Promoting Knowledge Base Question Answering by Directing LLMs to Generate Task-relevant Logical Forms

Jianqi Gao[1], Jian Cao[1*], Ranran Bu[1], Nengjun Zhu[2], Wei Guan[1], Hang Yu[2]

[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]School of Computer Engineering and Science, Shanghai University
{193139, cao-jian, buranran, guan-wei}@sjtu.edu.cn, {zhu_nj, yuhang}@shu.edu.cn

Abstract

Knowledge base question answering (KBQA) refers to the system that produces answers to user queries by reasoning with a large-scale structured knowledge base. Advanced works have achieved great success either by generating logical forms (LF) or directly generating answers. Although the former typically yields better performance, these generated LF could be inaccurate, e.g., non-executable. In this regard, large language models (LLMs) have shown exciting potential for accurate generation. However, it is challenging to fine-tune LLMs to generate LF. This is because the context retrieved for prediction typically leads to an excessive number of reasoning paths. In this context, LLMs can generate numerous LF corresponding to these reasoning paths, but a few LF can result in correct answers. Thus, fine-tuning LLMs to generate answer-relevant LF would conflict with the prior knowledge of the LLMs. In this work, we propose a novel learning framework, FM-KBQA, to fine-tune LLMs using multi-task learning for KBQA. Specifically, we propose to fine-tune LLMs using an additional objective: generating the index of reasoning paths that lead to correct answers. This will direct LLMs to pay attention to answer-relevant paths among numerous reasoning paths by completing a simple task where the selected reasoning paths can be supplementary for non-executable LF. Directly generating answers can make LLMs pay attention to the answer-relevant reasoning paths, but it is much more challenging than generating the index of reasoning paths. To verify FM-KBQA's effectiveness, we conduct experiments on mainstream benchmarks, such as WebQuestionsSP (WQSP) and ComplexWebQuestions (CWQ). Extensive evaluations across two public benchmark datasets underscore the superiority of FM-KBQA over current state-of-the-art methods.

Introduction

Knowledge base question answering (KBQA) refers to the system that answers natural language questions based on a large-scale structured knowledge base (Miller et al. 2016). Existing methods can be divided into two categories based on the answer-generation approach. Some works propose to generate answers in a retrieving-then-generating manner (Sun, Bedrax-Weiss, and Cohen 2019; Saxena, Kochsiek, and Gemulla 2022), where question-related information is first retrieved and then processed to generate answers. This is known as the direct-answer-prediction approach. The other works propose to produce answers in a generating-then-retrieving scheme (Luo et al. 2023b; Gu and Su 2022), where the question is parsed into some specific forms used to retrieve answers from a knowledge base (KB). This is known as the semantic parsing-based approach.

Producing answers using question-related information, i.e., a subgraph of a structured KB, is a straightforward approach. For example, PullNet (Sun, Bedrax-Weiss, and Cohen 2019) retrieves a subgraph of KB related to the input question and applies graph neural networks to predict the answer entities in the subgraphs. KGT5 (Saxena, Kochsiek, and Gemulla 2022) uses a sequence-to-sequence framework to directly generate answers only based on the input question. Although this approach is intuitive and can always produce answers, it usually underperforms semantic parsing-based methods on public benchmarks (Talmor and Berant 2018a; Gu et al. 2021, 2022). Semantic parsing-based methods mainly focus on parsing the input question into logical forms (LF), which is executed by an external executor, e.g., a SPARQL server, to retrieve answers. To make the generated LF accurate, ReTrack (Chen et al. 2021) uses a grammar-based decoder to generate LF based on pre-defined grammar rules, and a semantic checker to discourage generating programs that are semantically inconsistent with KB. TIARA (Shu et al. 2022a) proposes a multi-grained retrieval method to select relevant KB context for LF generation. Although previous empirical results (Ye et al. 2021; Das et al. 2021; Gu et al. 2022) show that the semantic parsing based methods can produce more accurate answers over benchmark datasets, these generated LF could be inaccurate, e.g., non-executable (Yu et al. 2022a).

Regarding accurate generation, large language models (LLMs) (Touvron et al. 2023; Luo et al. 2023c; Sui et al. 2024; Lu et al. 2022) have shown exciting success in various scenarios. Thus, employing LLMs for LF generation is a promising approach to promoting KBQA. Despite their impressive performance, LLMs have substantial limitations when facing complex knowledge reasoning tasks (Luo et al. 2023c; Sui et al. 2024) that require deep and responsible reasoning. Thus, some works leverage retrieved facts from the KB to prompt LLMs to improve reasoning perfor-

*Corresponding author is Jian Cao

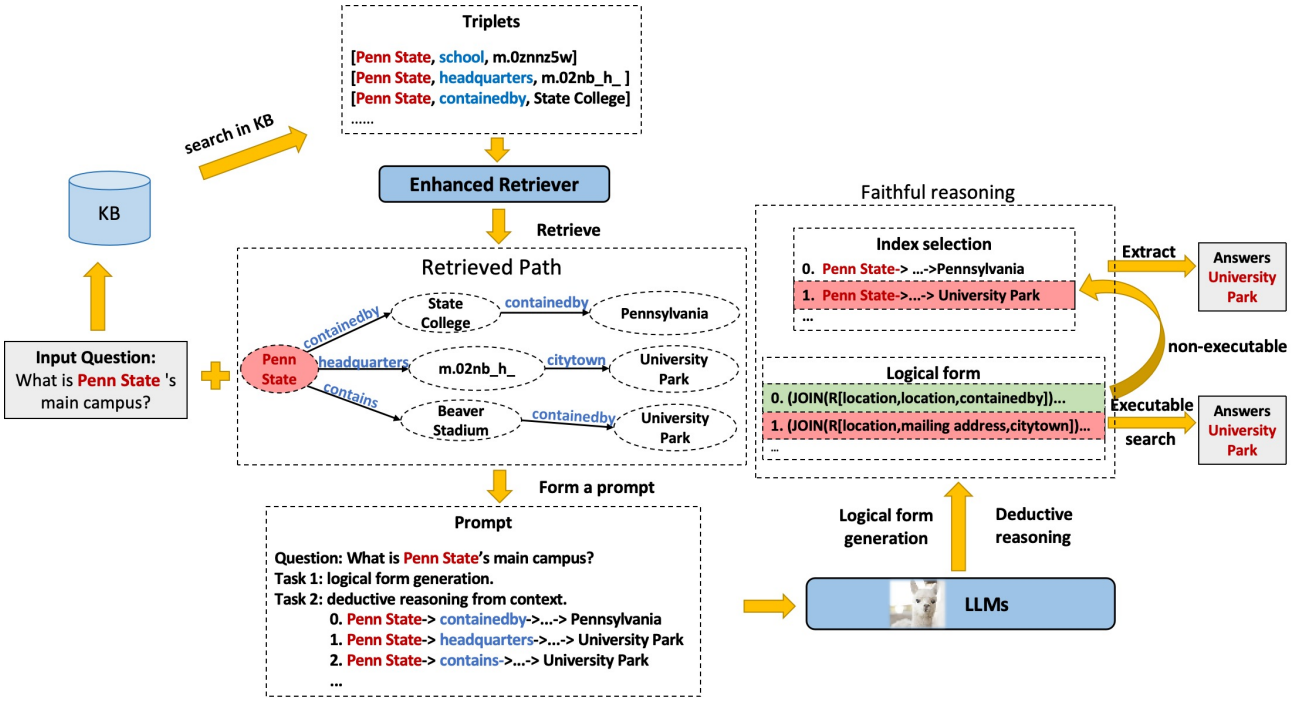

Figure 1: Overview of FM-KBQA. FM-KBQA enhances KBQA by providing the correct retrieval paths through an enhanced retriever, and guiding LLMs to generate task-relevant logical forms.

mance (Karpukhin et al. 2020b; Sun et al. 2023a). Meanwhile, advanced works propose to fine-tune LLMs to generate LF based on retrieved context and queries (Yu et al. 2022b; Luo et al. 2023a).

However, it could be challenging to fine-tune LLMs to generate logical forms (LF). This is because the context retrieved for answer prediction typically leads to an excessive number of reasoning paths. In this context, LLMs can generate numerous LF corresponding to these reasoning paths, but merely a few LF can result in correct answers. Thus, fine-tuning LLMs to generate answer-relevant LF would conflict with LLMs' prior knowledge, i.e., generating numerous LF. This aligns well with our experimental observations, i.e., results shown in Table 3. Namely, fine-tuning LLMs with queries and the retrieved context would underperform the approach of fine-tuning LLMs with only input queries (Luo et al. 2023b; Yu et al. 2022b). In this regard, recent work proposes to encourage LLMs to predict the answers when fine-tuning LLMs to predict LF, which makes LLMs pay attention to the answer-relevant reasoning paths. However, it is challenging to encourage LLMs to generate both LF and answers simultaneously. The intuition is consistent with the observation in the literature. In particular, directly generating answers is a different approach than generating LF. Therefore, there could be compromises in the optimization process of these objectives, leading to suboptimal performance of either the LF prediction or the direct answer prediction. Although multi-objective optimization has been widely applied to improving multi-task learning (Sener and Koltun 2018), it is challenging to model and mitigate objective conflicts in LLMs fine-tuning. This challenge motivates a fundamental question:

> *Can we design a simple objective to encourage LLMs to focus on answer-relevant reasoning paths when predicting logical forms?*

If the additional objective is simple, we can safely make the main objective, predicting LF, dominate the optimization process. In this work, we give an affirmative answer to this question by proposing a novel learning framework FM-KBQA, to *f*ine-tune LLMs using *m*ulti-task learning for *KBQA*. Specifically, we propose to fine-tune LLMs using an additional simple objective: generating the index of reasoning paths that lead to correct answers. This will direct LLMs to pay attention to answer-relevant paths among numerous reasoning paths by completing a simple task where the selected reasoning paths can be supplementary for non-executable LF. Directly generating answers can make LLMs pay attention to the answer-relevant reasoning paths, but it is much more challenging than generating the index of reasoning paths. To verify FM-KBQA's effectiveness, we conduct experiments on mainstream benchmarks, such as WebQuestionsSP (WQSP) and ComplexWebQuestions (CWQ). Comprehensive results show that Llama-2-7B, fine-tuned with the proposed FM-KBQA, can outperform all baselines (e.g., GPT4) with the retrieved context, achieving new state-of-the-art performance.

Our contribution can be summarized as follows.

- We empirically find that, in KBQA, augmenting user

queries with the retrieved context to prompt LLMs may fail to improve the quality of predicted logical forms.

- In this work, we attribute the counterintuitive observation to the conflict between the fine-tuning objective and the LLMs' prior knowledge. Namely, the retrieved context can lead to numerous reasoning paths, thus LLMs can generate numerous corresponding LF. However, the fine-tuning objective enforces LLMs to generate a few logical forms, leading to conflict.
- We propose a novel learning framework, FM-KBQA, to fine-tune LLMs with a simple additional objective function, which directs LLMs to focus on answer-relevant reasoning paths by encouraging LLMs to predict the index of answer-relevant reasoning paths. Comprehensive experimental results demonstrate that FM-KBQA can outperform baselines and achieve new state-of-the-art performance.

Related Work

The current KBQA primarily consists of two types: Information Retrieval-based KBQA and Semantic Parsing-based KBQA.

Information Retrieval-based KBQA retrieves information related to the question, then processes it to generate an answer (Sun, Bedrax-Weiss, and Cohen 2019; Saxena, Kochsiek, and Gemulla 2022). It can be divided into two stages: Information Retrieval (Sun et al. 2018; He et al. 2021a) and Knowledge Reasoning (Miller et al. 2016; Sun et al. 2018; Sun, Bedrax-Weiss, and Cohen 2019; Zhang et al. 2021). Information Retrieval aims at selecting relevant triples from the large-scale knowledge graph (KG) to form paths relevant to the question. Recent advancements in dense retrieval such as BM25 (Robertson, Zaragoza et al. 2009), Dense Passage Retrieval (DPR) (Karpukhin et al. 2020a), and Contriever (Izacard et al. 2021) convert queries and documents into low-dimensional dense vectors, and semantic similarity is measured by vector distance metrics (e.g., cosine similarity) to select question-relevant retrieval paths. Knowledge reasoning focuses on inferring the final answer based on the retrieved paths (Sun, Bedrax-Weiss, and Cohen 2019; Jiang et al. 2022a). Recent studies such as UniK-QA (Oguz et al. 2020a) and PullNet (Sun, Bedrax-Weiss, and Cohen 2019) use specialized network architectures (e.g., graph convolutional networks) to simulate multi-hop reasoning processes. However, these methods have limited reasoning capabilities. In the area of LLMs, ToG (Sun et al. 2023b) leverages the reasoning capabilities of open-source LLMs to iteratively explore various possible reasoning paths on KG to obtain the final answer. RoG (Luo et al. 2023c) proposes a planning retrieval reasoning framework that collaborates fine-tuning LLMs with KG to achieve faithful and explainable reasoning. FiDeLiS (Sui et al. 2024) proposes a retrieval-exploration interactive method that applies the logic and common sense reasoning of LLMs to the retrieval and reasoning of KG, achieving state-of-the-art performance. However, a large portion of the paths retrieved in large-scale KG are erroneous and invalid, which severely limits the performance of the model during reasoning.

Semantic Parsing-based KBQA focuses on parsing questions into a structured query language (e.g. logical forms) and executes them through a query engine to obtain answers (Lan, Wang, and Jiang 2019; Das et al. 2021; Huang, Kim, and Zou 2021). RnG-KBQA (Ye et al. 2021) enumerates potential LF based on input entities and employs a ranking and generation framework to determine LF. ArcaneQA (Gu and Su 2022) dynamically generates LF based on intermediate execution results. TIARA(Shu et al. 2022b) enumerates candidate entities, logical forms, and related schemas, feeding these multi-grained contexts into PLMs, and then decodes the final logical form using constraints. In the area of LLMs, DECAF (Yu et al. 2022a) combines LF and retrieved paths to generate answers using LLMs. ChatKBQA (Luo et al. 2023b) generates LF through fine-tuning LLMs based solely on the input question, and obtains the final answers through query engines. Compared with Information Retrieval-based KBQA, Semantic Parsing-based KBQA gets rid of the more complicated path retrieval process and has a higher accuracy. However, this method may generate some non-executable or invalid LF, which greatly limits its performance.

Proposed Method

In this section, we begin by presenting the problem's definition. Subsequently, we outline the implementation of FM-KBQA (as shown in Figure 1), which mainly includes three parts: (1) Constructing an enhanced retriever to retrieve reasoning paths from the KG; (2) Building a task-related LF generation module to generate task-related LF; (3) Performing faithful reasoning based on the generated LF and the selected reasoning paths.

Problem Definition

KBQA is a complex reasoning task aimed at producing answers from KG to respond to user questions. Specifically, given a natural language question q and a KG $\mathcal{G}$, the goal is to use the relational information r embedded in $\mathcal{G}$ to construct a mapping function f that can accurately predict answers a, where $a \in \mathcal{A}_q$ and $\mathcal{A}_q$ is the set of possible answers for the question q. Consistent with the previous work (Sun et al. 2020; Jiang et al. 2022b), our work assumes that the entities e_q referenced in q are pre-identified and have been appropriately aligned with their counterparts residing in the KG $\mathcal{G}$.

Enhanced Retriever

The retrieval module retrieves the relation paths related to the question from the knowledge base according to the input question. In this regard, there are two main methods: sparse retrieval (Robertson, Zaragoza et al. 2009) and dense retrieval (Karpukhin et al. 2020a). For an input question q, these methods apply the encoder $\mathrm{EC}(\cdot)$ to obtain its representation and then retrieve passages based on the similarity: $\mathrm{Iretrieve} = \mathrm{argtop} - k_i(\mathrm{EC}(p_i) \cdot \mathrm{EC}(q))$. However, these methods mainly involved setting similarity thresholds for filtering, leading to challenges in adapting to different KG.

In this work, we employ a generative model T5 (Raffel et al. 2020a) for a nuanced understanding of facts, enabling rapid adaptation to filter information across diverse KG. Specifically, we employ a prompt X = Question : q, Fact : r to textually represent both the question q and the factual relation r. Then, the prompt X serves as the input for the pre-trained generative model. Our learning objective is to maximize the likelihood of the token t_i given the input text X and the tokens $t_{<i}$ in the ground class $k \in K = \{yes, no\}$ with the objective function as follows:

$$\mathcal{L}_{\text{er}} = -\sum_{i=1}^{|k|} \log P\left(t_i | t_{<i}, X\right) \tag{1}$$

where $P\left(t_i | t_{<i}, X\right)$ is the log-likelihood of the i-th token of the ground class K. We use special tokens *"<extra_id_0>yes<extra_id_1>"* or *"<extra_id_0>no<extra_id_1>"* to represent k, making it easier to extract labels "*yes*" or "*no*" from the generated content. Besides, we propose a training method based on the confidence advantage of the positive set to reduce model sensitivity during the learning process. Specifically, we seek to ensure that the probability of the positive sample set outputting "*yes*" exceeds the probability of the negative sample set outputting "*no*".

$$\mathcal{L}_{\text{con}} = \log\left(1 + \sum_{i \in \Omega_{\text{neg}}, j \in \Omega_{\text{pos}}} e^{\lambda(s_i - s_j)}\right) \tag{2}$$

where λ is a margin value, Ω_{neg} refers to the negative sample set, and Ω_{pos} represents the positive sample set and s is the similarity score.

Task-relevant Logical Forms Generation

Training data preparation: To construct instruction fine-tuning training data, we first convert the SPARQL corresponding to the natural language questions of the training set in the KBQA dataset into equivalent LF, and then replace the entity IDs in these LF (e.g., " *m.03_dwn*") with corresponding entity labels (e.g., " *Lou Seal*") to allow LLMs to better understand the entity semantics. We then convert natural language questions (e.g., " *Lou Seal is the mascot for the team that last won the World Series when?*") into logical forms (e.g., " *(ARGMAX (JOIN (R [sports, sports team, championships]) (JOIN [sports, sports team, team mascot] [Lou Seal])) [time, event, start date])*").

Task-relevant logical forms generation. According to the transformed LF, we formulate the generation of LF as an optimization problem, aiming to maximize the probability of LF to question q from the KG $\mathcal{G}$.

$$P_\theta(l|q, \mathcal{G}) = \sum_{z \in \mathcal{Z}_l} P(l|q, z, \mathcal{G}) P_\theta(z|q) \tag{3}$$

where l is the ground truth LF, θ denotes the parameters of LLMs, z denotes the LF generated by LLMs, and $\mathcal{Z}_l$ denotes the set of possible LF. The latter term $P_\theta(z|q)$ is the probability of generating a LF grounded by KG, the former term $P(l|q, z, \mathcal{G})$ is the probability of reasoning a LF given the question q, logical forms z, and KG $\mathcal{G}$, which is computed by the reasoning-retrieval module.

Despite the great potential of LLMs, these generated LF may be inaccurate, e.g., non-executable. In particular, the retrieved context can lead to many reasoning paths, so that the LLMs can generate many corresponding LF. However, the fine-tuning objective forces the LLMs to generate answer-related LF, which may lead to conflicts. Namely, LLMs can generate numerous LF, while the designed objective is to generate answer-relevant LF. This leads to the conflict of prior knowledge of LLMs. To address this issue, we propose to fine-tune LLMs with a simple additional objective function, which guides the LLMs to focus on answer-related reasoning paths by encouraging it to predict the index of answer-related reasoning paths. Specifically, we formulate the generation of reasoning paths as an optimization problem, aiming to maximize the probability of the reasoning path to question q from the KG $\mathcal{G}$.

$$P_\theta(I_r|q, \mathcal{G}) = \sum_{r \in \mathcal{R}_{\text{p}}} P(I_r|q, r, \mathcal{G}) P_\theta(r|q) \tag{4}$$

where I_r stands for the index of the reasoning path r, and R_p denotes the set of possible reasoning paths. Note that we are merely interested in the answer-relevant reasoning path. Thus, we have $I_r = \texttt{None}$ if r is not answer-relevant. The final objective function of FM-KBQA is the combination of LF optimization and reasoning path optimization, which can be formulated as follows.

$$\mathcal{L} = \log P_\theta(l|q, \mathcal{G}) + \log P_\theta(I_r|q, \mathcal{G}) \tag{5}$$

Here, we adopt the same LLMs for both LF generation and reasoning path selection, which are jointly trained in a multi-task learning manner. We discuss the implementation details of these two tasks in the following subsections.

Fine-tuning process. Given the retrieval relation path and the training data, we design a simple instruction template that takes the question and reasoning path as input:

> Given a question: (*Question*), and a series of Propositions:
> Proposition *i*: the premise is (*Reasoning path*), its conclusion is (*the statement of the question* + *the tail entity*)
> Task 1: Generate logical forms just based on the question.
> Task 2: Verify which deductive reasoning is correct for the given question in a deductive manner, if it exists, return the correct number of deductive reasoning, if doesn't, return "no".

Here, *Question* is the question q. *Reasoning path* is the reasoning path retrieved through fine-tuned T5, i is the i-th proposition generated based on the i-th path. For *the statement of the question*, we transform the question into a conclusion and fill the position of the conclusion answer entity with a placeholder. This process is generated by designing the prompt input into LLMs GPT3.5. For example, for the question: "*what is the name of Justin bieber's*

brother", its conclusion is described as "*Justin Bieber's brother's name is *placeholder**." Then we can use the last tail entity of the reasoning path to replace **placeholder**. This design enables LLMs to select faithful reasoning paths based on the questions, using their expertise in deductive reasoning (Ling et al. 2024; Sui et al. 2024). Then, the LLMs need to generate LF and further output the number of the correct proposition. It is trained according to Eq. 5 through multi-task learning.

Faithful Reasoning

The faithful reasoning module takes the question q and a set of reasoning paths r as input and generates LF and the correct path index. Similarly, we design a reasoning instruction prompt to guide LLMs in conducting reasoning based on the question and the retrieved reasoning paths. The reasoning process can be written as follows.

$$\text{Answers} = \begin{cases} \arg\max(P_\theta(l_i|q,\mathcal{G})) & \text{if } l_i \text{ is executable} \\ r_{\text{correct}} & \text{otherwise,} \end{cases} \tag{6}$$

where l_i represents the i-th logical form. We perform beam search to generate LF and reasoning path index, selecting the executable LF with the highest probability ($\arg\max$), and convert it into a SPARQL query to retrieve the answer from the KG. If none of the generated LF are executable, we directly select the reasoning path corresponding to the output index of the LLMs and combine the tail entities of these reasoning paths as the final answer. Namely, we construct $r_{\text{correct}} = \{r|I_r \neq \texttt{None}\}$.

Experiments

Experiment Settings

In this section, we present the experimental setup, main results, and analysis of our proposed approach.

Datasets. In this paper, we employ two standard KBQA datasets: WebQuestionSP (WQSP) (Yih et al. 2016) and ComplexWebQuestions (CWQ) (Talmor and Berant 2018b), both of which can be reasoned on Freebase KG (Bollacker 2008). For the WQSP dataset, it contains 4,737 simple natural language questions paired with SPARQL queries, For the CWQ dataset, it contains 34,689 more complex questions with SPARQL queries.

Baselines. We compare our method with several existing KBQA approaches.

Rigel (Sen, Saffari, and Oliya 2021) proposes a technique to improve end-to-end question answering by leveraging differentiable KG and adding an intersection operation to handle multiple-entity questions. TIARA (Shu et al. 2022b) enhances question answering over knowledge bases by focusing on relevant contexts and using constrained decoding to reduce errors. UniK-QA (Oguz et al. 2020b) integrates structured, unstructured, and semi-structured knowledge sources by flattening them into text and applying a unified retriever-reader model. UniKGQA (Jiang et al. 2022b) combines retrieval and reasoning by using a unified architecture with semantic matching and information propagation modules for multi-hop KBQA. HGNet (Chen et al. 2022) proposes a hierarchical query graph generation method, consisting of an outlining stage for structural constraints, followed by a filling stage for instance selection. RoG (Luo et al. 2023d) uses a planning retrieval reasoning framework to combine LLMs with KG, enabling faithful and explainable reasoning. ToG (Sun et al. 2023a) leverages LLMs to iteratively explore reasoning paths on the KG until the question can be answered based on the current path. NSM (He et al. 2021b) uses a sequential model to replicate the multi-hop reasoning process. KD-CoT (Wang et al. 2023) retrieves relevant knowledge from KG to generate faithful reasoning paths for LLMs. DECAF (Yu et al. 2022a) combines LF and retrieved paths to generate answers using LLMs, achieving strong performance in KBQA tasks. FiDeLiS (Sui et al. 2024) proposes a retrieval exploration method that incorporates both the logical and common sense reasoning of LLMs and the topological connectivity of KG into KBQA.

Method	WebQSP		CWQ	
	F1	Hits@1	F1	Hits@1
Non-LLMs methods				
Rigel (Sen, Saffari, and Oliya 2021)	-	73.3	-	48.7
TIARA (Shu et al. 2022b)	78.9	75.2	-	-
UniK-QA (Oguz et al. 2020b)	79.1	-	-	-
UniKGQA (Jiang et al. 2022b)	72.2	77.2	49.4	51.2
HGNet (Chen et al. 2022)	76.6	76.9	68.5	68.9
Prompting-LLMs Only method				
Zero-shot(gpt-4)	59.71	62.32	37.93	42.71
Few-shot(gpt-4)	62.71	68.75	43.70	51.52
CoT(gpt-4)	65.37	72.11	44.76	53.51
Prompting- LLMs + KG				
ToG(gpt-3.5)	72.32	75,13	56.96	57.59
ToG(gpt-4) (Sun et al. 2023a)	75.97	81.84	60.20	68.51
FiDeLiS(gpt-3.5) (Sui et al. 2024)	76.78	79.32	63.12	61.78
FiDeLiS(gpt-4)	78.32	84.39	64.32	71.47
Finetuning- LLMs + KG				
NSM (He et al. 2021b)	-	74.31	-	53.92
DeCAF (Yu et al. 2022b)	-	82.1	-	70.42
KD-CoT (Wang et al. 2023)	50.2	73.7	-	50.5
RoG (Luo et al. 2023d)	69.81	83.15	56.17	61.39
FM-KBQA (Ours)	**84.24**	**87.34**	**68.68**	**79.50**

Table 1: Comparison results with baseline methods

Evaluation metrics. Following prior research (Yu et al. 2022b; Luo et al. 2023d; Jiang et al. 2022b), we use the F1 score and Hits@1 metric to represent the overall coverage of the answers and the top-ranked single answer, respectively.

Models. (1) Backboned PLMs. For candidate reasoning path retrieval, we leverage T5 (Raffel et al. 2020b) to preserve relevant relations while filtering out irrelevant relations for each given question. (2) Backbone LLMs. LF generation and path reasoning are performed using open-source LLMs. We adopt Llama-2-7B (Luo et al. 2023b) for fine-tuning and evaluation on the two different datasets.

Hyperparameters and environment. For the PLMs, we train T5 100 epochs on both datasets with a batch size 4 and a learning rate 2e-5. For LLMs, Llama-2 is fine-tuned with a batch size of 4 and a learning rate of 5e-5. The beam size is set to 5 during beam search in the evaluation process. All experiments are conducted on a single NVIDIA A40 GPU.

Overall results. To verify the effectiveness of the pro-

Model	WQSP task		CWQ Task	
	F1	Hits@1	F1	Hits@1
Only LF (Executable)	84.12	90.2	79.21	88.8
Non-executable LF%	8.7		33.08	
FM-KBQA (Only LF)	77.29	82.75	53.01	59.43
FM-KBQA (Only RP)	79.96	83.92	55.16	68.88
FM-KBQA (LF+RP)	**84.24**	**87.74**	**68.68**	**79.50**

Table 2: Ablation result of FM-KBQA on WQSP and CWQ dataset. Only LF means only use logical forms, only RP means only use reasoning path.

Model	WQSP task	
	F1	Hits@1
LF-KBQA (Executable)	68.28	77.7
Non-executable LF%	14.39	
FM-KBQA (Only LF-executable)	**84.12**	**90.2**
Non-executable LF%	8.7	

Table 3: Task-related logical forms analysis on the WQSP dataset. We further use the question and the retrieved path as input and fine-tune the LLMs (Llama-2-7B) to generate only LF, forming the comparative method LF-KBQA.

posed method, we compare our method with four types of methods, namely, non-LLMs methods, prompting-LLMs Only, prompting-LLMs + KG, and Finetuning-LLMs + KG. The corresponding F1 and Hits@1 are shown in Table 1.

First, FM-KBQA significantly outperforms the baselines across all metrics on the two public datasets. When using the F1 as the evaluation metric, the improvements over the state-of-the-art baseline, FiDeLiS (gpt-4.0), are 5.92% and 4.36% on the WQSP and CWQ datasets, respectively. When using the Hits@1 as the evaluation metric, the improvements are 2.95% and 8.03%, respectively.

Among the baselines, non-LLMs methods simulate the reasoning process of KG using specific neural networks (e.g., graph neural networks). However, the reasoning capabilities of non-LLMs models are limited, and their performance needs to be further improved. Prompting-LLMs only methods perform poorly because the LLMs lack knowledge from the KG. For prompting-LLMs + KG based methods, ToG and FiDeLiS first retrieve relational paths from the KG and then use these paths to perform logical reasoning with open-source LLMs (e.g., GPT-3.5 and GPT-4), achieving remarkable results. FiDeLiS takes advantage of the LLMs' ability to perform deductive reasoning, thus achieving state-of-the-art performance in this kind of method. However, too many retrieval paths may introduce noise, resulting in limited improvement in model performance. For finetuning-LLMs + KG, DeCAF and RoG are two strong baselines. Among them, DeCAF combines semantic parsing and information retrieval to directly generate answers. However, directly generating answers may cause LLMs to lack clear and orderly thinking for the given question and context. RoG employs a planning retrieval reasoning framework to combine LLMs with KG to achieve faithful and explainable reasoning. However, this approach struggles with noise caused by path retrieval. In contrast, our method generates task-specific LF and selects the correct path to effectively make up for the shortcomings of some generated LF that cannot be executed. As a result, our method achieves the state-of-the-art performance.

Ablation analysis. To validate the effectiveness of each part in FM-KBQA, we conduct ablation experiments by systematically removing each module, as shown in Table 2. Only LF means only use logical forms, only RP means only use reasoning path. It can be seen that for executable LF generated by the LLMs, the answers obtained by executing the LF have high accuracy. This also verifies the conclusions of some previous research (Ye et al. 2021; Das et al. 2021). However, the LF generated by the LLMs also have the problem of being partially non-executable. For example, in the WQSP task, the proportion of non-executable LF is 8.7%, and in the more complex CWQ task, it reaches 33.08%. The problem of non-executable generated LF seriously limits the performance of this type of method.

Moreover, in the reasoning stage, it is evident that the performance of the FM-KBQA method that only uses LF has degraded. This is because some non-executable LF fail to retrieve answers, thereby reducing the overall performance of FM-KBQA (Only LF). However, it can be seen that our method achieves the best performance after combining the LF and RP strategies. This is because (1) we design a multi-task learning approach to enable the LLMs to generate task-related LF (see Table 3), and (2) we use deductive reasoning to enable LLMs to select effective reasoning paths. The answers obtained from these reasoning paths effectively make up for the problems caused by non-executable LF.

Task-relevant logical forms analysis. To verify that the auxiliary task path selection module can help the LLMs generate task-related LF, we conduct the experiments shown in Table 3. It can be seen that when there is no auxiliary task path selection, the performance of LF-KBQA with executable LF drops by 15.84% and 12.5% in F1 and Hits@1 respectively, while the proportion of non-executable LF increases by 5.69%. This suggests that too many retrieval paths may introduce unnecessary noise paths, which may confuse the LLMs during the generation of LF, resulting in a serious degradation in the quality of the generated LF. Our method addresses this issue by introducing auxiliary tasks to guide the LLMs in generating task-related LF, leading to state-of-the-art performance.

T5 retriever performance analysis. To validate the effectiveness of the proposed enhanced T5 retriever, we compared our method with standard KBQA retrievers such as BM25, DPR, and Sentence-BERT. As shown in Figure 2, the accuracy of our method significantly outperforms the aforementioned methods, indicating that our approach can provide accurate contextual information needed for the reasoning stage of LLMs with fewer path searches. This approach not only reduces the impact of excessive path retrieval on the reasoning performance of LLMs but also decreases computational resource consumption.

Hyperparameter beam size selection. To choose an ap-

Question	What was Dr Seuss education?
Answer	Dartmouth College; University of Oxford; Lincoln College, Oxford
GPT+CoT	Think step by step. Given a question: (What was Dr Seuss education?) and a series of candidate triplets: 0: [Theodore Lesieg, people.person.education, m.0n1kynd], [m.0n1kynd, education.education.institution, University of Oxford] 1: [Theodore Lesieg, people.person.education, m.04ytk85], [m.04ytk85, education.education.institution, Dartmouth College] 2: [Theodore Lesieg, people.person.education, m.03p87mv], [m.03p87mv, education.education.institution, Lincoln College, Oxford] First, select the related triplets according to the given question. Then, retrieve the answer from the reasoning path. **Answer: Dartmouth College** in **triplet 1**.
proposed	Given a question: (What was Dr Seuss education?) and a series of reasoning paths: 0: The premise is: Theodore Lesieg->people.person.education->m.0n1kynd->education.education.institution->University of Oxford. Its conclusion is: Dr. Seuss's education was University of Oxford. 1: The premise is: Theodore Lesieg->people.person.education->m.04ytk85->education.education.institution->Dartmouth College. Its conclusion is: Dr. Seuss's education was Dartmouth College. 2: The premise is: Theodore Lesieg->people.person.education->m.03p87mv->education.education.institution->Lincoln College, Oxford. Its conclusion is: Dr. Seuss's education was Lincoln College, Oxford. Task 1: Generate logical forms just based on the question. Task 2: Verify which deductive reasoning is correct for the given question in a deductive manner, if it exits, return the correct number of deductive reasoning, if doesn't, return "no". **Output:** Logical form: **(JOIN (R [education , education , institution]) (JOIN (R [people , person , education]) [Theodore Lesieg]))))** Searched answer: **Dartmouth College; Lincoln College, Oxford; University of Oxford.** Path selection: **deductive reasoning 0, 1, 2.** Extracted answer: **University of Oxford; Dartmouth College; Lincoln College, Oxford.**

Table 4: Case study of ChatGPT-CoT and FM-KBQA

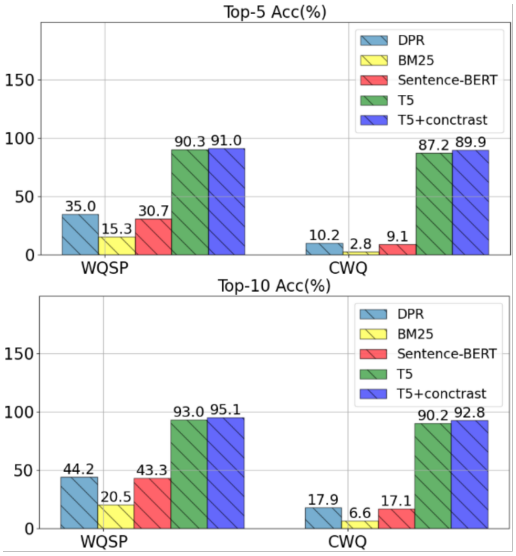

Figure 2: The accuracy of different retrieval methods on the WQSP and CWQ datasets. Top-5 Acc means that all gold relations are in the top 5 predicted by the models.

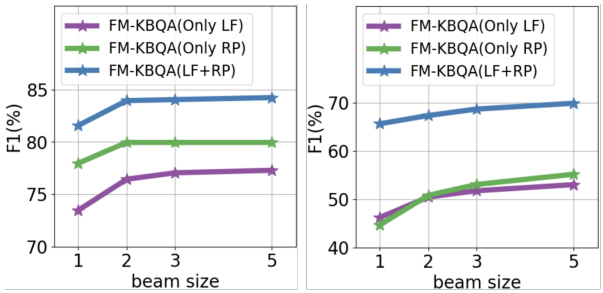

Figure 3: The F1 score under different beam size on WQSP (left) and CWQ (right) datasets. LF means the logical forms, RP represents reasoning path, and "LF+RP" denotes our method FM-KBQA.

propriate beam size, we conduct the experiment shown in Figure 3. The results reveal that as the beam size increases, the performance of our method gradually improves. When the beam size reaches 2, the model's performance begins to stabilize. This suggests that the model has identified the necessary answers with a beam size of 2, and demonstrates that the proposed method is not only capable of reducing computational consumption but also exhibits strong effectiveness and robustness.

Case study. We also present the case study in the Table 4, it can find that GPT+COT only correctly selects one reasoning path for the multi-answer question. In contrast, our method not only generates the correct LF but also selects the correct reasoning path through deductive reasoning. This suggests that LLMs may be better at reasoning with logical knowledge through deductive methods, while their performance in graph-based reasoning is limited. Furthermore, in our method, the LF and path selection can complement each other to generate more accurate answers.

Conclusion

In this paper, we introduce a novel learning framework, FM-KBQA, which leverages multi-task learning to fine-tune LLMs to generate task-relevant LF. Specifically, we design an additional objective function that instructs the LLMs to output the index of answer-relevant reasoning paths through deductive reasoning. This enables the LLMs to focus more on answer-relevant paths and guides them in generating task-related LF. Furthermore, the reasoning paths can serve as an effective supplement to the question's answer when the LF are non-executable. This approach significantly enhances the performance of LLMs in KBQA.

Acknowledgements

This work is supported by the Interdisciplinary Program of Shanghai Jiao Tong University (project number YG2024QNB05). This work is also supported by the collaborative research founding of Voicecomm Technology Co. Ltd..

References

Bollacker, K. 2008. Freebase : A collaboratively created graph database for structuring human knowledge. *Proc. SIGMOD' 08*.

Chen, S.; Liu, Q.; Yu, Z.; Lin, C.-Y.; Lou, J.-G.; and Jiang, F. 2021. ReTraCk: A flexible and efficient framework for knowledge base question answering. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing: system demonstrations*, 325–336.

Chen, Y.; Li, H.; Qi, G.; Wu, T.; and Wang, T. 2022. Outlining and filling: hierarchical query graph generation for answering complex questions over knowledge graphs. *IEEE Transactions on Knowledge and Data Engineering*, 35(8): 8343–8357.

Das, R.; Zaheer, M.; Thai, D.; Godbole, A.; Perez, E.; Lee, J.-Y.; Tan, L.; Polymenakos, L.; and McCallum, A. 2021. Case-based reasoning for natural language queries over knowledge bases. *arXiv preprint arXiv:2104.08762*.

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 *Proceedings of the Web Conference 2021*, 3477–3488.

Gu, Y.; Pahuja, V.; Cheng, G.; and Su, Y. 2022. Knowledge base question answering: A semantic parsing perspective. *arXiv preprint arXiv:2209.04994*.

Gu, Y.; and Su, Y. 2022. ArcaneQA: Dynamic program induction and contextualized encoding for knowledge base question answering. *arXiv preprint arXiv:2204.08109*.

He, G.; Lan, Y.; Jiang, J.; Zhao, W. X.; and Wen, J.-R. 2021a. Improving multi-hop knowledge base question answering by learning intermediate supervision signals. In *Proceedings of the 14th ACM international conference on web search and data mining*, 553–561.

He, G.; Lan, Y.; Jiang, J.; Zhao, W. X.; and Wen, J.-R. 2021b. Improving multi-hop knowledge base question answering by learning intermediate supervision signals. In *Proceedings of the 14th ACM international conference on web search and data mining*, 553–561.

Huang, X.; Kim, J. J.; and Zou, B. 2021. Unseen entity handling in complex question answering over knowledge base via language generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, 547–557.

Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Jiang, J.; Zhou, K.; Zhao, W. X.; and Wen, J.-R. 2022a. Great truths are always simple: A rather simple knowledge encoder for enhancing the commonsense reasoning capacity of pre-trained models. *arXiv preprint arXiv:2205.01841*.

Jiang, J.; Zhou, K.; Zhao, W. X.; and Wen, J.-R. 2022b. Unikgqa: Unified retrieval and reasoning for solving multi-hop question answering over knowledge graph. *arXiv preprint arXiv:2212.00959*.

Karpukhin, V.; Oğuz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; and Yih, W.-t. 2020a. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Karpukhin, V.; Oguz, B.; Min, S.; Wu, L.; Edunov, S.; Chen, D.; and Yih, W. 2020b. Dense Passage Retrieval for Open-Domain Question Answering. *CoRR*, abs/2004.04906.

Lan, Y.; Wang, S.; and Jiang, J. 2019. Knowledge base question answering with topic units.

Ling, Z.; Fang, Y.; Li, X.; Huang, Z.; Lee, M.; Memisevic, R.; and Su, H. 2024. Deductive verification of chain-of-thought reasoning. *Advances in Neural Information Processing Systems*, 36.

Lu, Y.; Liu, J.; Zhang, Y.; Liu, Y.; and Tian, X. 2022. Prompt distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5206–5215.

Luo, H.; E, H.; Tang, Z.; Peng, S.; Guo, Y.; Zhang, W.; Ma, C.; Dong, G.; Song, M.; and Lin, W. 2023a. ChatKBQA: A Generate-then-Retrieve Framework for Knowledge Base Question Answering with Fine-tuned Large Language Models. *CoRR*, abs/2310.08975.

Luo, H.; Tang, Z.; Peng, S.; Guo, Y.; Zhang, W.; Ma, C.; Dong, G.; Song, M.; Lin, W.; et al. 2023b. Chatkbqa: A generate-then-retrieve framework for knowledge base question answering with fine-tuned large language models. *arXiv preprint arXiv:2310.08975*.

Luo, L.; Li, Y.-F.; Haffari, G.; and Pan, S. 2023c. Reasoning on graphs: Faithful and interpretable large language model reasoning. *arXiv preprint arXiv:2310.01061*.

Luo, L.; Li, Y.-F.; Haffari, G.; and Pan, S. 2023d. Reasoning on graphs: Faithful and interpretable large language model reasoning. *arXiv preprint arXiv:2310.01061*.

Miller, A.; Fisch, A.; Dodge, J.; Karimi, A.-H.; Bordes, A.; and Weston, J. 2016. Key-value memory networks for directly reading documents. *arXiv preprint arXiv:1606.03126*.

Oguz, B.; Chen, X.; Karpukhin, V.; Peshterliev, S.; Okhonko, D.; Schlichtkrull, M.; Gupta, S.; Mehdad, Y.; and Yih, S. 2020a. Unik-qa: Unified representations of structured and unstructured knowledge for open-domain question answering. *arXiv preprint arXiv:2012.14610*.

Oguz, B.; Chen, X.; Karpukhin, V.; Peshterliev, S.; Okhonko, D.; Schlichtkrull, M.; Gupta, S.; Mehdad, Y.; and Yih, S. 2020b. Unik-qa: Unified representations of structured and unstructured knowledge for open-domain question answering. *arXiv preprint arXiv:2012.14610*.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020a. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020b. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. (140).

Robertson, S.; Zaragoza, H.; et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4): 333–389.

Saxena, A.; Kochsiek, A.; and Gemulla, R. 2022. Sequence-to-sequence knowledge graph completion and question answering. *arXiv preprint arXiv:2203.10321*.

Sen, P.; Saffari, A.; and Oliya, A. 2021. Expanding end-to-end question answering on differentiable knowledge graphs with intersection. *arXiv preprint arXiv:2109.05808*.

Sener, O.; and Koltun, V. 2018. Multi-Task Learning as Multi-Objective Optimization. In Bengio, S.; Wallach, H. M.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 525–536.

Shu, Y.; Yu, Z.; Li, Y.; Karlsson, B. F.; Ma, T.; Qu, Y.; and Lin, C.-Y. 2022a. Tiara: Multi-grained retrieval for robust question answering over large knowledge bases. *arXiv preprint arXiv:2210.12925*.

Shu, Y.; Yu, Z.; Li, Y.; Karlsson, B. F.; Ma, T.; Qu, Y.; and Lin, C.-Y. 2022b. Tiara: Multi-grained retrieval for robust question answering over large knowledge bases. *arXiv preprint arXiv:2210.12925*.

Sui, Y.; He, Y.; Liu, N.; He, X.; Wang, K.; and Hooi, B. 2024. FiDeLiS: Faithful Reasoning in Large Language Model for Knowledge Graph Question Answering. *arXiv preprint arXiv:2405.13873*.

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. W. 2018. Open domain question answering using early fusion of knowledge bases and text. *arXiv preprint arXiv:1809.00782*.

Sun, J.; Xu, C.; Tang, L.; Wang, S.; Lin, C.; Gong, Y.; Shum, H.-Y.; and Guo, J. 2023a. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697*.

Sun, J.; Xu, C.; Tang, L.; Wang, S.; Lin, C.; Gong, Y.; Shum, H.-Y.; and Guo, J. 2023b. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697*.

Sun, Y.; Zhang, L.; Cheng, G.; and Qu, Y. 2020. SPARQA: skeleton-based semantic parsing for complex questions over knowledge bases. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 8952–8959.

Talmor, A.; and Berant, J. 2018a. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*.

Talmor, A.; and Berant, J. 2018b. The Web as a Knowledge-base for Answering Complex Questions.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, K.; Duan, F.; Wang, S.; Li, P.; Xian, Y.; Yin, C.; Rong, W.; and Xiong, Z. 2023. Knowledge-driven cot: Exploring faithful reasoning in llms for knowledge-intensive question answering. *arXiv preprint arXiv:2308.13259*.

Ye, X.; Yavuz, S.; Hashimoto, K.; Zhou, Y.; and Xiong, C. 2021. Rng-kbqa: Generation augmented iterative ranking for knowledge base question answering. *arXiv preprint arXiv:2109.08678*.

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.

Yu, D.; Zhang, S.; Ng, P.; Zhu, H.; Li, A. H.; Wang, J.; Hu, Y.; Wang, W.; Wang, Z.; and Xiang, B. 2022a. Decaf: Joint decoding of answers and logical forms for question answering over knowledge bases. *arXiv preprint arXiv:2210.00063*.

Yu, D.; Zhang, S.; Ng, P.; Zhu, H.; Li, A. H.; Wang, J.; Hu, Y.; Wang, W.; Wang, Z.; and Xiang, B. 2022b. Decaf: Joint decoding of answers and logical forms for question answering over knowledge bases. *arXiv preprint arXiv:2210.00063*.

Zhang, Y.; Gong, M.; Liu, T.; Niu, G.; Tian, X.; Han, B.; Schölkopf, B.; and Zhang, K. 2021. Causaladv: Adversarial robustness through the lens of causality. *arXiv preprint arXiv:2106.06196*.