KNOWLEDGE GRAPH BASED AGENT FOR COMPLEX, KNOWLEDGE-INTENSIVE QA IN MEDICINE

Anonymous authors
Paper under double-blind review

ABSTRACT

Biomedical knowledge is uniquely complex and structured, requiring distinct reasoning strategies compared to other scientific disciplines like physics or chemistry. Biomedical scientists do not rely on a single approach to reasoning; instead, they use various strategies, including rule-based, prototype-based, and case-based reasoning. This diversity calls for flexible approaches that accommodate multiple reasoning strategies while leveraging in-domain knowledge. We introduce KGAREVION, a knowledge graph (KG) based agent designed to address the complexity of knowledge-intensive medical queries. Upon receiving a query, KGAREVION generates relevant triplets by using the knowledge base of the LLM. These triplets are then verified against a grounded KG to filter out erroneous information and ensure that only accurate, relevant data contribute to the final answer. Unlike RAG-based models, this multi-step process ensures robustness in reasoning while adapting to different models of medical reasoning. Evaluations on four gold-standard medical QA datasets show that KGAREVION improves accuracy by over 5.2%, outperforming 15 models in handling complex medical questions. To test its capabilities, we curated three new medical QA datasets with varying levels of semantic complexity, where KGAREVION achieved a 10.4% improvement in accuracy. The source code is provided at `https://anonymous.4open.science/r/KGARevion-B3B6`.

1 INTRODUCTION

Medical reasoning involves making diagnostic and therapeutic decisions while also understanding the pathology of diseases (Patel et al., 2005). Unlike many other scientific domains, medical reasoning often relies on vertical reasoning, using analogy more heavily (Patel et al., 2005). For instance, in biomedical research, an organism such as Drosophila is used as an exemplar to model a disease mechanism, which is then applied by analogy to other organisms, including humans. In clinical practice, the patient serves as an exemplar, with generalizations drawn from many overlapping disease models and similar patient populations (Charles et al., 1997; Menche et al., 2015). In contrast, fields like physics and chemistry tend to be horizontally organized, where general principles are applied to specific cases (Blois, 1988). This distinction highlights the unique challenges that medical reasoning poses for question-answering (QA) models.

While large language models (LLMs) (OpenAI, 2024; Dubey et al., 2024; Gao et al., 2024) have demonstrated strong general capabilities, their responses to medical questions often suffer from incorrect retrieval, missing key information, and misalignment with current scientific and medical knowledge. Additionally, they can struggle to provide contextually relevant answers that account for specific local contexts, such as patient demographics or geography, as well as specific areas of biology (Harris, 2023). A major issue lies in these models' inability to systematically integrate different types of evidence. Specifically, they have difficulty combining scientific factual (*structured, codified*) knowledge derived from formal, rigorous research with tacit (*noncodified*) knowledge—expertise and lessons learned—which is crucial for contextualizing and interpreting scientific evidence in relation to the specific modifying factors of a given medical question (Harris, 2023).

LLM-powered QA models often lack such *multi-source* and *grounded knowledge* necessary for medical reasoning, which requires understanding the nuanced and specialized nature of medical concepts. Additionally, LLMs trained on general knowledge may struggle to solve medical prob-

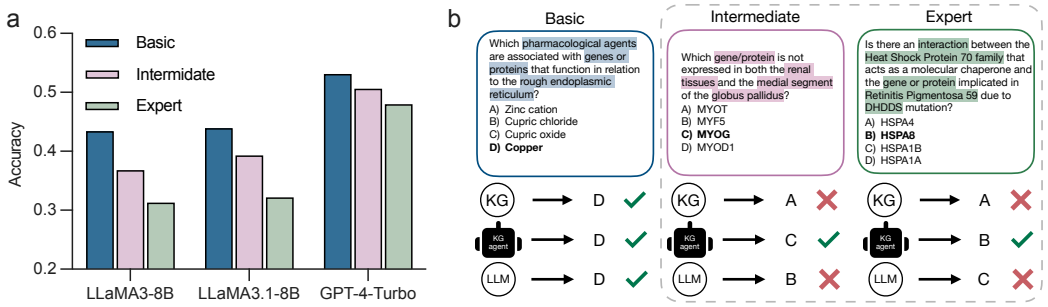

Figure 1: **a)** Performance of LLMs on three new datasets introduced in this paper with questions of varying difficulty. **b)** Sample questions from new MedDDx-Basic, MedDDx-Intermediate, and MedDDx-Expert datasets.

lems that demand *specialized in-domain knowledge*. This shortcoming arises from their inability to discern subtle, granular differences that are critical in medical contexts. As a result, LLMs face challenges in complex medical reasoning because such reasoning requires both: 1) simultaneous consideration of dependencies across multiple medical concepts within an input question, and 2) precise, local in-domain knowledge of semantically similar concepts that can carry different medical meanings, as we demonstrate in Fig. 1.

The prevailing strategy to address these challenges is the use of information retrieval techniques, such as retrieval-augmented generation (RAG), which follows a Retrieve-then-Answer paradigm (Xiong et al., 2024b; Shi et al., 2024). Although these methods can provide multi-source knowledge from external databases (Fan et al., 2024), the accuracy of the generated answers depends heavily on the quality of the retrieved information, making them vulnerable to potential errors (Karpukhin et al., 2020). Data repositories and knowledge bases these models draw from contain incomplete or incorrect information, leading to inaccurate retrieval Adlakha et al. (2024); Thakur et al. (2023). Further, many RAG-based methods lack post-retrieval verification mechanisms to validate that retrieved information is factually correct and does not miss key information (Zhao et al., 2023). Knowledge graphs (KGs) of medical concepts have been widely adopted as a grounded knowledge base to provide precise and specialized in-domain knowledge for medical QA models (Qi et al., 2024; Murali et al., 2023; Chandak et al., 2023; Jiang et al.). While KGs can enhance the performance of these models, they are often incomplete. Consequently, approaches that retrieve medical concepts from a KG based solely on the presence of direct associations (edges) between concepts are insufficient. For instance, concepts representing two proteins with distinct biological roles may not be directly connected in the KG, even though these proteins share similar biological representations (Menche et al., 2015). To advance LLM-powered models for knowledge-intensive medical QA, it is essential to develop models that can (1) consider complex associations between several medical concepts at the same time, (2) systematically integrate multi-source knowledge, and (3) effectively verify and ground the retrieved information to ensure contextual relevance and accuracy.

Present work. We introduce KGAREVION, a KG-based LLM agent for complex medical QA that leverages non-codified knowledge of LLMs and structured, codified knowledge of medical concepts within KGs. KGAREVION operates through four key actions, Fig. 2. First, KGAREVION prompts the LLM to generate relevant triplets based on the input question. To ensure the accuracy of the generated triples and to fully use structured KG, KGAREVION fine-tunes the LLM on a KG completion task by incorporating pre-trained structural embeddings of triplets as prefix tokens. The fine-tuned model is then used to evaluate the correctness of generated triplets. Following this, KGAREVION executes a 'Revise' action to correct any erroneous triplets, ultimately identifying the correct answer based on the verified triplets. Given the complexity of medical reasoning, KGAREVION agent adaptively selects the most appropriate reasoning approach for each input question, enabling a more nuanced and contextualized QA. This flexibility allows KGAREVION to tackle both *multi-choice* and *open-ended* QA. Our key contributions of KGAREVION are: ① We develop KGAREVION, a versatile KG-agent that dynamically adjusts reasoning strategy, leading to 6.75% improvement when compared to 15 models on seven datasets, including three new challenging datasets; ② Results on multiple KGs show that grounding through generated triplets can improve KGAREVION 's capabilities; ③ Results on both multiple-choice and open-ended setups show that KGAREVION can effectively handle complex, knowledge-intensive medical QA.

2 RELATED WORK

LLM-based reasoning. General-purpose LLMs (GPT (OpenAI, 2024), LLaMA family (Dubey et al., 2024; Touvron et al., 2023), Mistral (Jiang et al., 2023)), and LLMs fine-tuned on biomedical data (BioMedLM (Venigalla et al., 2022), Codex (Liévin et al., 2024), MedAlpaca (Han et al., 2023), Med-PaLM (Singhal et al., 2023), PMC-LLaMA (Wu et al., 2024a)) are used for medical reasoning by leveraging their vast embedded knowledge. Other models utilize the open-ended reasoning capabilities of LLMs to break down queries into sub-tasks, arriving at the final answer step by step, such as Chain-of-Though (CoT) (Wei et al., 2024), CODEX COT (Gramopadhye et al., 2024). However, these methods often struggle with knowledge-intensive medical queries requiring multi-sources an specific knowledge.

RAG-based models. Self-RAG (Asai et al., 2024) is a pioneering framework that enhances LLM performance through retrieval and self-reflection. LLM-AMT (Wang et al., 2023b) improves medical question answering by integrating authoritative medical textbooks into large language models with specialized knowledge retrieval and self-refinement techniques. Adaptive-RAG (Jeong et al., 2024) introduces a dynamic RAG framework that adapts retrieval strategies based on question complexity. However, its accuracy is constrained by the quality of retrieved knowledge (Zhang et al., 2024).

KG-based models. Before the rise of LLMs, several models, such as QAGNN (Yasunaga et al., 2021), JointLK (Sun et al., 2022), and Dragon (Yasunaga et al., 2022), were developed to tackle medical queries solely using KGs in an end-to-end manner. However, these methods cannot be easily applied to questions involving unseen nodes or incomplete knowledge within the graphs. In addition, KGs, with their structured and reliable information, have driven research toward RAG models based on graph data, motivating models like GraphRAG (Edge et al., 2024), KG-RAG (Soman et al., 2023), and MedGraphRAG (Wu et al., 2024b). To improve retrieval accuracy, KG-Rank (Yang et al., 2024) is introduced to rank retrieved triplets and filter out irrelevant knowledge. Additionally, Gen-Ground (Shi et al., 2024) uses a Generate-then-Ground pipeline that grounds answers by prompting LLMs to validate retrieved knowledge. However, all these approaches rely heavily on semantic dependencies, overlooking the rich structural information within KGs.

3 APPROACH

Given is a set of medical questions Q, each question comprising the question stem q, and a set of candidate answers C. For example, the sample question in Fig. 1b has a stem q = "*Is there an interaction between the Heat Shock Protein 70 family that acts as a molecular chaperone and the gene or protein implicated in Retinitis Pigmentosa 59 due to DHDDS mutation?*" along with a set of semantically related candidate answers $C = \{\text{HSPA4}, \text{HSPA8}, \text{HSPA1B}, \text{HSPA1A}\}$. The goal is to identify the correct answer $a \in C$ using an LLM (denoted as P) and a KG (denoted as G). Here, a KG is given as a set of triplets $G = \{(h, r, t)\}$, where each triplet consists of a head entity, a relationship, and a tail entity. Full notation is listed in Table E.2. Note that in addition to this multi-choice setting, we consider open-ended reasoning as well (see Results).

To address this problem, we propose developing an LLM-powered agent framework (Wu et al., 2023; Li et al., 2023) that leverages various actions (Schick et al., 2023; Shen et al., 2023; Nakano et al., 2021) to collaboratively perform complex tasks (Tang et al., 2023; Bran et al., 2023; Boiko et al., 2023). Fig. 2 shows an overview of KGAREVION, which comprises four key actions, including Generate (§3.1), Review (§3.2), Revise (§3.3), and Answer (§3.3) actions. The Generate action is responsible for generating triplets related to the input question. The Review action then assesses the correctness of each generated triplet, while the Revise action corrects any triplet identified as being incorrect. Finally, the Answer action outputs the final answer based on the triplets identified as correct by the Review action.

3.1 GENERATE ACTION

The Generate action aims to gather comprehensive structured knowledge from input questions. Specifically, this action first identifies all medical concepts involved in the input question stem q and then generates a set of triplets T related to the question based on the extracted medical concepts.

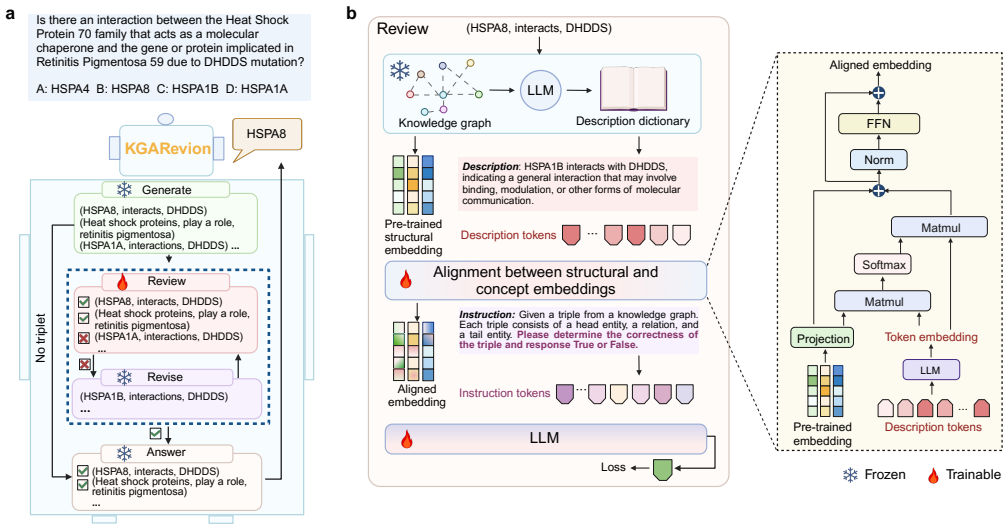

Figure 2: **a)** The overview of KGAREVION. **b)** The architecture of fine-tuning stage in the Review action, where embeddings get from KGs are structural embeddings, while concept embeddings from LLMs.

Depending on the content of each answer candidate $a \in C$, the input questions can be broadly categorized into two types: choice-aware and non-choice-aware. The answer candidates in the choice-aware group have specific contents, whereas the ones in the non-choice-aware group only contain yes-or-no options. These different types of questions require distinct reasoning processes: choice-aware questions involve analyzing the content of each answer candidate, while non-choice-aware questions only require focusing on the question stem.

To handle this, the Generate action is designed to prompt the LLM (Ouyang et al., 2022; Wang et al., 2023a) to follow different procedures for generating relevant triplets according to the type of input question.

- For choice-aware questions, the Generate action generates triplets based on the contents of each answer candidate and extracted medical concepts in question stem q;
- For non-choice-aware questions, the Generation action directly generates triplets based on medical concepts presented in question stem q.

The rationale behind this design is that LLMs have inherent biases in their knowledge, often generating more detailed information on familiar topics compared to less familiar ones when all answer candidates are presented simultaneously (Dai et al., 2024). Additionally, this approach helps reduce the impact of the order in which the answer candidates are presented. The process of the Generate action can be formulated as:

$$T = \begin{cases} \{P(q, a_i)\}, & 1 \leq i \leq |C|, \quad \text{if } C \not\subseteq \{\text{Yes}, \text{No}, \text{Maybe}\} \\ P(q), & \text{if } C \subseteq \{\text{Yes}, \text{No}, \text{Maybe}\} \end{cases} \tag{1}$$

where the LLM is prompted to extracts triplets from the medical concepts involved in input question stem q.

3.2 REVIEW ACTION

To enable LLMs to accurately judge the correctness of generated triplets, beyond relying solely on semantic dependencies inferred by LLMs (Shinn et al., 2023), the Review action also leverages the relationships among various medical concepts contained in KGs. This is achieved by fine-tuning the LLM on a KG completion task, explicitly integrating entity structural embeddings learned from KGs into the LLM. Once fine-tuned, the Review action utilizes the model to assess the correctness of generated triplets, as shown in Fig. 2.

To enable the LLM to capture the structural information embedded in KGs, we employ TransE (Bordes et al., 2013) to learn structural embeddings for both entities and relations in G. These obtained embeddings are fixed when fine-tuning the LLM on knowledge completion tasks.

Fine-tuning stage. Given a triplet $(h, r, t) \in G$ and its pre-trained embedding $\mathbf{e}_h \in \mathbb{R}^d$, $\mathbf{e}_r \in \mathbb{R}^d$, and $\mathbf{e}_t \in \mathbb{R}^d$, where d is the dimension of the embedding, the aim of this stage is to learn a function $f(\cdot)$ that makes the LLM capable of determining if a given triplet is True or False by considering its structural embeddings in the KG, as follows:

$$b = f(P(\cdot), (h, r, t), \mathbf{e}_h, \mathbf{e}_r, \mathbf{e}_t),\ b \in \{\text{True}, \text{False}\} \tag{2}$$

where b is a bool value (True or False) and $P(\cdot)$ denotes the LLM.

However, it is complex for LLMs to directly understand structural embeddings given that the embeddings in LLMs are obtained based on their vocabularies (Radford et al., 2019). To make sure that LLMs can understand structural embeddings, we generate a description for each triplet based on the relationship between the two entities. Then, we tokenize the generated descriptions (Touvron et al., 2023) and obtain the corresponding token embeddings. The token embeddings are aligned with the pre-trained structural embeddings of the triplets to produce new embeddings for each input triplet.

In detail, given the input triplet (h, r, t), we denote its description as $D(r)$, where D represents the description dictionary with the key as relation and the value as a pre-defined description template, as shown in Appendix 8. Assuming the token embedding of $D(r)$ obtained from LLM $P(\cdot)$ is $\mathbf{X} \in \mathbb{R}^{|l| \times d_p}$, where $|l|$ denotes the number of max tokens and d_p denotes the dimension of the embeddings in $P(\cdot)$, we first adopt a linear layer ($g(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^{d_p}$) to map the dimension of pre-trained embeddings of entities and relations to the same dimension of that of token embeddings. Then, we concatenate them to create the triplet embedding matrix $\mathbf{V} = [g(\mathbf{e}_h); g(\mathbf{e}_r); g(\mathbf{e})_t)] \in \mathbb{R}^{3 \times d_p}$. Afterwards, we adopt an attention block (Vaswani, 2017), followed by a two-layer feedforward neural network (denoted as FFN in Fig. 2b) to obtain the aligned triplet embedding matrix $\mathbf{Z} \in \mathbb{R}^{3 \times d_p}$, as follows:

$$\widehat{\mathbf{V}} = \mathbf{V} + \sigma(\mathbf{V}\mathbf{X}^T)\mathbf{X} \tag{3}$$

$$\mathbf{Z} = \widehat{\mathbf{V}} + ((\varphi(\widehat{\mathbf{V}})\mathbf{W}_1))\mathbf{W}_2 \tag{4}$$

where $\sigma(\cdot)$ denotes the $\mathrm{Softmax}$ function, $\varphi(\cdot)$ denotes the layer normalization function,$\mathbf{W}_1 \in \mathbb{R}^{d_p \times d_h}$ and $\mathbf{W_2} \in \mathbb{R}^{d_h \times d_p}$ denotes the trainable parameters in the two-layer FFN, and d_h denotes the dimension of hidden layer in the FFN.

The obtained aligned triplet embedding matrix is input into the LLM in the form of three prefix tokens together with an instruction s to fine-tune the LLM to execute the knowledge graph completion task. Then, the Eq. 2 could be rewritten by:

$$b = f(P(\cdot), D(r), g(\mathbf{e}_h), g(\mathbf{e}_r), g(\mathbf{e}_t)) = f(P(\cdot), \mathbf{X}, \mathbf{V}) = f(P(\mathbf{Z}, s)) \tag{5}$$

During the fine-tuning stage, the pre-trained entity and relation embeddings are frozen, and LoRA (Hu et al., 2022) is adopted to fine-tune the LLM. The trainable parameters are optimized using the next token prediction loss (Radford, 2018).

Inference stage. After fine-tuning, the model is used to evaluate the triplets in T derived from the Generate action (3.1). In detail, we first adopt UMLS code (Bodenreider, 2004) to map entities in each triplet $(h, r, t) \in T$ to KGs to get the corresponding pre-trained entity and relation embeddings, denoted as $\mathbf{e}_h, \mathbf{e}_r, \mathbf{e}_t$. Then, the triplets are input into the fine-tuned LLM to determine if they are correct or not by Eq. 5.

However, not all entities in generated triplet $(h, r, t) \in T$ can be mapped to those in KGs. To address this, the Review action applies a soft constraint rule to distinguish whether the generated triplet is factually incorrect or the result of incomplete knowledge in KGs, as follows:

- Factually Wrong: if we can map h and t to entities in KGs and $b = \text{False}$, then the triplet (h, r, t) is factually wrong and is removed from T.
- Incomplete Knowledge: if we cannot map either h or t to entities in KGs, then the triplet (h, r, t) is considered incomplete knowledge and is kept.

In this way, the triplet in T can be grouped into two categories, i.e., the True triplet set V and False triplet set F, where $T = V \cup F$ and $V \cap F = \emptyset$.

3.3 REVISE AND ANSWER ACTIONS

If F has triplets, KGAREVION calls the Revise action to adjust the triplets in F to include more knowledge that helps with the answering of the input question. The head and tail entities of the revised triplets are then reviewed by the Review action to make sure that they are correct and related to the input question. If the Review action outputs "True", then the revised triplets are added to the set of True triplets V. Otherwise, KGAREVION continues to call the Revise action until the max round k ($k \geq 1$) is achieved.

After obtaining the set of True triplets from the Review or Revise actions, KGAREVION finally calls the Answer action to prompt the LLM to select the most suitable answer y from the set of answer candidates C of input question q based on the triplets in V, where $y = P(q, V, C)$ and $y \in C$.

4 RESULTS

Datasets. We first start with four multi-choice medical QA benchmarks (Xiong et al., 2024a) (Table 1). In addition, we introduce a new benchmark for multi-choice complex medical QA focused on differential diagnosis (DDx), named MedDDx. We begin by collecting questions and corresponding answers from STaRK-Prime (Wu et al., 2024c). For each question, we then select the top three entities with the highest semantic similarity to serve as additional answer candidates. MedDDx comprises a total of 1,769 multi-choice QA samples. Based on the standard deviation of semantic similarity between answer candidates and the correct answer, we categorize the dataset into three difficulty levels: MedDDx-Basic, MedDDx-Intermediate, and MedDDx-Expert (The samples in each dataset are shown in Fig. 1b, and details are available in Appendix 5).

Datasets	Size	QA	OR
MMLU-Med	1,089	A/B/C/D	✔
MedQA-US	1,273	A/B/C/D	✔
PubMedQA*	500	Yes/No/Maybe	✔
BioASQ-Y/N	618	Yes/No	✔
MedDDx-Basic	483	A/B/C/D	✔
MedDDx-Intermediate	1,041	A/B/C/D	✔
MedDDx-Expert	245	A/B/C/D	✔

Table 1: QA benchmarks and three new MedDDx datasets. 'OR' indicates whether the open-ended reasoning evaluation is done.

Baselines. We consider 8 LLM-based reasoning models, 4 RAG-based models, and 3 KG-based models. The LLM-based reasoning models include LLaMA (2-7B/13B, 3-8B, 3.1-8B) (Touvron et al., 2023; Dubey et al., 2024), Mistral (Jiang et al., 2023), MedAlpaca (7B) (Han et al., 2023), PMC-LLaMA (7B) (Wu et al., 2024a), LLaMA3-OpenBioLLM-8B (Ankit Pal, 2024), and MEDITRON (Chen et al., 2023). The RAG-based models include Self-RAG (Asai et al., 2024), MedRAG (Xiong et al., 2024b), KG-RAG (Soman et al., 2023), and KG-Rank (Yang et al., 2024). The KG-based models include QAGNN (Yasunaga et al., 2021), JointLK (Sun et al., 2022), and Dragon (Yasunaga et al., 2022).

Evaluation setup. We consider two evaluation settings. **Multi-choice reasoning:** This setting evaluates the model's performances on all collected multi-choice QA datasets. The model is tasked to select the correct answer to a user input question from a set of candidate answers. **Open-ended reasoning:** All candidate answers are masked, meaning that the model has to generate a response to the input question independently without being presented with a set of candidate answers. The model produces an answer solely on its own generated response. Additionally, we design two new evaluation scenarios for each setting to test model abilities in solving complex medical questions by considering the number of medical concepts and the semantic similarity among answer candidates. **Query complexity scenario (QSS):** This is a hard evaluation scenario to test how the model performs with the increase of the number of medical concepts present in a question since the question often becomes more intricate, requiring more nuanced inferences between concepts to achieve the correct answer, with the increase of medical concepts. **Semantic complexity scenario (CSS):** This is a harder evaluation scenario that tests the model's ability to identify the correct answer among semantically similar and closely medically related candidate answers.

4.1 BENCHMARKING KGAREVION UNDER MULTI-CHOICE REASONING SETTING

Table 2 shows the accuracy and variance of KGAREVION and all baselines on all datasets. Evaluation on four gold standard medical QA datasets shows that KGAREVION improves the average accuracy by over 4.8%, outperforming all baselines in handling medical queries.

Multi-choice Reasoning	Established Medical QA Benchmarks				MedDDx		
Method	MMLU-Med	MedQA-US	PubMedQA*	BioASQ-Y/N	Basic	Intermidate	Expert
Metrics	Acc. (std)	Acc. (std)	Acc. (std)	Acc. (std)	Acc. (std)	Acc. (std)	Acc. (std)
LLaMA2-7B	0.376 (.006)	0.281 (.004)	0.448 (.010)	0.568 (.006)	0.215 (.030)	0.198 (.004)	0.192 (.012)
LLaMA2-7B (CoT)	0.318 (.005)	0.251 (.002)	0.465 (.011)	0.547 (.011)	0.289 (.010)	0.265 (.006)	0.229 (.023)
Mistral-7B	0.634 (.004)	0.477 (.007)	0.400 (.002)	0.644 (.001)	0.412 (.003)	0.356 (.003)	0.375 (.007)
Mistral-7B (CoT)	0.634 (.003)	0.474 (.002)	0.372 (.005)	0.651 (.002)	0.404 (.010)	0.368 (.023)	**0.379** (.027)
MedAlpaca-7B	0.600 (.004)	0.401 (.001)	0.333 (.015)	0.493 (.034)	0.399 (.012)	0.325 (.004)	0.311 (.009)
MedAlpaca-7B (CoT)	0.603 (.004)	0.399 (.003)	0.315 (.015)	0.485 (.025)	0.395 (.007)	0.321 (.011)	0.312 (.010)
PMC-LLaMA-7B	0.207 (.011)	0.247 (.004)	0.179 (.007)	0.346 (.017)	0.087 (.015)	0.086 (.002)	0.079 (.006)
PMC-LLaMA-7B (CoT)	0.204 (.008)	0.208 (.002)	0.125 (.014)	0.208 (.002)	0.088 (.002)	0.077 (.004)	0.063 (.005)
LLaMA3-8B	0.634 (.005)	**0.566** (.004)	**0.586** (.008)	0.654 (.005)	0.428 (.015)	0.319 (.002)	0.306 (.009)
LLaMA3-8B (CoT)	**0.651** (.003)	0.552 (.003)	0.574 (.002)	**0.681** (.002)	**0.434** (.010)	**0.368** (.004)	0.313 (.003)
Llama3-OpenBioLLM-8B	0.636 (.005)	0.383 (.003)	0.350 (.026)	0.623 (.005)	0.238 (.004)	0.235 (.011)	0.229 (.020)
Llama3-OpenBioLLM-8B (CoT)	0.571 (.003)	0.295 (.002)	0.283 (.001)	0.646 (.004)	0.370 (.013)	0.330 (.001)	0.327 (.011)
LLaMA3.1-8B	0.677 (.007)	**0.563** (.006)	0.596 (.009)	0.687 (.006)	0.434 (.018)	0.368 (.002)	0.306 (.021)
LLaMA3.1-8B (CoT)	**0.681** (.005)	0.549 (.003)	**0.600** (.005)	0.706 (.002)	**0.439** (.017)	0.393 (.005)	0.322 (.014)
LLaMA2-13B	0.442 (.002)	0.253 (.004)	0.252 (.004)	0.455 (.002)	0.286 (.003)	0.338 (.006)	0.317 (.006)
LLaMA2-13B (CoT)	0.415 (.002)	0.354 (.005)	0.232 (.006)	0.422 (.003)	0.309 (.005)	0.263 (.013)	0.243 (.016)
QAGNN	0.317 (.003)	0.450 (.005)	0.439 (.033)	0.644 (.002)	0.295 (.003)	0.265 (.002)	0.253 (.003)
JointLK	0.288 (.005)	0.472 (.003)	0.468 (.007)	0.640 (.007)	0.247 (.004)	0.250 (.003)	0.244 (.004)
Dragon	0.319 (.003)	0.475 (.002)	0.472 (.005)	0.646 (.003)	0.286 (.003)	0.247 (.005)	0.240 (.004)
Self-RAG (7B)	0.322 (.019)	0.380 (.028)	0.534 (.028)	0.594 (.012)	0.238 (.007)	0.199 (.037)	0.224 (.045)
Self-RAG (13B)	0.502 (.004)	0.408 (.020)	0.331 (.158)	0.646 (.050)	0.249 (.010)	0.290 (.018)	0.266 (.031)
KG-Rank (13B)	0.452 (.005)	0.362 (.011)	0.305 (.019)	0.503 (.015)	0.253 (.021)	0.256 (.013)	0.234 (.010)
KG-RAG (8B)	0.516 (.005)	0.343 (.001)	0.429 (.017)	0.662 (.005)	0.434 (.021)	**0.413** (.007)	**0.391** (.004)
MedRAG (70B)	0.579 (.015)	0.487 (.014)	0.574 (.022)	**0.719** (.018)	0.365 (.008)	0.348 (.011)	0.327 (.003)
KGAREVION (LLaMA3, w/o Review)	0.621 (.002)	0.528 (.003)	0.556 (.002)	0.713 (.004)	0.310 (.006)	0.334 (.004)	0.313 (.008)
KGAREVION (LLaMA3, w/o Revise)	0.657 (.004)	0.594 (.006)	0.562 (.002)	0.723 (.005)	0.386 (.008)	0.372 (.004)	0.327 (.003)
KGAREVION (LLaMA3, $k = 1$)	**0.703** (.004)	0.610 (.010)	0.562 (.002)	**0.744** (.003)	**0.473** (.008)	0.404 (.006)	0.395 (.003)
KGAREVION (LLaMA3, $k = 2$)	0.696 (.006)	0.616 (.008)	0.566 (.002)	0.723 (.010)	0.457 (.006)	0.414 (.006)	0.395 (.005)
KGAREVION (LLaMA3, $k = 3$)	0.678 (.006)	**0.628** (.002)	**0.590** (.005)	0.737 (.007)	0.469 (.008)	**0.451** (.004)	**0.411** (.005)
Improvement over best baseline	**+5.2%**	**+6.2%**	**+0.4%**	**+6.3%**	**+3.9%**	**+8.3%**	**+3.2%**
KGAREVION (LLaMA3.1, w/o Review)	0.695 (.006)	0.546 (.002)	0.560 (.004)	0.736 (.003)	0.298 (.015)	0.299 (.003)	0.327 (.006)
KGAREVION (LLaMA3.1, w/o Revise)	0.716 (.006)	0.573 (.005)	0.568 (.011)	0.749 (.003)	0.392 (.012)	0.337 (.006)	0.352 (.005)
KGAREVION (LLaMA3.1, $k = 1$)	**0.734** (.004)	0.618 (.002)	0.619 (.004)	**0.763** (.001)	**0.483** (.013)	**0.457** (.010)	0.409 (.005)
KGAREVION (LLaMA3.1, $k = 2$)	0.720 (.003)	0.616 (.005)	0.656 (.006)	0.745 (.005)	0.396 (.003)	0.454 (.008)	0.342 (.004)
KGAREVION (LLaMA3.1, $k = 3$)	0.716 (.004)	**0.620** (.003)	**0.638** (.004)	0.749 (.003)	0.469 (.012)	0.411 (.010)	**0.447** (.005)
Improvement over best baseline	**+5.3%**	**+5.7%**	**+3.8%**	**+4.4%**	**+4.4%**	**+4.4%**	**+5.6%**

Table 2: The accuracy of KGAREVION and all baselines on four gold standard and three newly created datasets under multi-choice reasoning settings. The value highlighted in Blue indicates the best result among LLM-based reasoning models with a size smaller than 8B, including LLaMA3-8B, while Red marks the top value among LLaMA3.1-8B and other types of baselines. std means the standard deviation under three runs.

Results under QSS. Fig. 3a illustrates the trends of KGAREVION alongside the top-performing baselines as the number of medical concepts increases. It can first be observed that KGAREVION outperforms baselines of the same size, regardless of the number of medical concepts involved. Moreover, KGAREVION maintains stable performance as the number of medical concepts increases and even improves when processing questions with n = 6, compared to those with n = 5. In contrast, baseline models struggle with complex questions containing 5 or 6 medical concepts. Such an observation indicates that KGAREVION advances in handling complex medical questions involving multiple medical concepts.

Results under CSS. As seen in Table 2, evaluations on three difficult levels in MedDDx indicate that KGAREVION exhibits a strong ability in handling differential diagnosis questions that request professional and accurate knowledge. In addition, the obtained results also show that KGAREVION excels in identifying the correct answer among semantically similar answer candidates since it improves the accuracy on MedDDx-Expert by 3.2% and 5.6% with LLaMA3-8B and LLaMA3.1-8B as the backbone LLM, respectively.

4.2 BENCHMARKING KGAREVION UNDER OPEN-ENDED REASONING SETTING

We transform multiple-choice questions into descriptive, open-ended ones to better simulate real-world medical scenarios, where such inquiries are more common (see details in Appendix A.3). This adjustment requires our model to generate responses without predefined choices, encouraging holistic reasoning and the integration of diverse knowledge sources. By removing answer choices, we can more effectively assess the reasoning ability of KGAREVION in complex medical situations, resulting in a more realistic evaluation of its capabilities. Table 3 shows the accuracy and variance

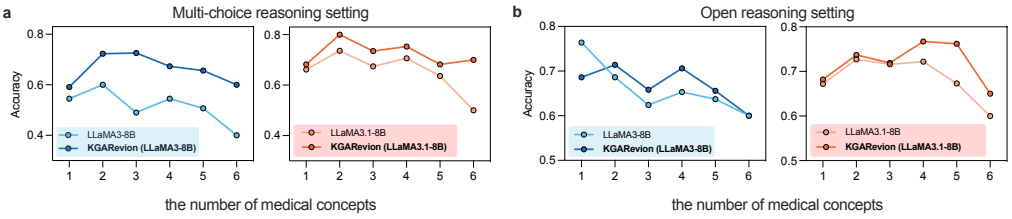

Figure 3: The accuracy of KGAREVION and pure LLMs with the medical concepts increase under **a**) multi-choice reasoning setting and **b**) open-ended reasoning setting.

Open-ended Reasoning	**Open-ended inquiries without pre-defined choices**				MedDDx-No-Opt		(**Open-ended**)
Method	MMLU-Med	MedQA-US	PubMedQA*	BioASQ-Y/N	Basic	Intermediate	Expert
Metrics	Acc. ΔAcc.	Acc. ΔAcc.	Acc. ΔAcc.	Acc. ΔAcc.	Acc. ΔAcc.	Acc. ΔAcc.	Acc. ΔAcc.
LLaMA2-7B	0.328 (-0.048)	0.302 (+0.021)	0.546 (+0.098)	0.625 (+0.057)	0.286 (+0.071)	0.305 (+0.107)	0.302 (+0.110)
LLaMA2-7B (CoT)	0.362 (+0.044)	0.243 (-0.008)	0.418 (-0.047)	0.642 (+0.095)	0.265 (-0.024)	0.270 (+0.005)	0.280 (+0.051)
Mistral-7B	0.591 (-0.043)	0.412 (-0.065)	0.344 (-0.056)	0.629 (-0.015)	0.249 (-0.163)	0.228 (-0.128)	0.273 (-0.102)
Mistral-7B (CoT)	0.583 (-0.051)	0.398 (-0.076)	0.212 (-0.160)	0.657 (+0.006)	0.245 (-0.159)	0.232 (-0.136)	0.286 (-0.093)
PMC-LLaMA-7B	0.073 (-0.134)	0.082 (-0.165)	0.090 (-0.089)	0.139 (-0.201)	0.249 (+0.162)	0.191 (+0.105)	0.232 (+0.153)
PMC-LLaMA-7B (CoT)	0.080 (-0.124)	0.079 (-0.129)	0.092 (-0.033)	0.125 (-0.083)	0.220 (+0.132)	0.245 (+0.168)	0.228 (+0.165)
LLaMA3-8B	0.595 (-0.039)	**0.458** (-0.108)	0.532 (-0.054)	0.672 (+0.018)	**0.343** (-0.085)	**0.314** (-0.005)	0.317 (+0.011)
LLaMA3-8B (CoT)	**0.608** (-0.043)	0.449 (-0.103)	**0.562** (-0.012)	**0.714** (+0.033)	0.289 (-0.145)	0.304 (-0.064)	**0.327** (+0.014)
Llama3-OpenBioLLM-8B	0.324 (-0.312)	0.157 (-0.226)	0.157 (-0.193)	0.324 (-0.299)	0.016 (-0.222)	0.006 (-0.229)	0.008 (-0.221)
Llama3-OpenBioLLM-8B (CoT)	0.398 (-0.173)	0.146 (-0.149)	0.100 (-0.183)	0.142 (-0.539)	0.098 (-0.272)	0.084 (-0.246)	0.108 (-0.219)
LLaMA3.1-8B	0.607 (-0.070)	0.551 (-0.012)	0.514 (-0.082)	0.694 (+0.007)	**0.322** (-0.112)	0.289 (-0.079)	**0.335** (+0.029)
LLaMA3.1-8B (CoT)	**0.697** (+0.016)	**0.563** (+0.014)	**0.572**(-0.028)	**0.706** (-0.000)	0.306 (-0.133)	**0.294** (-0.099)	0.315 (-0.007)
LLaMA2-13B	0.348 (-0.094)	0.283 (+0.030)	0.218 (-0.034)	0.421 (-0.034)	0.190 (-0.096)	0.123 (-0.215)	0.153 (-0.164)
LLaMA2-13B (CoT)	0.311 (-0.104)	0.267 (-0.087)	0.266 (+0.034)	0.471 (+0.049)	0.269 (-0.040)	0.268 (+0.005)	0.282 (+0.039)
Self-RAG (7B)	0.256 (-0.066)	0.235 (-0.145)	0.316 (-0.218)	0.379 (-0.215)	0.167 (-0.071)	0.246 (+0.047)	0.213 (-0.011)
Self-RAG (13B)	0.309 (-0.193)	0.297 (-0.111)	0.438 (+0.107)	0.539 (-0.107)	0.212 (-0.037)	0.232 (-0.058)	0.226 (–0.040)
KG-Rank (13B)	0.151 (-0.301)	0.189 (-0.173)	0.203 (-0.102)	0.188 (-0.315)	0.127 (-0.126)	0.133 (-0.123)	0.133 (-0.101)
KG-RAG (8B)	0.310 (-0.206)	0.290 (-0.053)	0.316 (-0.113)	0.359 (-0.303)	0.216 (-0.218)	0.220 (-0.193)	0.213 (-0.178)
KGAREVION (LLaMA3, w/o Review)	0.645 (+0.024)	0.609 (+0.081)	0.552 (-0.004)	0.701 (-0.012)	0.400 (+0.090)	0.360 (+0.026)	0.356 (+0.043)
KGAREVION (LLaMA3, w/o Revise)	0.668 (+0.011)	0.626 (+0.032)	0.572 (+0.010)	0.716 (-0.007)	0.426 (+0.040)	0.403 (+0.031)	0.412 (+0.085)
KGAREVION (LLaMA3, $k = 1$)	0.687 (-0.016)	0.628 (+0.018)	**0.578** (+0.016)	0.730 (-0.014)	0.465 (-0.008)	0.430 (+0.026)	0.428 (+0.033)
KGAREVION (LLaMA3, $k = 2$)	0.682 (-0.014)	**0.638** (+0.022)	0.566 (-0.000)	**0.736** (+0.013)	**0.527** (+0.070)	**0.463** (+0.049)	**0.489** (+0.094)
KGAREVION (LLaMA3, $k = 3$)	**0.696** (+0.018)	0.632 (+0.004)	0.572 (-0.018)	0.733 (-0.004)	0.489 (+0.020)	0.411 (-0.040)	0.429 (+0.018)
Improvement over best baseline	**+8.8%**	**+18.0%**	**+1.6%**	**+2.2%**	**+18.4%**	**+14.9%**	**+16.2%**
KGAREVION (LLaMA3.1, w/o Review)	0.659 (-0.036)	0.526 (-0.020)	0.556 (-0.004)	0.726 (-0.010)	0.457 (+0.159)	0.435 (+0.136)	0.439 (+0.112)
KGAREVION (LLaMA3.1, w/o Revise)	0.695 (-0.021)	0.626 (+0.053)	0.556 (-0.012)	0.736 (-0.013)	**0.489** (+0.097)	0.436 (+0.099)	0.451 (+0.099)
KGAREVION (LLaMA3.1, $k = 1$)	**0.720** (-0.014)	**0.644** (+0.026)	0.560 (-0.059)	**0.757** (-0.006)	0.469 (-0.014)	0.454 (-0.003)	0.437 (+0.028)
KGAREVION (LLaMA3.1, $k = 2$)	0.704 (-0.016)	0.636 (+0.020)	**0.572** (-0.084)	0.734 (-0.011)	0.469 (+0.073)	0.446 (-0.008)	0.432 (+0.090)
KGAREVION (LLaMA3.1, $k = 3$)	0.712 (-0.004)	0.639 (+0.019)	0.562 (-0.076)	0.748 (-0.001)	0.449 (-0.020)	**0.470** (+0.059)	**0.451** (+0.004)
Improvement over best baseline	**+2.4%**	**+8.1%**	**+0.0%**	**+4.9%**	**+16.7%**	**+17.6%**	**+11.6%**

Table 3: The accuracy of KGAREVION and baselines on four gold standard and three newly created datasets under open-ended reasoning settings. The value highlighted in Blue indicates the best result among LLM-based reasoning models with a size smaller than 8B, including LLaMA3-8B, while Red marks the top value among LLaMA3.1-8B and other types of baselines. ΔAcc. denotes the difference in performance between the open-ended and multiple-choice reasoning settings.

across all datasets. The variance here denotes the difference in accuracy compared with that in the multi-choice reasoning setting.

Results under QSS. Fig. 3b shows the accuracy obtained by pure LLM and KGAREVION with the increase of medical concepts under open-ended reasoning setting. Compared to pure LLMs in the open-ended reasoning setting, KGAREVION shows a significant improvement in handling complex medical reasoning tasks involving more than 4 medical concepts and a comparable performance in questions with less than 3 medical concepts.

Results under CSS. To evaluate the model's ability to solve complex medical QA with differential diagnosis, we compare KGAREVION and all baselines on newly created datasets, as shown in Table 3. KGAREVION still achieves the best performance in the open-ended reasoning setting. In addition, compared with the results in multi-choice reasoning setting, KGAREVION performs better. On the one hand, such a result demonstrates the strong ability of KGAREVION in the open-ended reasoning setting. On the other hand, it also indicates that these semantically candidates may affect the reasoning process to a certain extent.

4.3 ABLATION ANALYSES

Effect of the 'Review' action. As shown in Table 2, 3, and Fig. 4 (KGAREVION (w/o Review) vs. KGAREVION (w/o Revise))), the Review action plays an important role in answering medical questions under two settings, which improves the average accuracy across all datasets by 3.3% and 3%, respectively. Fig. 4 shows that the Review action has a more pronounced effect on the MedDDx dataset than the four gold-standard datasets under two settings, suggesting that its integration enhances the model's ability to tackle complex medical questions. Furthermore, the Review action leads to greater accuracy improvements in the four gold-standard datasets under the open-ended reasoning setting compared to the multiple-choice reasoning setting. This highlights the significance of verifying generated answers, particularly in an open-ended reasoning setup.

Number of refinement rounds in the 'Revise' action. The Revise action is designed to enhance accuracy by correcting erroneous triplets until they are verified as true by the Review action. Tables 2 and 3, and Fig. 4 demonstrate its positive impact on KGAREVION across both settings. Specifically, Figure 4 indicates that the Review action significantly improves performance on MedDDx dataset, yielding average enhancements of 9% and 4% in accuracy for both settings compared to questions in the gold-standard datasets. Additionally, we investigate the impact of the number of revision rounds across all datasets in both settings, as shown in Table 2 and 3. The results indicate that KGAREVION can achieve optimal performance with $k = 1$ on most of datasets in the multi-choice reasoning setting. However, it benefits from additional iterations when addressing complex questions, such as those in the MedDDx-Expert dataset. In the open-ended reasoning setting, KGAREVION typically requires more iterations to arrive at the correct answer.

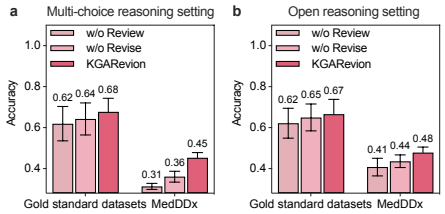

Figure 4: The results of ablation studies across all datasets under two settings.

4.4 VERSATILITY OF KGAREVION

KGAREVION can be used with different LLMs. KGAREVION is a versatile agent that can be implemented with a variety of LLMs. We implement KGAREVION using three distinct models: LLaMA3-8B, LLaMA3.1-8B, and GPT-4o. The averaged results across all datasets, as shown in Fig. 5a, demonstrate the effectiveness of KGAREVION's architecture, consistently improving the performance of the backbone LLMs by 6%, 7%, and 2%, respectively.

KGAREVION can be used with different medical KGs. The Review action in KGAREVION grounds the generated triplets using KGs. To evaluate the impact of different KGs (see details in Appendix B.5), we implement KGAREVION with two comprehensive KGs and assess its performance across all datasets, as shown in Fig. 5b. The results show that KGAREVION is not sensitive to the choice of knowledge bases, highlighting its robustness and generalizability, despite PrimeKG (Chandak et al., 2023) being much larger than OGB-biokg (Hu et al., 2020). This robustness arises because KGs are used only in the Review action to verify generated triplets rather than as a source for retrieving knowledge. This also explains why KGAREVION outperforms KG-based RAG models, which heavily rely on the chosen KGs, whereas KGAREVION uses comprehensive KGs simply to ensure that the generated triplets are aligned with medical knowledge.

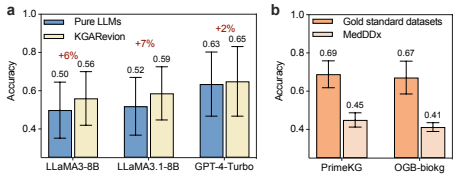

Figure 5: **a)** Performance of KGAREVION with different backbone LLMs across all datasets. **b)** Performance of KGAREVION with different KGs used in the fine-tuning stage of the Review action.

4.5 SENSITIVITY ANALYSES

Recent studies have revealed that LLMs can be surprisingly sensitive to both how candidate answers are ordered and indexed in multi-choice setups (Zheng et al., 2023; Pezeshkpour & Hruschka, 2023).

These studies found that LLMs are not robust multiple-choice selectors and exhibit order sensitivity, favoring answers at the first position (Li et al., 2024). To investigate this issue, we examine how the order and indexing of answers affect model performance. We evaluate KGAREVION using LLaMA3-8B and LLaMA3.1-8B as the backbone and compare its performance with their LLM-only counterparts across all datasets (details are provided in Appendix B.4.1). Fig. 6 illustrates the changes in accuracy when the order or labels of the candidate answers are altered.

Ordering of candidate answers in multi-choice setups. Fig. 6a shows that pure LLMs are sensitive to answer order, with an average accuracy shift of 8.4% for LLaMA3-8B and 16.0% for LLaMA3.1-8B. In contrast, KGAREVION demonstrates significantly greater robustness to answer order. This robustness is primarily due to KGAREVION's ability to fairly evaluate each answer using the Generate action, effectively mitigating the impact of answer order on model performance.

Indexing of candidate answers in multi-choice setups. The accuracy of pure LLMs shows a substantial shift when relabeling answers from ABCD to EFGH, as illustrated in Fig. 6b. Specifically, the average accuracy shift is 8.1% for LLaMA3-8B and 12.9% for LLaMA3.1-8B. In contrast, our agent KGAREVION significantly improves the stability of these LLMs, reducing the accuracy loss to 2.59% and 3.86%, respectively. This finding further highlights the robustness of KGAREVION.

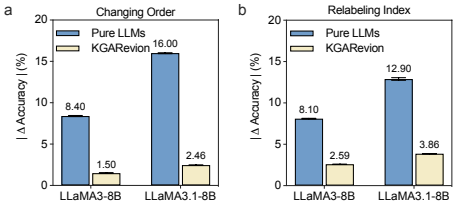

Figure 6: $|\Delta\text{Accuracy}|$ of LLMs and KGAREVION when changing order or relabeling index.

4.6 CASE STUDIES

Fig. 7 illustrates the reasoning process of KGAREVION in both settings, using the same input question. In both cases, KGAREVION arrives at the correct answer, but the reasoning processes differ. In the open reasoning setting, KGAREVION requires more iterations to revise triplets and guide the reasoning compared to the multiple-choice setting. Additionally, the verified correct triplets provide a reasoning path that helps explain the final answer, such as '*partial deletion of the long arm of chromosome 19 → associated with → 19q13.11 deletion syndrome and partial chromosome 19 deletions → associated with → disease*'.

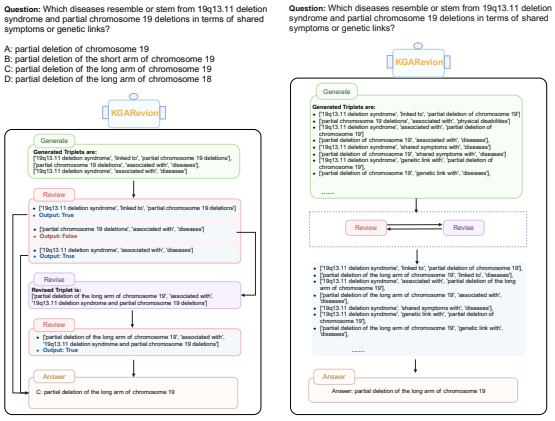

Figure 7: The reasoning process of KGAREVION under multi-choice setting (Left) and open reasoning setting (Right).

5 CONCLUSION

Medical reasoning presents unique challenges that require integrating multi-source, grounded, and specialized domain knowledge. In this work, we introduced KGAREVION, a KG-based LLM agent that addresses these challenges by combining the non-codified knowledge of LLMs with the structured, codified knowledge of medical concepts stored in KGs. Through its adaptive reasoning and mechanisms for generating, verifying, and revising knowledge, KGAREVION can handle complex medical QA. Experiments across multiple-choice and open-ended tasks, using a variety of datasets—including challenging new benchmarks—demonstrate KGAREVION's ability to systematically improve accuracy. By grounding LLM-generated knowledge in KGs, KGAREVION ensures contextual relevance and reliability, making it a valuable tool for knowledge-intensive medical QA.

REFERENCES

Vaibhav Adlakha, Parishad BehnamGhader, Xing Han Lu, Nicholas Meade, and Siva Reddy. Evaluating correctness and faithfulness of instruction-following models for question answering. *Transactions of the Association for Computational Linguistics*, 12:775–793, 2024.

Malaikannan Sankarasubbu Ankit Pal. Openbiollms: Advancing open-source large language models for healthcare and life sciences. https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B, 2024.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Marsden S Blois. Medicine and the nature of vertical reasoning. *The New England journal of medicine*, 318(13):847–851, 1988.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004.

Daniil A Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013.

Andres M Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew White, and Philippe Schwaller. Augmenting large language models with chemistry tools. In *NeurIPS 2023 AI for Science Workshop*, 2023. URL https://openreview.net/forum?id=wdGIL6lx3l.

Payal Chandak, Kexin Huang, and Marinka Zitnik. Building a knowledge graph to enable precision medicine. *Scientific Data*, 10(1):67, 2023.

Cathy Charles, Amiram Gafni, and Tim Whelan. Shared decision-making in the medical encounter: what does it mean?(or it takes at least two to tango). *Social science & medicine*, 44(5):681–692, 1997.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, et al. Meditron-70b: Scaling medical pretraining for large language models. *arXiv preprint arXiv:2311.16079*, 2023.

Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. Bias and unfairness in information retrieval systems: New challenges in the llm era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 6437–6447, 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, and Angela Fan. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 6491–6501, 2024.

Shanghua Gao, Ada Fang, Yepeng Huang, Valentina Giunchiglia, Ayush Noori, Jonathan Richard Schwarz, Yasha Ektefaie, Jovana Kondic, and Marinka Zitnik. Empowering biomedical discovery with ai agents. *arXiv preprint arXiv:2404.02831*, 2024.

Ojas Gramopadhye, Saeel Sandeep Nachane, Prateek Chanda, Ganesh Ramakrishnan, Kshitij Sharad Jadhav, Yatin Nandwani, Dinesh Raghu, and Sachindra Joshi. Few shot chain-of-thought driven reasoning to prompt llms for open ended medical question answering. *arXiv preprint arXiv:2403.04890*, 2024.

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*, 2023.

Emily Harris. Large language models answer medical questions accurately, but can't match clinicians' knowledge. *JAMA*, 2023.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C Park. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. *arXiv preprint arXiv:2403.14403*, 2024.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Pengcheng Jiang, Cao Xiao, Adam Richard Cross, and Jimeng Sun. Graphcare: Enhancing healthcare predictions with personalized knowledge graphs. In *The Twelfth International Conference on Learning Representations*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for" mind" exploration of large language model society. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Wangyue Li, Liangzhi Li, Tong Xiang, Xiao Liu, Wei Deng, and Noa Garcia. Can multiple-choice questions really be useful in detecting the abilities of llms? *arXiv preprint arXiv:2403.17752*, 2024.

Valentin Liévin, Christoffer Egeberg Hother, Andreas Geert Motzfeldt, and Ole Winther. Can large language models reason about medical questions? *Patterns*, 5(3), 2024.

Jörg Menche, Amitabh Sharma, Maksim Kitsak, Susan Dina Ghiassian, Marc Vidal, Joseph Loscalzo, and Albert-László Barabási. Uncovering disease-disease relationships through the incomplete interactome. *Science*, 347(6224):1257601, 2015.

Lino Murali, G Gopakumar, Daleesha M Viswanathan, and Prema Nedungadi. Towards electronic health record-based medical knowledge graph construction, completion, and applications: A literature study. *Journal of biomedical informatics*, 143:104403, 2023.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

OpenAI. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=TG8KACxEON.

Vimla L Patel, José F Arocha, and Jiajie Zhang. Thinking and reasoning in medicine. *The Cambridge handbook of thinking and reasoning*, 14:727–750, 2005.

Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. *arXiv preprint arXiv:2308.11483*, 2023.

Kunxun Qi, Jianfeng Du, and Hai Wan. Learning from both structural and textual knowledge for inductive knowledge graph completion. *Advances in Neural Information Processing Systems*, 36, 2024.

Alec Radford. Improving language understanding by generative pre-training. 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=Yacmpz84TH.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. HuggingGPT: Solving AI tasks with chatGPT and its friends in hugging face. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=yHdTscY6Ci.

Zhengliang Shi, Shuo Zhang, Weiwei Sun, Shen Gao, Pengjie Ren, Zhumin Chen, and Zhaochun Ren. Generate-then-ground in retrieval-augmented generation for multi-hop question answering. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7339–7353, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.acl-long.397.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik R Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

Karthik Soman, Peter W Rose, John H Morris, Rabia E Akbas, Brett Smith, Braian Peetoom, Catalina Villouta-Reyes, Gabriel Cerono, Yongmei Shi, Angela Rizk-Jackson, et al. Biomedical knowledge graph-enhanced prompt generation for large language models. *arXiv preprint arXiv:2311.17330*, 2023.

Yueqing Sun, Qi Shi, Le Qi, and Yu Zhang. Jointlk: Joint reasoning with language models and knowledge graphs for commonsense question answering. In Marine Carpuat, Marie-Catherine de Marneffe, and Iván Vladimir Meza Ruíz (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pp. 5049–5060. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.NAACL-MAIN.372. URL https://doi.org/10.18653/v1/2022.naacl-main.372.

Xiangru Tang, Anni Zou, Zhuosheng Zhang, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. Medagents: Large language models as collaborators for zero-shot medical reasoning. *arXiv preprint arXiv:2311.10537*, 2023.

Nandan Thakur, Luiz Bonifacio, Xinyu Zhang, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Boxing Chen, Mehdi Rezagholizadeh, and Jimmy Lin. Nomiracl: Knowing when you don't know for robust multilingual retrieval-augmented generation. *CoRR*, abs/2312.11361, 2023. doi: 10.48550/ARXIV.2312.11361. URL https://doi.org/10.48550/arXiv.2312.11361.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, and Nikolay Bashlykov. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

A Venigalla, Jonathan Frankle, and M Carbin. Biomedlm: a domain-specific large language model for biomedical text. *MosaicML. Accessed: Dec*, 23(3):2, 2022.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. In *Intrinsically-Motivated and Open-Ended Learning Workshop @NeurIPS2023*, 2023a. URL https://openreview.net/forum?id=nfx5IutEed.

Yubo Wang, Xueguang Ma, and Wenhu Chen. Augmenting black-box llms with medical textbooks for clinical question answering. *arXiv preprint arXiv:2309.02233*, 2023b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9781713871088.

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Weidi Xie, and Yanfeng Wang. Pmc-llama: toward building open-source language models for medicine. *Journal of the American Medical Informatics Association*, pp. ocae045, 2024a.

Junde Wu, Jiayuan Zhu, and Yunli Qi. Medical graph rag: Towards safe medical large language model via graph retrieval-augmented generation. *arXiv preprint arXiv:2408.04187*, 2024b.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. *arXiv preprint arXiv:2308.08155*, 2023.

Shirley Wu, Shiyu Zhao, Michihiro Yasunaga, Kexin Huang, Kaidi Cao, Qian Huang, Vassilis N. Ioannidis, Karthik Subbian, James Zou, and Jure Leskovec. Stark: Benchmarking llm retrieval on textual and relational knowledge bases. 2024c.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. Benchmarking retrieval-augmented generation for medicine. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics ACL 2024*, pp. 6233–6251, Bangkok, Thailand and virtual meeting, August 2024a. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-acl.372.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. Benchmarking retrieval-augmented generation for medicine, 2024b. URL https://arxiv.org/abs/2402.13178.

Rui Yang, Haoran Liu, Edison Marrese-Taylor, Qingcheng Zeng, Yuhe Ke, Wanxin Li, Lechao Cheng, Qingyu Chen, James Caverlee, Yutaka Matsuo, and Irene Li. KG-rank: Enhancing large language models for medical QA with knowledge graphs and ranking techniques. In Dina Demner-Fushman, Sophia Ananiadou, Makoto Miwa, Kirk Roberts, and Junichi Tsujii (eds.), *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pp. 155–166, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.bionlp-1.13.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. QA-GNN: Reasoning with language models and knowledge graphs for question answering. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 535–546, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.45. URL https://aclanthology.org/2021.naacl-main.45.

Michihiro Yasunaga, Antoine Bosselut, Hongyu Ren, Xikun Zhang, Christopher D. Manning, Percy Liang, and Jure Leskovec. Deep bidirectional language-knowledge graph pretraining. In *Neural Information Processing Systems (NeurIPS)*, 2022.

Jiahao Zhang, Haiyang Zhang, Dongmei Zhang, Liu Yong, and Shen Huang. End-to-end beam retrieval for multi-hop question answering. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1718–1731, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.96. URL https://aclanthology.org/2024.naacl-long.96.

Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. Verify-and-edit: A knowledge-enhanced chain-of-thought framework. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5823–5840, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.320. URL https://aclanthology.org/2023.acl-long.320.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*, 2023.

APPENDIX

Datasets **17**
Gold Standard Multi-choice Medical QA Dataset 17
Newly Created Medical QA Dataset 17
Conversion of Multi-Choice Type Questions To Descriptive Type 18

Implementation Details **18**
Experiment Environments 18
Fine-tuning Details 19
Implementation Details of Baseline Models 19
Implementation Details of Sensitivity Analysis 20
Effect of Answer Order/Index 20
Implementation Details of Adaptability Analysis 20

KGARevion **21**

Prompts **21**
Prompt Template for Evaluating Baseline Models Under Multi-choice Reasoning Setting 21
Prompt Template for Evaluating Baseline Models Under Open Reasoning Setting 22
Prompt Template in KGAREVION 22

Tables **25**
Description Template 25
Notations 26

A DATASETS

A.1 GOLD STANDARD MULTI-CHOICE MEDICAL QA DATASET

In this work, we use four well-known multi-choice medical QA datasets to evaluate the model performance, including two medical examination QA datasets (MMLU-Med, MedQA-US) and two biomedical research QA datasets (PubMedQA*, BioASQ-Y/N). These datasets are derived from (Xiong et al., 2024a). The samples in these datasets are shown in Table 4.

Dataset Name	Sample
MMLU-Med	Which of the following best describes the structure that collects urine in the body? A: Bladder B: Kidney C: Ureter D: Urethra
MedQA-US	A microbiologist is studying the emergence of a virulent strain of the virus. After a detailed study of the virus and its life cycle, he proposes a theory: Initially, a host cell is co-infected with 2 viruses from the same virus family. Within the host cell, concomitant production of various genome segments from both viruses occurs. Ultimately, the different genome segments from the viruses are packaged into a unique and novel virus particle. The newly formed virus particle is both stable and viable and is a new strain from the virus family that caused the outbreak of infection. Which of the following viruses is capable of undergoing the above-mentioned process? A: Epstein-Barr virus B: Human immunodeficiency virus C: Rotavirus D: Vaccinia virus
PubMedQA*	Is anorectal endosonography valuable in dyschesia? A: yes B: no C: maybe
BioASQ-Y/N	Can losartan reduce brain atrophy in Alzheimer's disease? A: yes B: no

Table 4: Examples of four widely used medical QA datasets

A.2 NEWLY CREATED MEDICAL QA DATASET – MEDDDX

MedDDx is a newly constructed dataset designed to test model performance on semantically complex answers. The motivation behind creating this dataset is twofold:

- While large language models (LLMs) can perform QA tasks, they often rely heavily on semantic dependencies, making it difficult for them to identify the correct answer among semantically similar answer candidates;
- In real-world medical scenarios, researchers often focus on identifying subtle differences between similar molecules, particularly in treatment or diagnostic settings. For instance, proteins may share similar names but have significantly different structures and functions, making it crucial to distinguish these differences to be able to provide accurate answers (see Table 5 for an example).

Because of these reasons, we construct MedDDx, a multi-choice medical QA dataset that focuses on answering semantically complex multi-choice QA. These questions are sourced from STaRK-Prime (Wu et al., 2024c), which provides both the questions and their corresponding answers. We extract questions with a single correct answer from the STaRK-Prime testing set and transform them into the multi-choice format. To generate three strong alternative answer candidates, we use semantic similarity to increase the difficulty, selecting the top three entities that have the highest semantic similarity as the correct answer. The semantic embeddings used for this process are derived from `text-embedding-ada-002` model from OpenAI.

Dataset Name	Sample
Basic	Can you recommend medications effective against peptic ulcer disease that also suppress Helicobacter pylori in the stomach? A: Rebamipide B: Ecabet C: Bendazac D: Nepafenac
Intermediate	Can you recommend medications that treat both eosinophilic pneumonia and a parasitic worm infection? A: Thiabendazole B: Albendazole C: Diethylcarbamazine D: Triclabendazole
Expert	Which genes or proteins are expressed exclusively in the pericardium and not in either the dorsal or ventral regions of the thalamus? A: ADH1A B: ADH1C C: ADH4 D: ADH1B

Table 5: Examples of four widely used medical QA datasets.

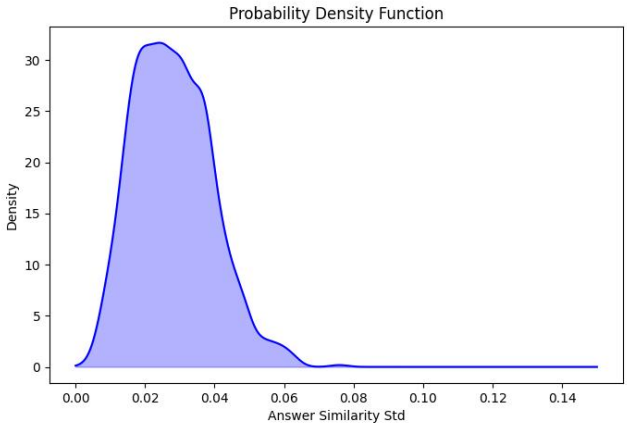

Figure 8: The distribution of the standard deviation of semantic similarities between answer candidates and the correct answer. A lower value indicates greater similarity among the answers.

The semantic similarity is calculated using cosine similarity. We also compute the standard deviation of semantic similarity between the correct answer and the other three candidates. The density distribution of these values is shown in Fig. 8. Based on this distribution, we divide the queries into three complexity groups using quantile analysis: MedDDx-Expert (0-0.02), MedDDx-Intermediate (0.02-0.04), and MedDDx-Basic (>0.04).

A.3 Conversion of Multi-Choice Type Questions to Descriptive Type

The conversion of multi-choice questions to descriptive ones is aimed to evaluate real-world medical scenarios where open-ended inquiries are prevalent. To achieve this, we modify the question by adding more descriptive terms, as shown in Table 6.

B Implementation Details

B.1 Experiment Environments

Hardware. All experiments are conducted on a machine equipped with 4 NVIDIA H100. We use 1 NVIDIA H100 to implement baselines with small LLMs. In the fine-tuning stage, we use 4 NVIDIA H100 to fine-tune the review module.

Multi-Choice type	Open-ended type
Which of the following best describes the structure that collects urine in the body? A: Bladder B: Kidney C: Ureter D: Urethra	What best describes the structure that collects urine in the body?
A microbiologist is studying the emergence of a virulent strain of the virus. After a detailed study of the virus and its life cycle, he proposes a theory: Initially, a host cell is co-infected with 2 viruses from the same virus family. ..... Which of the following viruses is capable of undergoing the above-mentioned process? A: Epstein-Barr virus B: Human immunodeficiency virus C: Rotavirus D: Vaccinia virus	A microbiologist is studying the emergence of a virulent strain of the virus. After a detailed study of the virus and its life cycle, he proposes a theory: Initially, a host cell is co-infected with 2 viruses from the same virus family. ..... Which virus is capable of undergoing the above-mentioned process?

Table 6: Examples of conversation of multi-choice type question to descriptive type.

Software. We implement KGAREVION using Python 3.9.19, PyTorch 2.3.1, Transformers 4.43.1, and Tokenizers 0.19.1. All LLMs adopted in this study are downloaded from Hugging Face, except for OpenAI models.

B.2 FINE-TUNING DETAILS

During the fine-tuning stage, we first split PrimeKG Chandak et al. (2023) into two parts: a training set and a testing set, in a ratio of 8:2. We use LoRA to fine-tune the LLMs on a single machine equipped with 4 NVIDIA H100 GPUs for knowledge graph completion tasks in the Review action.

For hyperparameter tuning, we use grid search to identify the optimal parameter combinations by evaluating the fine-tuned model's performance on the knowledge graph completion task using the testing set. Specifically, we focus on the parameter r in LoRA training and the batch size during the fine-tuning stage. The values explored for r are 16, 32, 64, 128, while the tested batch sizes bz are 128, 256, 512, 1024. The best parameters identified are $r = 32$, $bz = 256$.

B.3 IMPLEMENTATION DETAILS OF BASELINE MODELS

All the results reported in this paper are averaged over multiple runs with different random seeds. For consistency and reproducibility, we select three commonly used seeds: 42, 777, and 1234.

B.3.1 IMPLEMENTATION DETAILS OF BASELINE MODELS UNDER MULTI-CHOICE SETTING

LLM-based Reasoning Models. We evaluate these LLMs across all datasets in two distinct settings: the traditional inference setting and the CoT inference setting, using the Transformers package (v 4.31.1) with default parameters. The prompts used for evaluation are provided in D.1.

KG-based Models. Since the baseline models are trained in an end-to-end way, they cannot be applied directly. To ensure fairness, we train these three models on a collective dataset called MedMCQA, which contains 4182 question-answering samples, and then evaluate them on all datasets used in this study. In addition, we utilize PrimeKG as the knowledge base for all three methods to construct subgraphs for each sample in both the training and evaluation stages. For knowledge graph processing, we follow the same procedure as JointLK, converting each entity in the KG to its corresponding UMLS code and retrieving entities that match those in the question to construct the subgraph for each question. In the case of Dragon, since it requires a pre-training stage, we first complete its pre-training on the MedMCQA dataset and then directly infer the model on all datasets adopted in this study. We set the epoch number for all three models as 20. All other parameters were derived from the original publications.

RAG-based Models. We implement the Self-RAG with Llama2-7B/13B with VLLM (v 0.4.3). We set the temperature in *SamplingParams* as the default value as their original LLMs. MedRAG is implemented using the original code repository. We adopt Llama2-70B as its backbone model, '*textbooks*' as corpus, and MedCPT as retriever. KG-RAG is implemented with the same KG as in KGAREVION, which is PrimeKG and is utilized only during the inference stage. For KG-Rank, we report the results using its original backbone model, which is Llama2-13B and adopt the MedCPT as the ranking method due to limitations with the Cohere API.

B.3.2 IMPLEMENTATION DETAILS OF BASELINE MODELS UNDER OPEN-ENDED REASONING SETTING

To test baseline models under the open-ended reasoning setting, we add a new module for both LLM-based reasoning models and RAG-based models. In terms of KG-based models, they could not be deployed under the open-ended reasoning setting due to their model architectures, as shown in D.2.

For each model in the other two groups, we first get a response to the input descriptive query and then adopt the same LLM as their original design to match the response to the content of the correct answer without considering the input question.

B.4 IMPLEMENTATION DETAILS OF SENSITIVITY ANALYSIS

B.4.1 EFFECT OF ANSWER ORDER/INDEX

To assess the sensitivity of each model to the answers' order, we swap the positions of two options. For datasets with four options, the order is changed from ABCD to BCAD. For three-option datasets, the order is adjusted from ABC to CAB, and for two-option datasets, it is reversed from AB to BA. It is important to note that we alter the order along with the corresponding content, not just the answer labels. To test the model sensitivity to the answers' index, we relabel the answer indices from ABCD to EFGH. The question obtained after changing the order or the index is presented in Table 7.

	Sample
Original	Which of the following best describes the structure that collects urine in the body? A: Bladder B: Kidney C: Ureter D: Urethra
Changing Order	Which of the following best describes the structure that collects urine in the body? B: Kidney C: Ureter A: Bladder D: Urethra
Relabeling Index	Which of the following best describes the structure that collects urine in the body? E: Bladder F: Kidney G: Ureter H: Urethra

Table 7: Examples of four widely used medical QA datasets.

B.5 IMPLEMENTATION DETAILS OF ADAPTABILITY ANALYSIS

To show the flexibility of KGAREVION, we further conduct adaptability analysis by implementing the model with different backbones and KGs. Overall, in this work, we test the KGAREVION with two KGs (i.e., PrimeKG and OGB-biokg). Their details are as follows:

PrimeKG (Chandak et al., 2023) is a precision medicine-oriented knowledge graph that provides a holistic view of diseases. PrimeKG integrates 20 high-quality resources to describe 17,080 diseases with 4,050,249 relationships representing ten major biological scales, including disease-associated protein perturbations, biological processes and pathways, anatomical and phenotypic scale, and the entire range of approved and experimental drugs with their therapeutic action, considerably expanding previous efforts in disease-rooted knowledge graphs.

The OGB-biokg (Hu et al., 2020) dataset contains 5 types of entities: diseases (10,687 nodes), proteins (17,499), drugs (10,533 nodes), side effects (9,969 nodes), and protein functions (45,085

nodes). There are 51 types of directed relations connecting two types of entities, including 38 kinds of drug-drug interactions, 8 kinds of protein-protein interaction, as well as drug-protein, drug-side effect, and function-function relations.

C KGAREVION

C.1 IMPLEMENTATION DETAILS IN REVISE ACTION

The Revise action is carried out by providing an instruction to an unfine-tuned LLM, such as LLaMA3.1-8B, instead of retraining the model. The instruction directs the LLM to correct the "False" triplet identified by the Review action by modifying either the head or tail entity.

- **Input of False Triplets** The Revise action begins by receiving False triplets, such as (HSPA1A, interactions, DHDDS), which were flagged as False by the Review action.
- **Instruction to the LLM** The system then provides an instruction to the LLM, telling it that the given triplet is incorrect. The instruction asks the LLM to revise the triplet, replacing either the head or tail entity, and to generate a corrected triplet that is relevant to the input query. For example, the instruction could be: "The following triplet is incorrect: (HSPA1A, interactions, DHDDS). Please revise it to a correct triplet related to the input query." The details of Instruction is available in D.3
- **Revised Triplet Generation** Simultaneously, the LLM is prompted to output the revised triplet in a specified format, such as a JSON structure: {'RevisedTriplets': (HSPA1B, interactions, DHDDS)}.
- **Review of Revised Triplet** Once the revised triplet is generated, KGARevion sends it to the Review action for validation. The Review action checks the correctness of the newly generated triplet.
- **Iterative Revision Process**If the revised triplet is identified as True by the Review action, it is passed on to the Answer action for final output. If the revised triplet is still identified as False, it is sent back to the Revise action for further modification. This process repeats until either a True triplet is identified or the maximum number of revision rounds is reached.

C.2 IMPLEMENTATION DETAILS IN ANSWER ACTION

The Answer action is performed by prompting an unfine-tuned LLM, such as LLaMA3.1-8B, instead of retraining the model. This approach mirrors the Revise action, allowing the system to leverage pre-trained models without the need for additional training.

- **Input of Verified Triplets** The input to the Answer action consists of all triplets that have been verified as True by the Review actions.
- **Instruction to the LLM** Once the verified triplets are received, KGARevion sends an instruction to the LLM. This instruction directs the LLM to generate the final answer based on the provided triplets.
- **Final Answer Generation** The LLM receives all True triplets and outputs the final answer based on those verified True triplets.

D PROMPTS

D.1 PROMPT TEMPLATE FOR EVALUATING BASELINE MODELS UNDER MULTI-CHOICE REASONING SETTING

Prompts for Evaluating LLMs

The following is a multiple-choice medical question. Please select and provide the correct answer from options 'A', 'B', 'C' or 'D'.

Question: {question}

Answer:

Prompts for Evaluating LLMs with CoT

The following is a multiple-choice medical question. Let's think step by step. Please select and provide the correct answer from options 'A', 'B', 'C' or 'D'.

Question: {question}

Answer:

D.2 Prompt Template for Evaluating Baseline Models Under Open Reasoning Setting

Prompts for Evaluating LLMs

The following are medical questions. Please generate a response for input question.

Question: {question}

Answer:

Prompts for Evaluating LLMs with CoT

The following are medical questions. Let's think step by step. Please generate a response for input question.

Question: {question}

Answer:

Prompts for Evaluating LLMs

Given a context, please select the most match answer from options by using 'A', 'B', 'C', and 'D'.

Context: {context}

Options: {options}

Answer:

D.3 Prompt Template in KGARevion

The Generate action is implemented with two prompts. One is responsible for identifying medical concepts involved in question stem, the other is for generating triplets.

Prompts for Generate Action

Instruction:

Given the following multiple-choice question, extract all relevant medical entities contained within the question stem. Identify and extract all medical entities, such as diseases, proteins, genes, drugs, phenotypes, anatomical regions, treatments, or other relevant medical entities. Ensure that the extracted entities are specific and medically relevant. If no medical entities are found in a particular part, return an empty list for that section. Only return the extracted entities in JSON format with the key "medical_terminologies" and the value is a list of extracted entities.

Input:

Question: {question}

Response:

Prompts for Generate Action

Instruction:

Given the following question stem, medical terminologies, and options, generate a set of related undirected triplets. Each triplet should consist of a head entity, a relation, and a tail entity. The relations should describe meaningful interactions or associations between the entities, particularly in a medical or biomedical context. Use the query stem and the medical entities contained each option to extract triplets that are relevant to the query and can answer the query correctly. Each triplet should be in the format: (Head Entity, Relationship, Tail Entity). Since the triplets are undirected, the order of Head Entity and Tail Entity does not imply any directional relationship between them. The relationship should be one of the following: ['protein_protein', 'carrier', 'enzyme', 'target', 'transporter', 'contraindication', 'indication', 'off-label use', 'synergistic interaction', 'associated with', 'parentchild', 'phenotype absent', 'phenotype present', 'side effect', 'interacts with', 'linked to', 'expression present', 'expression absent']. Ensure that each entity in the triplet is specific and concise, such as diseases, proteins, conditions, symptoms, drugs, treatments, anatomical parts, or other relevant medical entities.

Generate 1-3 triplets for each option, focusing on the ones most relevant to answering the query.

Only return the generated triplets in a structured JSON format with the key as "Triplets" and the value as a list of triplets. The format should be: "Triplets": [(Head Entity, Relationship, Tail Entity), (Head Entity, Relationship, Tail Entity)]

Input:

Question: {query_stem}

Medical_Terminologies: {medical terminologies}

Options: {option}

Response:

Prompts for Review Action

Below is an instruction that describes a task, paired with an input that provides further context. Write a response that appropriately completes the request.

Instruction:

Given a triple from a knowledge graph. Each triple consists of a head entity, a relation, and a tail entity. Taking (PHYHIP, protein_protein, KIF15) as an example, it means that protein PHYHIP has an interaction with protein KIF15. Please determine the correctness of the triple and response True or False. Please directly output 'True' or 'False'.

Input:

{triplet}

Response:

Prompts for Revise Action

Instruction:

Given the following triplet consisting of a head entity, relation, and tail entity, please review and revise the triplet to ensure it is correct and helpful for answering given question. The revision should focus on correcting the head entity, relation, or tail entity as needed to make the triplet accurate and relevant. The triplet should follow the format (head entity, relation, tail entity). Ensure that the revised triplet is factually accurate and contextually appropriate. The relation should clearly define the relationship between the head entity and the tail entity. If no changes are necessary, return the original triplet.

Only return the revised triplet in JSON format with the key 'Revised_Triplets' and the value as the corrected triplet. The format should be: "Revised_Triplets": [(Head Entity, Relationship, Tail Entity)]

Input:

Triplets: t

Questions: q

Response:

E TABLES

E.1 DESCRIPTION TEMPLATE

Relation	Description
protein_protein	Protein {A} interacts with protein {B}, indicating that the two proteins directly or indirectly associate with each other to perform a biological function.
carrier	$\{A\}$ acts as a carrier for $\{B\}$, facilitating its transport or delivery to specific locations within the body or within a cell
enzyme	$\{A\}$ functions as an enzyme that catalyzes a reaction involving $\{B\}$, converting it into a different molecule or modifying its structure
target	$\{A\}$ serves as a target for $\{B\}$, meaning that $\{B\}$ binds to or interacts with $\{A\}$ to exert its biological effect.
transporter	$\{A\}$ is a transporter that facilitates the movement of $\{B\}$ across cellular membranes or within different compartments of the body.
contraindication	The interaction between $\{A\}$ and $\{B\}$ is contraindicated, meaning that the presence of one molecule may have adverse effects or reduce the efficacy of the other
indication	$\{A\}$ is indicated for the treatment or management of a condition associated with $\{B\}$, suggesting that $\{A\}$ has a therapeutic role related to $\{B\}$
off-label use	$\{A\}$ is used off-label in relation to $\{B\}$, meaning it is utilized in a manner not specifically approved but based on clinical judgment.
synergistic interaction	$\{A\}$ and $\{B\}$ interact synergistically, where their combined effect is greater than the sum of their individual effects
associated with	$\{A\}$ is associated with $\{B\}$, indicating a relationship or correlation between the two, often in the context of disease or biological processes
parent-child	$\{A\}$ is related to $\{B\}$ in a parent-child relationship, where $\{A\}$ gives rise to or influences the formation of $\{B\}$
phenotype absent	The interaction between $\{A\}$ and $\{B\}$ results in the absence of a specific phenotype, indicating that the normal trait is not expressed
phenotype present	The interaction between $\{A\}$ and $\{B\}$ results in the presence of a specific phenotype, indicating that a particular trait is expressed
side effect	The interaction between $\{A\}$ and $\{B\}$ can cause a side effect, where the presence of one molecule leads to unintended and possibly adverse effects
interacts with	$\{A\}$ interacts with $\{B\}$, indicating a general interaction that may involve binding, modulation, or other forms of molecular communication.
linked to	$\{A\}$ is linked to $\{B\}$, suggesting a connection or association between the two molecules, often in a biological or pathological context.
expression present	$\{A\}$ is expressed in the presence of $\{B\}$, indicating that the existence or activity of $\{B\}$ leads to or correlates with the expression of $\{A\}$
expression absent	$\{A\}$ is not expressed in the presence of $\{B\}$, indicating that the existence or activity of $\{B\}$ suppresses or does not correlate with the expression of $\{A\}$

Table 8: The description templates

E.2 NOTATIONS

Variable	Description
Q	A set of medical queries
q	One question stem in Q
C	A set of answer candidates for one question stem
a	Correct answer in C for the question q
P	A large language model
G	Knowledge graph
h	A head entity in one triplet
r	A relationship in one triplet
t	A tail entity in one triplet
T	A set of triplets generated in the Generation action
a_i	One answer candidate in C
M	A set of medical concepts in q
$\mathbf{e}_h$	Pre-trained embeddings of h
$\mathbf{e}_r$	Pre-trained embeddings of r
$\mathbf{e}_t$	Pre-trained embeddings of t
d	Dimension of pre-trained embeddings
f	The fine-tuned model used in Review action
b	Bool value that determining the correctness of triplet
D	The description dictionary for relationship r
$\|l\|$	The max length of description tokens
d_p	The dimension of token embeddings in LLM
$\mathbf{X}$	Token embedding matrix with the shape of $\|l\| \times d_p$ obtained from P
g	A linear layer to map the dimension of pre-trained embeddings d to that of token embeddings d_p
$\mathbf{V}$	The triplet embedding matrix
$\mathbf{Z}$	Aligned triplet embedding matrix
$\sigma(\cdot)$	The $\mathrm{Softmax}$ function
$\varphi(\cdot)$	The layer normalization function
$\mathbf{W}_1$ and $\mathbf{W}_2$	Trainable parameters in the two layer forward neural network
d_h	The dimension of hidden layers in the two layer forward neural network
s	An instruction to the LLM
V	True triplet set determined by Review action
F	False triplet set determined by Review action
k	Max round of iterative review and revise actions
y	Predicted answer from the Answer action

Table 9: Additional notation.