introduced a strategically designed role-playing prompt method to enhance reasoning abilities by assigning appropriate expert roles for tasks. Additionally, Shen et al. (2024) assessed the changes in decision-making abilities when LLM assumes different personality traits. Inspired by these studies, we explore the benefit of multi-agent role differentiation and debates for complex reasoning on knowledge graphs.

Method

Task Formulation

Given a knowledge graph $\mathcal{G}$ consisting of N triples, represented as $\{(e_i^l, r_l, e_{i+1}^l)|e_i \in \mathcal{E}, r_l \in \mathcal{R}, i \in [1, I], l \in [1, L]\}$, where e_i^l and e_{i+1}^l denote the head and tail entity, respectively, I is the number of entities, L denotes the number of relations, and r_l is the relation between entities, KGQA requires machines to answer natural language questions q based on retrieved evidence paths $P = \{p_j\}_{j=1}^m$ with p_j representing a triple and m denoting the number of triples. In this paper, we leverage LLMs to reason over P and generate answers $\hat{a}$ word by word.

Overview

As depicted in Fig. 2, given a K-hop question q and the initial topic entity e_i^l within q, our framework first invokes knowledge graphs to retrieve the set of candidate relations R linked to e_i^l. Then, it enables LLMs to filter out the most relevant relation $\hat{r}_l$ from R based on in-context learning. Subsequently, the knowledge graph is invoked again to complete the triple information from $(e_i^l, \hat{r}_l, ?)$ to $(e_i^l, \hat{r}_l, e_{i+1}^l)$. Fourthly, DoG focuses on the current reasoning state and employs LLMs to decide on the subsequent action based on the completed triple: providing a direct answer to the question or performing deep thinking with further iterations. In the latter scenario, a multi-role LLM team leverages the mentioned triple to transform the K-hop question to a K-1 hop (slightly easier) one through debate, with the tail entity e_{i+1}^l being the subsequent topic entity for the simplified question in the next iteration. All of these debate steps are autonomously executed by the LLM team. The iteration will be ended until LLMs generate answers in the fourth step.

Knowledge Graph Invoking

Reasoning on graphs requires LLMs first to identify relevant knowledge triples. To facilitate this, we have designed two interactive interfaces specifically tailored to retrieve these triples from knowledge graphs. The interfaces are invoked as needed, depending on the requirements.

- *get_relations*(e_i^l): This interface is designed to retrieve the candidate relation set R associated with the entity e_i^l. For example, in Fig. 2, it is invoked to retrieve the candidate relation set of `Joe Anderson`.
- *triple_filling*(e_i^l, $\hat{r}_l$): This interface is responsible for obtaining the tail entity $<e_i^l, \hat{r}_l, ?>$ given the head entity and the filtered relation. We will introduce relation filtering in the next subsection.

The underlying mechanisms of these interfaces are implemented through either SPARQL (for Freebase queries) or specific matching (for questions in MetaQA). To facilitate comprehension and generation by LLMs, all entities and relationships above the interfaces are expressed in natural language, with the conversion between a Machine ID (MID) and a corresponding friendly name carried out exclusively within the interfaces. The MID facilitates efficient access to comprehensive details related to the entity. More specifically, in Freebase, the MID is a unique identifier assigned to each entity, allowing for straightforward retrieval of entity-specific information. The friendly name of the MID is a natural language descriptor. For example, the MID of the friendly name `Jamaican` is $m.03\_r3$.

Relation Filtering

Through *get_relations*(e_i^l), we obtain a candidate relation set R associated with the initial entity in the question. Subsequently, DoG selects the optimal relation $\hat{r}_l$ from this set through in-context learning. The prompt and in-context examples are detailed in the *In-context Learning* subsection of the appendix. Specifically, DoG first utilizes LLMs to identify the first-hop problem to be solved in the given question q. Then, it allows LLMs to choose the optimal relation according to the mentioned sub-question. This serves as a guiding principle for relation selection, avoiding the constant reliance on the complete multi-hop question throughout the entire reasoning stage, as seen in previous studies (Jiang et al. 2023a; Sun et al. 2024). We believe this short-sighted greedy strategy can guide a correct progression on the graph, alleviating the need to account for future inferential information regarding the multi-hop question. For example, for the question in Fig. 2 "In what year

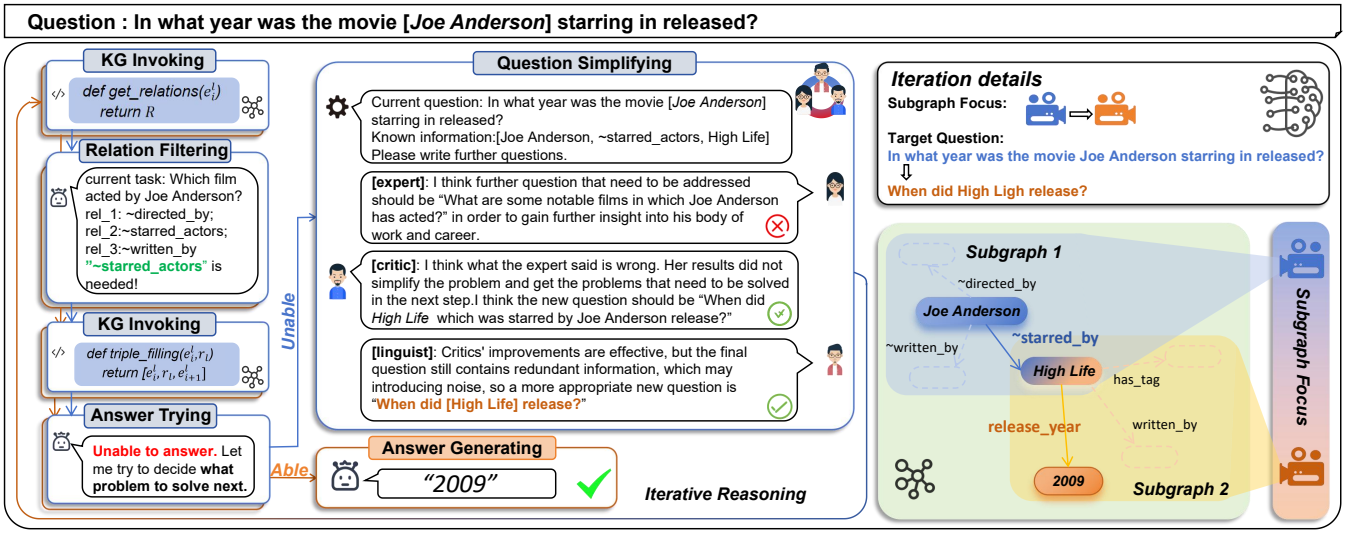

Figure 2: DoG framework. Given a question, our framework first enables LLMs to interact with knowledge graphs to retrieve the most relevant triple. Subsequently, it employs a subgraph-focusing mechanism, allowing LLMs to attempt answering at each reasoning step. If further reasoning is required, DoG leverages a multi-role LLM team to simplify the question from complex to easy based on the retrieved triples.

was the movie Joe Anderson starring in released", the first-hop question to be addressed is "Which film starred Joe Anderson?". The linearized relation set is {~directed_by; ~starred_actors; ~written_by} ("~" represents a passive relationship), from which the optimal relation ~starred_actors can be easily selected.

Answer Trying

After obtaining the optimal relation, our architecture invokes the triple-filling interface *triple_filling*(e_i^l, $\hat{r}_l$) to acquire a complete triple, such as <Joe Anderson, ~starred_actors, High Life> in Fig. 2. Then, DoG utilizes LLMs to determine whether the retrieved triple can sufficiently support answering the question. If the triple is insufficient, DoG prompts LLMs to deeply contemplate the current question based on the provided triple. This allows DoG to generate answers based on a single triple, thus avoiding excessively long and potentially confusing paths composed of multiple triples. The prompt and in-context examples are detailed in the *In-context Learning* subsection of the appendix. Notably, if the maximum iteration limit is reached without successfully generating an answer, the parameterized knowledge of LLMs is utilized to respond.

Question Simplifying

Once LLMs determine that a question is unanswerable with the current retrieved triple, it represents that further exploration is required. Inspired by how humans tackle complex questions, our architecture employs a question-simplifying strategy to transform questions from K hop to K-1 hop based on the retrieved triple. Specifically, DoG utilizes a team of agents with distinct roles to engage in debate, ensuring the reliability of the reasoning process. The debate team consists of three roles.

- Question simplifying expert (R1). This expert provides initial simplifications for questions, which may contain apparent errors. For example, the original question in Fig. 2 is initially simplified as "What are some notable films in which Joe Anderson has acted?". This is far from the intention of the original question.
- Critic (R2). The critic examines the simplification efforts of the above expert and offers suggestions for modifications. For instance, the above question is modified into "When did High Life which was starred by Joe Anderson release?".
- Linguist (R3). This role ensures that the simplified question is not only semantically correct but also free from redundant information of previously resolved sub-questions. For example, the mentioned question is further refined to "When did [High Life] release?".

Due to the interdependency and progressive nature of the roles played by the three agents, DoG employs a one-by-one discussion strategy (Chan et al. 2023). Each agent, implemented by ChatGPT, takes turns contributing to the ongoing optimization of the simplified question, with the statements made by other agents serving as references for guiding subsequent remarks generation. After simplification, we obtain a slightly easier K-1 hop question, prompting LLMs to undergo iteration once again. In this way, the relation in the first-hop sub-question is removed in the simplified question, effectively avoiding the impact of false positive relations. The iteration process, from knowledge graph invocation to question simplification, continues until LLMs make an answerable decision in the answer-trying module. The prompt and in-context examples are shown in the *In-context Learning* subsection of the appendix.

Method	Class	LM	MetaQA			WebQSP	CWQ	WebQ	GrailQA
			1-hop	2-hop	3-hop				
KV-Mem	SL	-	96.2	82.7	48.9	46.7	18.4	-	-
GraftNet		-	97.0	94.8	77.7	66.4	36.8	-	-
PullNet		-	97.0	99.9	91.4	68.1	45.9	-	-
EmbedKGQA		RoBERTa	97.5	98.8	94.8	66.6	-	-	-
NSM		-	97.1	99.9	98.9	68.7	47.6	-	-
TransferNet		BERT	97.5	**100.0**	**100.0**	71.4	48.6	-	-
UniKGQA		RoBERTa	**98.0**	99.9	99.9	**77.2**	**51.2**	-	-
StructGPT	ICL	GPT-3.5-Turbo	97.1	97.3	87.0	72.6	-	-	-
KG-GPT		GPT-3.5-Turbo	96.3	94.4	94.0	-	-	-	-
KB-BINDER		Codex	93.5	**99.9**	**99.5**	74.4	-	-	58.5
ToG		GPT-3.5-Turbo	-	-	-	76.2	57.1	54.5	68.7
DoG	ICL	GPT-3.5-Turbo	98.6	96.6	90.9	88.6	**58.2**	78.2	77.8
DoG		Qwen-14B	99.5	92.4	79.8	83.2	48.1	65.6	74.6
DoG		Llama-3-8B	99.8	91.0	84.8	90.2	55.9	70.8	74.8
DoG		GPT-4	**100.0**	99.0	96.0	**91.0**	56.0	**80.0**	**80.0**

Table 1: Comparison with previous state-of-the-art Supervised Learning (SL) and In-Context Learning based methods. The best results for SL and ICL methods are marked in bold, and the second-best results are underlined. WebQ denotes the WebQuestions dataset. The ToG measurement on WebQSP is based on the F1 score rather than EM (Hits@1).

Experiments

Dataset and Evaluation

We select five public datasets to evaluate the reasoning ability over knowledge graphs: MetaQA (Zhang et al. 2018), WebQSP (Yih et al. 2016), CWQ (Talmor and Berant 2018), WebQuestions (Berant et al. 2013), and GrailQA (Gu et al. 2021). MetaQA comprises a movie ontology derived from the WikiMovies dataset (Miller et al. 2016) and contains three sets of natural language question-answer pairs: 1-hop, 2-hop, and 3-hop. WebQSP contains questions sourced from the WebQuestions dataset, which are answerable using Freebase. CWQ is designed for answering complex questions that require reasoning over multiple web snippets. GraiQA, which tests three-level generalizations including i.i.d., compositional, and zero-shot, covers 3,720 relations and 86 domains from Freebase. Following (Xiong, Bao, and Zhao 2024; Sun et al. 2024), we uniformly sample 500 instances per type for the mentioned five datasets to reduce computational cost. We use `exact match accuracy` (Hits@1) to evaluate the reasoning performance of our framework and baselines following previous works (Jiang et al. 2023a; Xiong, Bao, and Zhao 2024; Sun et al. 2024; Baek, Aji, and Saffari 2023). For the experiment of integrating DoG with GPT-4, we uniformly sample only 100 instances per type from the mentioned datasets to reduce costs.

Implementation Settings

We preprocess the MetaQA dataset to construct a structured knowledge graph, facilitating subsequent query and retrieval operations. A local Virtuoso server is deployed for datasets derived from the Freebase. We utilize the OpenAI API to call ChatGPT (gpt-3.5-turbo-0125) and GPT-4 (gpt-4-0613). Additionally, we employ Qwen-14B and Llama-3-8B, running on 8 V100 GPUs, to verify the flexibility of DoG. The maximum number of debate rounds for the multi-agent team is limited to three, with only the best unique relation being recalled. We implement in-context learning across multiple modules: specifically, 10 exemplars for *Relation Filtering* and *Answer Trying*, and one exemplar for *Question Simplifying*.

Baselines

Inspired by (Jiang et al. 2023a), we compare DoG with previous state-of-the-art supervised learning and in-context learning-based methods, to verify its effectiveness and superiority. Supervised learning: KV-Mem (Miller et al. 2016), GraftNet (Sun et al. 2018), PullNet (Sun, Bedrax-Weiss, and Cohen 2019), EmbedKGQA (Saxena, Tripathi, and Talukdar 2020), NSM (He et al. 2021), TransferNet (Shi et al. 2021), UniKGQA (Jiang et al. 2023b). In-context learning: StructGPT (Jiang et al. 2023a), KG-GPT (Kim et al. 2023), KB-BINDER (Li et al. 2023b), ToG (Sun et al. 2024). The baselines are detailed in the *Baseline Introduction* subsection of the appendix.

Reasoning on Knowledge Graphs

Main Result Table 1 presents a comparison across five public datasets. Taking GPT-3.5 as an example, we observe that DoG enables it to achieve competitive results on MetaQA and the best results on the other four datasets compared with baselines. Specifically, DoG outperforms the best-supervised method, UniKGQA, by 11.4% on WebQSP. Additionally, it surpasses the best in-context learning method, ToG, by 23.7% and 9.1% on WebQuestions and GrailQA, respectively. These datasets comprise complex and compositional questions. Therefore, these results not only highlight the effectiveness and superiority of DoG but also confirm its capability for complex reasoning.

Flexibility Verification We conduct experiments on the aforementioned datasets to explore whether DoG enables other LLMs, including QWen, Llama, and GPT-4, to achieve complex reasoning on knowledge graphs. Experimental results in Table 1 show that DoG facilitates improvements in some cases compared to GPT-3.5. Specifically, DoG with Llama achieves a 1.6% improvement on WebQSP. It also allows GPT-4 to achieve the most significant improvement

Num.	Settings	MetaQA^2	MetaQA^3	WebQSP	CWQ	WebQ	GrailQA	Avg.
1	w/o SF and QS	76.6	38.8	77.4	43.0	67.8	69.3	-
2	w/ SF and R1	91.4$^{+14.8}$	83.4$^{+44.6}$	81.0$^{+3.6}$	50.0$^{+7.0}$	69.8$^{+2.0}$	75.2$^{+5.9}$	+13.0
3	w/ SF, R1 and R2	90.6$^{+14.0}$	85.2$^{+46.6}$	83.6$^{+6.2}$	52.2$^{+9.2}$	72.2$^{+4.4}$	**78.2**$^{+8.9}$	+14.9
4	w/ SF, R1, R2, and R3	**96.6**$^{+20.0}$	**90.9**$^{+52.1}$	**88.6**$^{+11.2}$	**58.2**$^{+15.2}$	**78.2**$^{+10.4}$	77.8$^{+8.5}$	**+19.6**
5	w/ SF and QS'	86.6$^{+10.0}$	68.2$^{+30.6}$	81.4$^{+4.0}$	46.8$^{+3.8}$	71.2$^{+3.4}$	71.8$^{+2.5}$	+9.3

Table 2: Ablation results. MetaQA$^{\#}$ denotes the #-hop split of this dataset. SF and QS refer to the subgraph focusing and question simplifying, respectively. R1, R2, and R3 are the different experts in QS. QS' indicates that the tasks of the mentioned three roles are fused into a single agent. Avg. represents the average performance increase across the datasets.

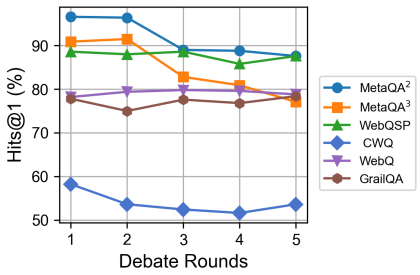

Figure 3: Impact of debate rounds for LLM reasoning on knowledge graphs. It is unnecessary to simplify the question for the 1-hop question within MetaQA.

on the mentioned datasets. These results clearly demonstrate the flexibility and effectiveness of our architecture. We observe that the performance of DoG with Qwen is slightly lower than with other LLMs. This could be attributed to its marginally weaker complex reasoning capabilities compared to other LLMs.

Ablation Studies

We conduct ablation experiments on the aforementioned datasets to analyze the contribution of each component of DoG. The ablation results for DoG with GPT-3.5 are presented in Table 2. We perform experiments on the 2-hop and 3-hop splits of MetaQA, as the 1-hop questions do not require complex reasoning. Row 1 shows the results without the subgraph-focusing and question-simplifying components. In other words, this configuration allows LLMs to answer complex questions directly after collecting the whole set of evidence triples, rather than reasoning step by step. We observe a significant performance decrease compared to the results in Row 4, strongly demonstrating the effectiveness of the mentioned modules. Rows 2 and 3 aim to verify the contribution of the expert role in the debate team. The results show consistent improvements across five public datasets, suggesting that each agent plays a critical role in simplifying questions. This also highlights the importance of transforming complex questions into simpler ones for LLMs step-by-step reasoning on knowledge graphs. Row 5 aims to verify the necessity of the debating process, where the tasks of the three roles are performed by a single agent. The average result decreases by 10.3% compared to Row 4, strongly supporting the effectiveness of the debating mechanism.

Analyses for Debate Rounds

We conduct experiments to explore how the number of debate rounds affects LLM reasoning on knowledge graphs. Fig. 3 shows the performance trend of DoG with GPT-3.5 as the number of debate rounds increases across the five datasets mentioned. We observe that DoG achieves the best results on the majority of datasets with just a single round of debates. Additionally, increasing the number of debate rounds leads to a performance decrease in some datasets. DoG utilizes a one-by-one discussion strategy, which makes each agent aware of the historical debate record. This makes the agents more susceptible to being influenced by the views of others, potentially leading to inaccurate decisions for question simplifications. We may also conclude that the agent is sufficiently strong to achieve the goal of instructions without needing iterative debates.

Exemplar Impacts

DoG leverages in-context learning to guide LLMs in performing relation filtering, question simplification, and answer trying during iterative reasoning. Specifically, DoG provides instructions and exemplars to help LLMs achieve these objectives. We conduct experiments on five public datasets to explore the impact of the number of exemplars on LLM reasoning. Fig. 4 shows the analyses for the mentioned three modules. In *Relation Filtering*, we observe that reasoning performance improves as the number of exemplars increases in the majority of datasets. However, reasoning errors caused by relation filtering account for a large proportion, which we will discuss in the next subsection. In *Question Simplifying*, the performance improvement is not significant with the increase in the number of exemplars, likely due to the complexity of this task. Converting questions from complex to simple step-by-step may be challenging for LLMs, and they may not be able to infer strategies for addressing this issue from exemplars. In *Answer Trying*, we see that reasoning performance improves with the increase in the number of exemplars in most cases. In summary, the number of exemplars plays a critical role in decision-making, especially for less complex tasks. In contrast, for more complex tasks, detailed instructions may have a greater impact on LLM reasoning.

Error Analyses

To analyze the deficiency of DoG, we randomly select 50 failure cases from each dataset, including MetaQA, WebQSP, and GrailQA, for manual inspection. Fig. 5 shows

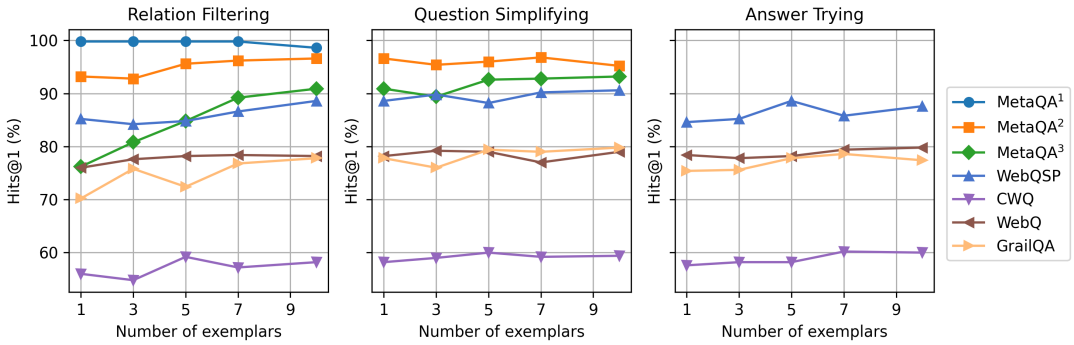

Figure 4: Impacts of the number of exemplars on the performance of LLM reasoning. It is unnecessary to perform question simplifying for the 1-hop question within MetaQA. DoG does not utilize LLMs to generate answers for questions within MetaQA. Instead, it provides answers based on the last retrieved triple after iterative reasoning.

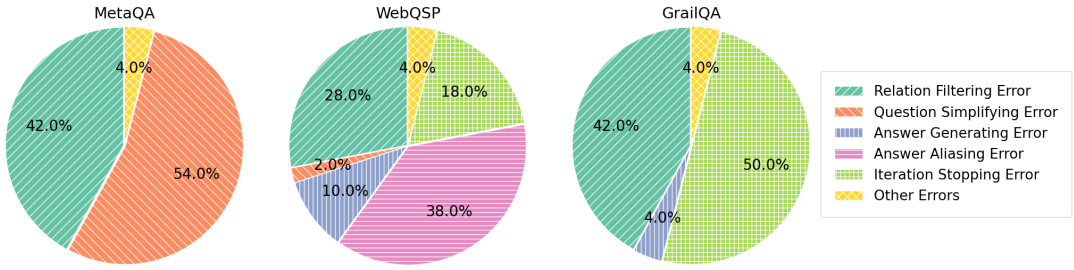

Figure 5: Analysis of 50 sampled failure cases per dataset. We visualize the proportion of factors contributing to errors. We do not perform manual inspection for the failure cases in CWQ and WebQ due to the lack of annotations, such as those for the ground-truth relations.

the proportion of factors contributing to these errors. We observe that relation filtering errors are quite common. This may be caused by too many relations linked to the entities in questions, making it difficult for LLMs to accurately filter the most relevant relation. Iteration stopping errors denote LLMs make inaccurate decisions in the answer-trying module, either terminating the iterative reasoning too early or too late. This type of error is particularly prevalent in GrailQA cases. Answer aliasing errors mean the generated answers do not have the same description or wording as the annotations, even though they are semantically consistent. This error can be mitigated by introducing a rich collection of aliases. Answer generation errors refer to that LLMs provide incorrect answers based on accurately retrieved triples and simplified questions. Question simplifying errors represent that LLMs fail to transform questions from complex to easy. Additionally, other errors account for 4% of the failure cases in each dataset. This type of error often occurs due to API access issues, an excessively long context, or exceeding the token limit per minute. More details can be found in the *Failure Cases* subsection of the appendix.

Conclusion and Future Work

This paper proposes an iterative interactive framework, DoG, for knowledge graph question answering. It leverages the interactive learning and reasoning capabilities of LLMs to perform debating on knowledge graphs. Specifically, it employs a team of multi-role agents to transform questions from complex to simple, enabling LLMs to perform reliable step-by-step reasoning based on the retrieved knowledge triples. Extensive experiments across five public datasets demonstrate the effectiveness and superiority of DoG in the few-shot setting, outperforming nearly all in-context and supervised learning-based baselines. Additionally, the integration results with different LLMs verify its flexibility. In the future, we will explore enhancing relation filtering performance from knowledge graphs given the entity of questions.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (2022YFC3303600), the National Natural Science Foundation of China (U22B2019, 62477037, 62450005, 62437002, 62306229, 62293553, 62372362), the Key Research and Development Program of Shaanxi Province (2024GX-ZDCYL-02-12), the Natural Science Basic Research Program of Shaanxi (2023-JC-YB-593), the Youth Innovation Team of Shaanxi Universities "Multi-modal Data Mining and Fusion", the Shaanxi Undergraduate and Higher Education Teaching Reform Research Program (23BY195), the Youth Talent Support Program of Shaanxi Science and Technology Association (20240113), the China Postdoctoral Science Foundation (2024M752585), and CCF-Zhipu Large Model Innovation

Fund (NO.CCF-Zhipu202413).

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Baek, J.; Aji, A.; and Saffari, A. 2023. Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering. In *ACL*.

Bai, J.; Liu, X.; Wang, W.; Luo, C.; and Song, Y. 2023. Complex query answering on eventuality knowledge graph with implicit logical constraints. In *NeurIPS*, 30534–30553.

Berant, J.; Chou, A.; Frostig, R.; and Liang, P. 2013. Semantic parsing on freebase from question-answer pairs. In *EMNLP*, 1533–1544.

Bollacker, K. D.; Evans, C.; Paritosh, P. K.; Sturge, T.; and Taylor, J. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD*.

Chan, C.-M.; Chen, W.; Su, Y.; Yu, J.; Xue, W.; Zhang, S.; Fu, J.; and Liu, Z. 2023. ChatEval: Towards Better LLM-based Evaluators through Multi-Agent Debate. In *ICLR*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Cui, Y.; Wang, Y.; Sun, Z.; Liu, W.; Jiang, Y.; Han, K.; and Hu, W. 2023. Lifelong embedding learning and transfer for growing knowledge graphs. In *AAAI*, 4217–4224.

Dong, Q.; Li, L.; Dai, D.; Zheng, C.; Wu, Z.; Chang, B.; Sun, X.; Xu, J.; and Sui, Z. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Du, Y.; Li, S.; Torralba, A.; Tenenbaum, J. B.; and Mordatch, I. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

Gu, Y.; Deng, X.; and Su, Y. 2023. Don't Generate, Discriminate: A Proposal for Grounding Language Models to Real-World Environments. In *ACL*, 4928–4949.

Gu, Y.; Kase, S.; Vanni, M.; Sadler, B.; Liang, P.; Yan, X.; and Su, Y. 2021. Beyond iid: three levels of generalization for question answering on knowledge bases. In *WWW*, 3477–3488.

Guan, X.; Liu, Y.; Lin, H.; Lu, Y.; He, B.; Han, X.; and Sun, L. 2024. Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting. In *AAAI*, 18126–18134.

Gunjal, A.; Yin, J.; and Bas, E. 2024. Detecting and preventing hallucinations in large vision language models. In *AAAI*, 18135–18143.

Guo, T.; Yang, Q.; Wang, C.; Liu, Y.; Li, P.; Tang, J.; Li, D.; and Wen, Y. 2023a. KnowledgeNavigator: Leveraging Large Language Models for Enhanced Reasoning over Knowledge Graph. *arXiv preprint arXiv:2312.15880*.

Guo, T.; Yang, Q.; Wang, C.; Liu, Y.; Li, P.; Tang, J.; Li, D.; and Wen, Y. 2023b. KnowledgeNavigator: Leveraging Large Language Models for Enhanced Reasoning over Knowledge Graph. *arXiv preprint arXiv:2312.15880*.

He, G.; Lan, Y.; Jiang, J.; Zhao, W. X.; and Wen, J.-R. 2021. Improving multi-hop knowledge base question answering by learning intermediate supervision signals. In *WSDM*, 553–561.

Hu, N.; Chen, J.; Wu, Y.; Qi, G.; Bi, S.; Wu, T.; and Pan, J. Z. 2024. Benchmarking Large Language Models in Complex Question Answering Attribution using Knowledge Graphs. *CoRR*, abs/2401.14640.

Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; and Yu, P. S. 2022. A Survey on Knowledge Graphs: Representation, Acquisition, and Applications. *IEEE TNNLS*, 33(2): 494–514.

Jiang, J.; Zhou, K.; Ye, K.; Zhao, X.; Wen, J.-R.; et al. 2023a. StructGPT: A General Framework for Large Language Model to Reason over Structured Data. In *EMNLP*.

Jiang, J.; Zhou, K.; Zhao, W. X.; Song, Y.; Zhu, C.; Zhu, H.; and Wen, J.-R. 2024. KG-Agent: An Efficient Autonomous Agent Framework for Complex Reasoning over Knowledge Graph. *arXiv preprint arXiv:2402.11163*.

Jiang, J.; Zhou, K.; Zhao, X.; and Wen, J. 2023b. UniKGQA: Unified Retrieval and Reasoning for Solving Multi-hop Question Answering Over Knowledge Graph. In *ICLR*.

Kim, J.; Kwon, Y.; Jo, Y.; and Choi, E. 2023. KG-GPT: A General Framework for Reasoning on Knowledge Graphs Using Large Language Models. In *EMNLP*.

Kirk, J. R.; Wray, R. E.; Lindes, P.; and Laird, J. E. 2024. Improving Knowledge Extraction from LLMs for Task Learning through Agent Analysis. In *AAAI*, 18390–18398.

Kong, A.; Zhao, S.; Chen, H.; Li, Q.; Qin, Y.; Sun, R.; and Zhou, X. 2023. Better zero-shot reasoning with role-play prompting. *arXiv preprint arXiv:2308.07702*.

Li, G.; Hammoud, H.; Itani, H.; Khizbullin, D.; and Ghanem, B. 2023a. CAMEL: Communicative Agents for "Mind" Exploration of Large Language Model Society. In *NeurIPS*, 51991–52008.

Li, T.; Ma, X.; Zhuang, A.; Gu, Y.; Su, Y.; and Chen, W. 2023b. Few-shot In-context Learning on Knowledge Base Question Answering. In *ACL*, 6966–6980.

Li, Z.; Fan, S.; Gu, Y.; Li, X.; Duan, Z.; Dong, B.; Liu, N.; and Wang, J. 2024. Flexkbqa: A flexible llm-powered framework for few-shot knowledge base question answering. In *AAAI*, 18608–18616.

Liang, T.; He, Z.; Jiao, W.; Wang, X.; Wang, Y.; Wang, R.; Yang, Y.; Tu, Z.; and Shi, S. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*.

Lin, Q.; Liu, J.; Mao, R.; Xu, F.; and Cambria, E. 2023. TECHS: Temporal logical graph networks for explainable extrapolation reasoning. In *ACL*, 1281–1293.

Liu, H.; Xue, W.; Chen, Y.; Chen, D.; Zhao, X.; Wang, K.; Hou, L.; Li, R.; and Peng, W. 2024. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253*.

Liu, X.; Yu, H.; Zhang, H.; Xu, Y.; Lei, X.; Lai, H.; Gu, Y.; Ding, H.; Men, K.; Yang, K.; et al. 2023. AgentBench: Evaluating LLMs as Agents. In *ICLR*.

Luo, L.; Li, Y.-F.; Haf, R.; and Pan, S. 2023. Reasoning on Graphs: Faithful and Interpretable Large Language Model Reasoning. In *ICLR*.

Ma, J.; Wang, P.; Wang, Z.; Kong, D.; Hu, M.; Han, T.; and Liu, J. 2023. Adaptive loose optimization for robust question answering. *arXiv preprint arXiv:2305.03971*.

Mandi, Z.; Jain, S.; and Song, S. 2023. Roco: Dialectic multi-robot collaboration with large language models. *arXiv preprint arXiv:2307.04738*.

Miller, A.; Fisch, A.; Dodge, J.; Karimi, A.-H.; Bordes, A.; and Weston, J. 2016. Key-Value Memory Networks for Directly Reading Documents. In *EMNLP*, 1400–1409.

Nie, Z.; Zhang, R.; Wang, Z.; and Liu, X. 2024. Code-style in-context learning for knowledge-based question answering. In *AAAI*, 18833–18841.

Qian, C.; Cong, X.; Yang, C.; Chen, W.; Su, Y.; Xu, J.; Liu, Z.; and Sun, M. 2023. Communicative agents for software development. *arXiv preprint arXiv:2307.07924*.

Quintero-Narvaez, C. E.; and Monroy, R. 2024. Integrating Knowledge Graph Data with Large Language Models for Explainable Inference. In *WSDM*, 1198–1199.

Rawte, V.; Sheth, A.; and Das, A. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*.

Saxena, A.; Tripathi, A.; and Talukdar, P. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *ACL*, 4498–4507.

Schumann, R.; Zhu, W.; Feng, W.; Fu, T.-J.; Riezler, S.; and Wang, W. Y. 2024. Velma: Verbalization embodiment of llm agents for vision and language navigation in street view. In *AAAI*, 18924–18933.

Shen, C.; Xie, G.; Zhang, X.; and Xu, J. 2024. On the Decision-Making Abilities in Role-Playing using Large Language Models. *arXiv preprint arXiv:2402.18807*.

Shi, J.; Cao, S.; Hou, L.; Li, J.; and Zhang, H. 2021. TransferNet: An Effective and Transparent Framework for Multi-hop Question Answering over Relation Graph. In *EMNLP*, 4149–4158.

Shi, X.; Zhu, Z.; Zhang, Z.; and Li, C. 2023. Hallucination mitigation in natural language generation from large-scale open-domain knowledge graphs. In *EMNLP*, 12506–12521.

Shu, Y.; Yu, Z.; Li, Y.; Karlsson, B.; Ma, T.; Qu, Y.; and Lin, C.-Y. 2022. TIARA: Multi-grained Retrieval for Robust Question Answering over Large Knowledge Base. In *EMNLP*, 8108–8121.

Sun, H.; Bedrax-Weiss, T.; and Cohen, W. W. 2019. Pullnet: Open domain question answering with iterative retrieval on knowledge bases and text. *arXiv preprint arXiv:1904.09537*.

Sun, H.; Dhingra, B.; Zaheer, M.; Mazaitis, K.; Salakhutdinov, R.; and Cohen, W. 2018. Open Domain Question Answering Using Early Fusion of Knowledge Bases and Text. In *EMNLP*, 4231–4242.

Sun, J.; Xu, C.; Tang, L.; Wang, S.; Lin, C.; Gong, Y.; Ni, L.; Shum, H.-Y.; and Guo, J. 2024. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph. In *ICLR*.

Talmor, A.; and Berant, J. 2018. The Web as a Knowledge-Base for Answering Complex Questions. In *NAACL*, 641–651.

Toroghi, A.; Guo, W.; Pour, M. M. A.; and Sanner, S. 2024. Right for Right Reasons: Large Language Models for Verifiable Commonsense Knowledge Graph Question Answering. *arXiv preprint arXiv:2403.01390*.

Waldendorf, J.; Haddow, B.; and Birch, A. 2024. Contrastive Decoding Reduces Hallucinations in Large Multilingual Machine Translation Models. In *ECAI*, 2526–2539.

Wang, S.; Scells, H.; Koopman, B.; and Zuccon, G. 2023. Can ChatGPT write a good boolean query for systematic review literature search? In *SIGIR*, 1426–1436.

Wang, Y.; Lipka, N.; Rossi, R. A.; Siu, A.; Zhang, R.; and Derr, T. 2024. Knowledge graph prompting for multi-document question answering. In *AAAI*, 19206–19214.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 24824–24837.

Wei, Y.; Huang, Q.; Zhang, Y.; and Kwok, J. 2023. KICGPT: Large Language Model with Knowledge in Context for Knowledge Graph Completion. In *Findings of EMNLP*, 8667–8683.

Xiong, G.; Bao, J.; and Zhao, W. 2024. Interactive-KBQA: Multi-Turn Interactions for Knowledge Base Question Answering with Large Language Models. *CoRR*, abs/2402.15131.

Yang, D.; Yang, K.; Wang, Y.; Liu, J.; Xu, Z.; Yin, R.; Zhai, P.; and Zhang, L. 2023. How2comm: communication-efficient and collaboration-pragmatic multi-agent perception. In *NeurIPS*, 25151–25164.

Ye, X.; Yavuz, S.; Hashimoto, K.; Zhou, Y.; and Xiong, C. 2022. RNG-KBQA: Generation Augmented Iterative Ranking for Knowledge Base Question Answering. In *ACL*, 6032–6043.

Yih, W.-t.; Richardson, M.; Meek, C.; Chang, M.-W.; and Suh, J. 2016. The value of semantic parse labeling for knowledge base question answering. In *ACL*, 201–206.

Zhang, J.; Xu, X.; and Deng, S. 2023. Exploring collaboration mechanisms for llm agents: A social psychology view. *arXiv preprint arXiv:2310.02124*.

Zhang, Y.; Dai, H.; Kozareva, Z.; Smola, A.; and Song, L. 2018. Variational reasoning for question answering with knowledge graph. In *AAAI*.

Zhang, Y.; Li, Y.; Cui, L.; Cai, D.; Liu, L.; Fu, T.; Huang, X.; Zhao, E.; Zhang, Y.; Chen, Y.; et al. 2023. Siren's song in the AI ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Zhao, A.; Huang, D.; Xu, Q.; Lin, M.; Liu, Y.-J.; and Huang, G. 2024. Expel: Llm agents are experiential learners. In *AAAI*, 19632–19642.