Appendices

A Related Works

This section provides comprehensive technical details and broader context for the related work summarized in Section 2, focusing on the methodological foundations and evaluation challenges that motivate our framework.

A.1 KG-RAG Methods.

Knowledge graph-based retrieval-augmented generation systems address hallucination and factual grounding limitations in large language models by integrating structured symbolic knowledge into the generation process. These systems retrieve relevant subgraphs based on input queries to serve as structured context, effectively combining broad parametric knowledge with the precision and verifiability of knowledge bases.

Recent developments have produced several distinct architectural approaches addressing different aspects of the integration challenge. Graph-guided reasoning methods, exemplified by Reasoning-on-Graph (`RoG`) [32], enable interpretable multi-step reasoning by having language models explicitly verbalize their traversal through knowledge graph structures, generating relation paths that are then grounded in actual graph connections. This approach provides both answer generation and explanation capabilities, making the reasoning process more transparent and debuggable. Extensions include GNN-RAG [35], which incorporates graph neural networks to better capture structural patterns in retrieved subgraphs. In contrast, constrained generation approaches like Graph-Constrained

Reasoning (GCR) [33] focus on ensuring faithful outputs by incorporating explicit graph-based constraints into the decoding process, using KG-Trie structures to restrict generation to only those paths that exist in the knowledge base.

Modular and memory-augmented architectures represent another important direction. Frameworks like FRAG [18] propose adaptive combinations of different retrieval and generation components, while Generate-on-Graph [69] treats the language model as both an agent and a knowledge graph component, enabling interactive knowledge graph expansion during question answering. Memory-augmented approaches such as MemQ [68] introduce dedicated memory modules that separate language model reasoning from knowledge graph tool usage. More sophisticated integration strategies include DeCAF [71], which jointly generates natural language answers and corresponding logical forms, and ReknoS [64], which introduces abstract relationship reasoning through super-relations for complex compositional queries.

A.2 KGQA Benchmarks.

Despite major progress in KGQA dataset construction, existing benchmarks exhibit persistent limitations that hinder effective evaluation of modern KG-augmented retrieval systems. Early human-curated datasets such as WebQuestions [6] and ComplexQuestions [5] focused on capturing authentic user queries and introducing compositional constraints, but these resources are dominated by simple factoid questions or contain ambiguities and incomplete annotations, making them insufficient for testing models that require deeper reasoning and precise answer grounding.

To address these shortcomings, semi-automated and automated approaches have become prevalent. Hybrid pipelines like LC-QuAD [60, 14] and GraphQuestions [52] use SPARQL templates and graph-structured logic to guide question generation, with subsequent human editing for naturalness. While this increases structural diversity and complexity, the template-based nature often leads to unnatural or overly constrained question styles. More recent fully automated frameworks, including Maestro [41], CHATTY-Gen [38], and DYNAMIC-KGQA [10], attempt to scale question generation via rules, popularity heuristics, or dialogue simulation, yet these methods still struggle to balance natural language fluency, logical completeness, and answer correctness for challenging multi-hop or compositional queries.

The evolution of KGQA benchmarks reflects the field's growing complexity, progressing from simple factoid questions to sophisticated multi-hop reasoning scenarios. Early benchmarks established foundational paradigms: WebQuestions pioneered collecting natural language questions through search engine suggestions with human annotation, WebQuestionsSP [70] added SPARQL annotations aligned with Freebase, and ComplexWebQuestions [56] advanced complexity by programmatically generating SPARQL queries with logical constructs. However, these early datasets relied heavily on Freebase, which ceased maintenance in 2016, motivating migration efforts to more sustainable knowledge bases like DBpedia and Wikidata, though such migrations often introduced conceptual mismatches and alignment difficulties. Specialized evaluation objectives drove targeted benchmark development. GrailQA [20] introduced systematic evaluation of generalization scenarios including i.i.d., compositional, and zero-shot settings, while KQAPro [8] emphasized compositional reasoning through explicit program annotations and MetaQA [76] focused on multi-hop reasoning using controlled movie domain knowledge graphs. Conversational benchmarks like CSQA [49] introduced multi-turn dialogue structures, and recent work incorporated dialogue-style questions with coreference and ellipsis to better reflect real-world query patterns. Meanwhile, evaluation methodology improvements from CBench [42] and SmartBench [40] provided comprehensive analysis of SPARQL query complexity and linguistic diversity.

Despite these varied approaches, our systematic audit reveals that annotation quality remains a persistent challenge across benchmarks, with even widely-used datasets like WebQSP and CWQ exhibiting correctness rates below 60 percent. Furthermore, the predominant reliance on exact-match evaluation metrics fails to capture semantic equivalence, leading to systematic underestimation of model capabilities and underscoring the need for more rigorous benchmark construction methodologies that prioritize both annotation accuracy and semantically-aware evaluation protocols.

B KGQA Datasets and Inspection Protocol

B.1 Basic Dataset Statistic

This section provides additional context for the dataset analysis presented in Section 3, reviewing existing KGQA benchmarks and their construction methodologies. We categorize these datasets based on their generation approaches and underlying knowledge bases. These datasets differ in construction methods (manual vs. automated), target knowledge graphs (Freebase [7], DBpedia [3], Wikidata [61], YAGO [53]), and reasoning complexity. Some emphasize compositional or multi-hop reasoning, while others focus on multilingual support, dialogue capabilities, or logical form mapping. Our review examines these datasets to understand their construction approaches, key features, and relevance for evaluating knowledge graph-enhanced question answering systems.

GraphQuestions [52] is constructed through a semi-automated pipeline that begins with the generation of structured queries over Freebase. These queries are designed to capture varied reasoning functions such as counting, superlatives, and conjunctions, as well as different structural complexities and answer cardinalities. Queries that are infrequent or low-quality are filtered using web statistics. The remaining set is then verbalized into natural language using pre-defined templates, creating a dataset well-suited for evaluating semantic parsing and logical compositionality.

WebQuestionsSP [70] builds upon the WebQuestions [6] dataset by providing SPARQL annotations aligned with Freebase. It employs a modular annotation interface that guides workers to select topic entities, predicates, and relevant constraints, ensuring consistent semantic representation. The dataset offers executable queries, enabling precise evaluation of models' ability to map natural language questions to structured representations.

ComplexWebQuestions [56] extends WebQuestionsSP by programmatically generating more complex SPARQL queries that incorporate logical constructs such as conjunctions, comparatives, and superlatives. These queries are automatically translated into machine-generated questions, which are then paraphrased into natural language by crowdworkers. The resulting dataset challenges QA models with deeper compositional reasoning and varied surface forms.

QALD-9 [37] is a multilingual benchmark constructed through manual curation, where annotators write natural language questions and align them with SPARQL queries over DBpedia. It emphasizes diversity in question types and supports multiple languages. Its extension, QALD-9-Plus, translates these questions into additional languages and maps them to Wikidata, enabling cross-lingual and cross-knowledge-base evaluation.

MetaQA [76] frames question answering as a latent variable problem, where both the topic entity and the reasoning path are unobserved. Built over a movie-related knowledge graph, the dataset includes 1-hop, 2-hop, and 3-hop questions, designed to evaluate multi-step reasoning. A variational neural framework is used to learn both entity disambiguation and graph-based reasoning simultaneously.

SimpleDBpediaQA [4] remaps the widely used SimpleQuestions dataset from Freebase to DBpedia. This conversion involves aligning entities and predicates via owl:sameAs links and rewriting SPARQL queries to account for conceptual mismatches such as directionality, ambiguity, and redirections. The resulting dataset enables QA over an actively maintained knowledge base while preserving the simplicity of the original benchmark.

CSQA [49] introduces a large-scale, multi-turn dialogue dataset for complex question answering over Wikidata. It combines crowd-sourced and in-house annotations to generate over 200K question-answer pairs, including clarification, comparison, and logical reasoning within dialogue context. CSQA is designed to benchmark conversational agents that require memory and contextual understanding.

LC-QuAD 1.0 and 2.0 [60, 14] are created through a three-stage semi-automated pipeline. First, SPARQL queries are instantiated using templates and selected entities. Second, these queries are mapped to question templates. Finally, crowdworkers paraphrase them into fluent natural language. The datasets support a wide range of question types, including boolean, temporal, compositional, and multi-relation queries, and target DBpedia and Wikidata respectively.

FreeBaseQA [23] compiles trivia-style factoid question–answer pairs from public sources and aligns them with Freebase triples using a two-way entity linking approach. Human annotators then validate

relevance and correctness. The dataset features over 54K entity-answer pairs covering 28K unique natural questions, offering high linguistic diversity and a challenging alternative to SimpleQuestions.

Compositional Freebase Questions [25] is developed to assess compositional generalization. It begins by generating logical forms and corresponding SPARQL queries using a unified grammar. Natural questions are written to express these logical forms. The dataset is then split using a divergence-based strategy to maximize the gap in compositional structure between training and test sets, enabling rigorous evaluation of generalization.

GrailQA [20] constructs question–answer pairs through a structured pipeline involving logical form generation over Freebase, expert-written canonical questions, and crowd-sourced paraphrases. It includes three evaluation splits: i.i.d., compositional, and zero-shot, making it suitable for analyzing generalization across different reasoning complexities and linguistic expressions.

QALD-9 Plus [46] enhances QALD-9 by adding more questions, refining SPARQL annotations, and expanding multilingual support. The updated version improves coverage of complex queries and diverse linguistic phenomena, making it more suitable for modern multilingual QA systems.

KQA Pro [8] is constructed by generating compositional reasoning programs (KoPL) and corresponding SPARQL queries over a curated knowledge base. These are paraphrased into natural questions via crowdsourcing. The dataset supports fine-grained evaluation across logical reasoning categories such as multi-hop inference, comparison, boolean logic, and temporal constraints.

Dynamic-KGQA [10] departs from static benchmarks by generating question–answer pairs on-the-fly. It samples compact, semantically coherent subgraphs from YAGO 4.5 [53] and uses LLMs to produce multi-hop questions grounded in these subgraphs. This design minimizes data leakage and supports controlled, reproducible QA benchmarking with adaptive complexity.

B.2 Recent KG-RAG Publications with WebQSP and/or CWQ

WebQSP [70] and CWQ [56] have emerged as the primary benchmarks for empirical evaluation in LLM-based knowledge graph question answering. Table 3 presents a chronological summary of recent LLM-based KGQA models, indicating for each whether WebQSP and CWQ were used for evaluation and providing a brief description of the model's main approach. Notably, nearly 30 KG-ARG models released between 2023 and 2025 have adopted one or both datasets as central components of their experimental protocols, despite the quality issues identified in Section 3.

Early LLM–KG hybrids, including ReaRev [34] and DECAF [71], set a precedent by reporting results on both datasets, but typically focused on Hit@1 as the main metric. As the field progressed, it became clear that Hit@1 alone could obscure over-generation and other qualitative issues. This recognition prompted a shift in the community: subsequent agent-style models such as ToG [55], RoG [32], ChatKBQA [31], and KD-CoT [63] began to report full precision, recall, and F1 scores on both benchmarks, enabling more nuanced and meaningful comparison. By 2024 and 2025, evaluation on WebQSP and CWQ had become an established standard for the field. State-of-the-art systems—such as GCR [33], Effi-QA [13], GNN-RAG [35], PoG [9], CLEAR-KGQA [65], and ReKnoS [64]—universally benchmarked on these datasets before extending to additional corpora or domain-specific tasks. Even in research targeting specialised knowledge graphs, models like RARoK [72] and Efficient-G-Retriever [51] included WebQSP or CWQ for calibration and comparability. As captured in Table 3, the adoption trajectory of these datasets not only reflects their ubiquity but also highlights their role in shaping rigorous and transparent evaluation practices for the next generation of KGQA systems.

B.3 Manual Inspection Protocol

We evaluate a broad selection of prominent KGQA benchmarks, including , and others commonly used in recent KGQA literature. Each dataset provides a set of natural language questions paired with ground-truth answers, and, where available, supporting triples or subgraphs from the underlying knowledge graph (e.g., Freebase, Wikidata, DBpedia, WikiMovies and some subsets).

For our systematic audit, we followed a unified sampling and review protocol. For WebQSP and CWQ—the most widely adopted benchmarks—we randomly sampled 100 and 300 test examples, respectively, to ensure sufficient coverage of prevalent error types. For all other datasets, we

Table 3: Chronological summary of recent LLM-based KGQA models, with dataset usage based on reported experiments (✓).

Author [Citation]	Model	WebQSP	CWQ	Year	Short Introduction
Mavromatis et al. [34]	ReaRev	✓	✓	2022	LLM + GNNs refine reasoning on incomplete graphs.
Yu et al. [71]	DECAF	✓	✓	2022	Joint answer/logical form decoding from free-text retrieval.
Sun et al. [55]	ToG	✓	✓	2023	LLM agent explores KGs via beam search for deep, interpretable reasoning.
Luo et al. [32]	RoG	✓	✓	2023	Relation-grounded KG paths guide LLM reasoning with explanations.
Luo et al. [31]	ChatKBQA	✓	✓	2023	LLM-generated logical forms, improved with KG retrieval.
Wang et al. [63]	KD-CoT	✓	✓	2023	External KG knowledge injected into CoT reasoning.
Liu et al. [30]	DualR	✓	✓	2024	GNN for structural reasoning, frozen LLM for semantic reasoning.
Luo et al. [33]	GCR	✓	✓	2024	KG-Trie constrains LLM decoding for logic-faithful KG reasoning.
Dong et al. [13]	Effi-QA	✓	✓	2024	Iterative LLM planning, KG exploration, and self-reflection for QA.
Mavromatis et al. [35]	GNN-RAG	✓	✓	2024	GNN-based subgraph reasoning with LLM in RAG pipeline.
Xu et al. [67]	READS	✓	✓	2024	LLM decomposes KGQA into retrieval, pruning, inference.
Li et al. [26]	DoG	✓	✓	2024	LLM generates "well-formed chains" via constrained decoding.
Fang et al. [17]	KARPA	✓	✓	2024	LLM pre-plans, matches KG paths, reasons in training-free manner.
Xu et al. [69]	GoG	✓	✓	2024	LLM agent selects, generates, reasons on incomplete KGs.
Zhan et al. [72]	RARoK	✓	✓	2024	RAG-augmented CoT for complex medical KGQA.
Li et al. [27]	SubgraphRAG	✓	✓	2024	MLP + triple-scoring for efficient subgraph extraction.
Fang et al. [16]	DARA	✓		2024	LLM decomposes and grounds formal KG queries.
Hu et al. [22]	GRAG	✓		2024	Text-to-graph, retrieves/prunes subgraphs for RAG.
Xiong et al. [66]	Interactive-KBQA	✓	✓	2024	LLM agent generates SPARQL via multi-turn KB interaction.
Dehghan et al. [12]	EWEK-QA	✓	✓	2024	Web retrieval + KG triple extraction for citation-based QA.
Chen et al. [9]	PoG	✓	✓	2024	Self-correcting LLM planner for decomposed KGQA.
Wen et al. [65]	CLEAR-KGQA	✓	✓	2025	Interactive clarification and Bayesian inference for ambiguity.
Tan et al. [57]	Path-Over-Graphs	✓	✓	2025	LLM agent explores/prunes multi-hop KG paths.
Wang et al. [64]	ReKnoS	✓	✓	2025	Aggregates "super-relations" for LLM forward/backward reasoning.
Xu et al. [68]	MemQ	✓	✓	2025	Memory module separates LLM reasoning from KG tool use.
Gao et al. [18]	FRAG	✓	✓	2025	Modular KG-RAG adapts retrieval to query complexity.
Shen et al. [50]	RwT	✓	✓	2025	LLM-guided MCTS refines KG reasoning chains.
Solanki et al. [51]	Efficient-G-Retriever	✓		2025	Attention-based subgraph retriever for LLM-aligned RAG.
Tang et al. [58]	GGI-MAB	✓	✓	2025	Multi-armed bandit adapts RAG retrieval for KGQA.
Zhang et al. [75]	TrustUGA	✓		2025	Unified Condition Graph, two-level LLM querying.

sampled 60 examples per set to enable a balanced cross-dataset comparison. Each sampled item was independently reviewed by two annotators with KGQA expertise, resolving disagreements through discussion.

During inspection, we assessed each example along three main dimensions: (1) factual correctness of the annotated answer; (2) clarity and appropriateness of the question; and (3) faithfulness of the supporting SPARQL, where available. We flagged instances with incorrect, incomplete, or ambiguous annotations, as well as questions that were underspecified, trivial, or unanswerable.

C Pitfalls of Existing KGQA Benchmarks

Many benchmarks are anchored to deprecated resources like Freebase, and even migration efforts to Wikidata [44] introduce further inconsistencies in entity mappings and answer verification. Most critically, almost all existing datasets are evaluated using rigid exact match (EM) metrics, which fail to recognize semantically equivalent answers phrased differently. In summary, current KGQA datasets

face two central challenges: (1) data quality issues, including inaccurate, incomplete, or artificial annotations, and (2) narrow evaluation protocols that do not capture true semantic correctness. These limitations underscore the need for new benchmarks—such as KGQAGen—that emphasize both annotation quality and robust, semantically-aware evaluation.

C.1 Current KGQA Dataset Issues

This appendix presents a detailed case study of data quality issues involved the most widely used KGQA benchmarks, drawing from our broader review of 16 major datasets. For each dataset, we randomly sampled question-answer pairs and performed careful manual verification following the evaluation criteria in Section B. By presenting both problematic examples and the reasoning behind their classification, this analysis provides concrete, case-based evidence that complements the aggregate statistics reported in Table 1 and discussed in Section 3.

Our review identifies three principal categories of data quality problems, as defined in Section 3.1: **Inaccurate Ground Truth Answers**, **Low-Quality or Ambiguous Questions**, and **Limitations of Exact-Match Evaluation**. These recurring issues—rooted in annotation, question design, and evaluation protocols—can seriously undermine the reliability of KGQA benchmarks. A comprehensive breakdown, along with additional annotated examples for each dataset, is provided in the supplementary materials at the shared repository[5].

C.1.1 Inaccurate Ground Truth Answers

Following the categorization presented in Section 3, we provide additional examples for each type of answer annotation error identified in our analysis: A major challenge in KGQA benchmarks is the prevalence of ground truth annotation errors, including incorrect, incomplete, or outdated answers. These undermine evaluation reliability and can mislead model training.

Incorrect annotations. Incorrect annotations occur when the labeled answer does not actually address the question or contains factual mistakes.

- **ID: WebQTest-273**
  Question: When did Michael Jordan return to the NBA?
  Answer: 1984
  Issue: 1984 is the year of his NBA debut, not his return. The correct answer should be 1995.
- **ID: CWQ-848_ac67410188d0f2258139a3c84773885e**
  Question: What is the time zone in the location where the time zone is Central Western?
  Answer: Parliamentary system, Constitutional monarchy, Federal monarchy
  Issue: The answers are not time zones but forms of government. This is an incorrect annotation, and the question is self-answering and non-informative.
- **ID: QALD9Plus-120**
  Question: Who is the daughter of Bill Clinton married to?
  Answer: Chelsea Clinton
  Issue: Answer is Chelsea Clinton herself, not her spouse. The correct answer should be Marc Mezvinsky.
- **ID: GrailQA-2101221015000**
  Question: The 1912–13 Scottish Cup season is part of what sports league championship?
  Answer: 1913 Scottish Cup Final
  Issue: The answer refers to a final match, not the correct league entity (“Scottish Cup”).
- **ID: DynamicKGQA-26720**
  Question: Which actor starred in both ’The Hospital’ and was born in the same country as Albert Einstein?
  Answer: George C. Scott
  Issue: George C. Scott was born in the United States, while Einstein was born in Germany. The answer does not satisfy the nationality constraint.

[5] https://drive.google.com/drive/folders/1hH-NxqbUkOSeLC3q1ifLpq6iOlByait4?usp=drive_link

Outdated answers. Outdated answers reflect facts that were once correct but have become obsolete due to real-world changes and key issue is the outdated knowledge source.

- **ID: WebQTest-182**
  Question: Who is Khloe Kardashian's husband?
  Answer: Lamar Odom
  Issue: The answer is outdated; Khloe Kardashian and Lamar Odom divorced in 2016.
- **ID: KQAPro-14**
  Question: What city's population is 6,690,432?
  Answer: Dalian
  Issue: The question lacks a specified time period, making the answer potentially outdated. Additionally, there is no country constraint, which may lead to ambiguity.
- **ID: FreebaseQA-eval-836**
  Question: Who is currently the creative director at the house of Chanel?
  Answer: Karl Lagerfeld
  Issue: Karl Lagerfeld passed away in 2019 and no longer holds this position.
- **ID: ComplexQuestions-33**
  Question: Who is Greece's leader now?
  Answer: Karolos Papoulias
  Issue: The answer is outdated; Karolos Papoulias served as president until 2015. The correct answer should be the current leader.
- **ID: MetaQA-46**
  Question: What were the release dates of films directed by the director of [Rosemary's Baby]?
  Answer: 1986, 1948, 1992, 1982, 1994, 1979, 1999, 1965, 1974, 1967, 1971, 2002, 1988, 2005, 1976, 2011, 2010, 2013
  Issue: The answer is incomplete and outdated; it does not include the director's most recent films, such as a 2023 release.

Incomplete annotations. Incomplete annotations occur when the gold set omits other valid answers, penalizing models that provide equally correct alternatives.

- **ID: GrailQA-2100689004000**
  Question: What fossil specimen dates from the Eocene?
  Answer: Darwinius masillae
  Issue: The annotation is incomplete; many Eocene fossils (e.g., Moeritherium lyonsi) would also be valid answers.
- **ID: DynamicKGQA-14927**
  Question: Which American professor at the University of Wisconsin–Madison worked in the same city where Charles V. Bardeen died?
  Answer: E. Ray Stevens
  Issue: The annotation is incomplete; any American professor at UW–Madison would satisfy the condition, not just E. Ray Stevens.
- **ID: DynamicKGQA-24330**
  Question: Which athlete from Novosibirsk shares their nationality with the owner of the United Shipbuilding Corporation?
  Answer: Yevgeni Nikolayevich Andreyev
  Issue: The annotation is incomplete; multiple Russian athletes from Novosibirsk could be correct.
- **ID: GraphQuestions-281000202**
  Question: The British coat of arms has which heraldic supporters included?
  Answer: Lion
  Issue: The annotation is incomplete; both a lion (left) and a unicorn (right) are supporters. Only listing "lion" is insufficient.
- **ID: KQAPro-72547**
  Question: What person has a notable work titled "The Scorpion King," which was produced by Vince McMahon?

Answer: Dwayne Johnson
Issue: Incomplete annotation; any individual involved in the production could be considered correct, not just Dwayne Johnson.

C.1.2 Low-Quality or Ambiguous Questions

Low-quality questions include those that are ambiguous, overly simple, or fundamentally unanswerable. We highlight key cases by dataset.

Ambiguous phrasing. Ambiguous questions make it unclear what the intended answer should be, undermining evaluation objectivity.

- **ID: CWQ-80_3dafd01d90b628c5913e0c64c1143354**
  Question: Who inspired the famous person who went to Willem II College?
  Answer: Eugène Delacroix, Claude Monet, Jean-François Millet, Jozef Israëls, etc.
  Issue: The query is open-ended; it does not specify which famous person.
- **ID: GrailQAPlus-2101990009000**
  Question: Which automotive designer designed NA?
  Answer: Koichi Hayashi, Tom Matano, Bob Hall
  Issue: The meaning of "NA" is ambiguous and could refer to multiple models or entities.
- **ID: GraphQuestions-358000401**
  Question: sci came after what video game engine?
  Answer: Adventure Game Interpreter
  Issue: The question is ambiguous as "sci" does not have a defined meaning or Wikidata label.
- **ID: GrailQA-3200295001000**
  Question: What is the number of theater plays that are in mystery?
  Answer: 1
  Issue: The question is too vague and does not provide enough context to determine the correct answer.
- **ID: WebQTest-108**
  Question: What was the book written by Charles Darwin?
  Answer (length: 153, unique: 94): On evolution, The Autobiography of Charles Darwin, The Voyage of the Beagle, The Origin of Species, ...
  Issue: The question is ambiguous—Darwin wrote many books, but the prompt refers to "the" book. The ground truth annotation is also excessively broad and noisy, containing 153 answers (including duplicates), which undermines reliable evaluation.

Low-complexity questions. Low-complexity questions can be solved by simple lookup or direct retrieval, offering little test of reasoning or multi-hop capabilities.

- **ID: QALD9Plus-143**
  Question: What is the area code of Berlin?
  Answer: 030
  Issue: This is a straightforward 1-hop fact lookup with no reasoning required.
- **ID: FreebaseQA-eval-1536**
  Question: Who wrote the 1970 book 'Future Shock'?
  Answer: Alvin Toffler
  Issue: This is a basic factoid query that tests only surface-level knowledge.
- **ID: FreebaseQA-eval-2363**
  Question: In which ocean is the island of Madeira?
  Answer: Atlantic Ocean
  Issue: This is a simple fact lookup with no reasoning challenge.
- **ID: CWQ-848_ac67410188d0f2258139a3c84773885e**
  Question: What is the time zone in the location where the time zone is Central Western?
  Answer: Parliamentary system, Constitutional monarchy, Federal monarchy
  Issue: The question is self-answering, non-informative, and offers little reasoning challenge.

- **ID: CSQA-7**
  Question: Which sex does Wolfgang Brandstetter belong to?
  Answer: male
  Issue: This is a 1-hop lookup question with no requirement for reasoning ability.
- **ID: SimpleDBpedia-17597**
  Question: What was Karl Dönitz's place of death?
  Answer: Schleswig Holstein
  Issue: This is a simple factual retrieval, requiring no reasoning.

Unanswerable, subjective, or ill-formed questions. Such questions may rest on false premises, omit crucial context, or rely on subjective interpretations.

- **ID: CWQ-2621_7360e892294860c6ef7ad9a10e540e1b**
  Question: Which author wrote editions of "Notes from My Travels 's husband"?
  Answer: Brad Pitt
  Issue: The question is ill-formed and refers to a non-existent book.
- **ID: CWQ-1304_f0c99f377c8b944364e85b70b9f9b331**
  Question: Which education institution is Bundesrealgymnasium Linz the leader of?
  Answer: Hitler Youth, Gestapo, Nazi Party, etc.
  Issue: Ill-formed query; Bundesrealgymnasium Linz is itself an educational institution, not a leader of other entities.
- **ID: GrailQA-2104467004000**
  Question: The time zone from UTC of 12.75 has been offset what number of times?
  Answer: 1
  Issue: The question is uninterpretable; UTC+12.75 is not a standard offset, and the phrasing lacks clarity.
- **ID: QALD9Plus-81**
  Question: Butch Otter is the governor of which U.S. state?
  Answer: [Missing Ground Truth]
  Issue: Unanswerable in the present; Butch Otter no longer holds that position.
- **ID: FreebaseQA-eval-3290**
  Question: What is measured in Scoville units?
  Answer: Pungency
  Issue: Subjective; the question could accept "spiciness" or "pungency," but only one is annotated as correct.
- **ID: GraphQuestions-462000201**
  Question: Find the bearers of the coat of arms granted by queen.
  Answer: Western Australia
  Issue: The question does not specify which queen or which coat of arms, making it ambiguous and unanswerable.
- **ID: KQAPro-3**
  Question: Which has lower elevation above sea level, Bristol or Jerusalem whose ISNI is 0000 0001 2158 6491?
  Answer: Bristol
  Issue: Problematic: the Jerusalem referenced is a musician, not a location. Multiple cities named Bristol exist, with no way to determine which is intended.

C.2 Limitations of Exact-Match Evaluation

Existing KGQA benchmarks are further limited by their reliance on rigid exact-match evaluation protocols. Such criteria do not accommodate semantically correct answers that are phrased differently from the annotated ground truth. As a result, models are often penalized for generating correct answers that differ only in surface form, leading to false negatives and an underestimation of true model performance.

- **ID: QALD9-174**
  Question: Who is the novelist of the work a song of ice and fire?

Ground Truth: George_R._R._Martin
Issue: Other semantically correct forms such as "George Raymond Richard Martin," "George R. R. Martin" (with or without punctuation), "G. R. R. Martin," "George RR Martin," "Martin, George R. R.," "George R. Martin," or "G.R.R. Martin" are equally valid. Exact-match evaluation penalizes correct answers that differ in surface form or formatting.

- **ID: QALD9-114**
  Question: How big is the earth's diameter?
  Ground Truth: 1.2742e+07
  Issue: Acceptable answers include "12,742 km," "12,742,000 meters," "about 7,918 miles," "1.2742×10^7 meters," and "approximately 12,700 kilometers." Variations in units, notation, or approximation are all reasonable, but exact match evaluation may reject them as incorrect.
- **ID: CSQA-60**
  Question: Which nucleic acid sequence encodes Ufm1-specific protease 2?
  Ground Truth: Ufsp2
  Issue: Other valid forms include "Ufsp2 gene," "the gene encoding Ufm1-specific protease 2," "gene symbol: UFSP2," or Ensembl/NCBI identifiers. Exact match allows only the annotated form, penalizing equally correct alternatives.
- **ID: WebQTest-6**
  Question: Where is JaMarcus Russell from?
  Ground Truth: Mobile
  Issue: Answers such as "Mobile, Alabama," "the city of Mobile," "Mobile (city)," "Mobile, AL," or "JaMarcus Russell was born in Mobile, Alabama" all convey the same information, but may not be accepted unless they match the ground truth exactly.
- **ID: FreebaseQA-eval-607**
  Question: Who wrote the 1990 Booker Prize winner Possession?
  Ground Truth: a. s. byatt
  Issue: Other correct answers like "Antonia Susan Byatt," "Dame A. S. Byatt," or "Antonia Byatt" are semantically equivalent, but only the annotated form is accepted under exact match.

D Prompt Templates and Generation Examples

This section provides detailed documentation of the prompt templates used in KGQAGen, along with representative examples of generated questions and error cases. We first present the core generator prompt that guides question construction and knowledge sufficiency checking. We then describe the simplified evaluator prompt used during answer validation. Finally, we analyze example outputs and common error patterns to illustrate the framework's capabilities and limitations.

D.1 Question Generation Prompt

The generator component of KGQAGen utilizes a carefully designed prompt template to ensure high-quality question generation. The prompt consists of input specifications and structured generation rules that guide the LLM in producing well-formed question-answer instances.

The input format specifies RDF triples from Wikidata, where each triple contains an entity label with its Q-ID, a predicate label with its P-ID, and another entity label with Q-ID. The LLM evaluates whether the given subgraph provides sufficient information for generating a meaningful KGQA instance. The generation rules enforce five key requirements for question construction. (1) Reasoning complexity demands that generated questions involve at least 2-hop logical reasoning paths. The framework prioritizes specific, instance-level entities while avoiding generic categories. Factual constraints are incorporated only when necessary for disambiguation or meaningful answer space reduction. (2) Entity selection criteria require that candidate entities must be concrete instances rather than abstract types. The system favors entities that enable meaningful multi-hop paths through affiliations, awards, locations, or temporal relationships, while avoiding generic class-level concepts. (3) Question difficulty ensures that questions are designed to require structured knowledge graph reasoning by incorporating inverse relations, numerical constraints, comparative logic, or set-based conditions. The framework employs factual filters to reduce ambiguity and ensures questions cannot be answered through general knowledge alone. (4) Natural language quality mandates that questions

must use natural, fluent phrasing typical of real user queries. The prompt enforces self-contained and concise formulation while avoiding references to underlying data structures or unnecessary repetition. (5) Semantic clarity requires that all generated questions must unambiguously specify their intended answers. The prompt explicitly prohibits vague or underspecified formulations that could lead to multiple valid interpretations.

The LLM returns a JSON response indicating either insufficiency with candidate entities for expansion or a complete question-answer instance containing the natural language question, answer set, supporting proof triples, and corresponding logical constraints. The full prompt template is provided below:

Prompt

```
You are given a small set of RDF triples from Wikidata.
Format: Each triple is a 3-item array: ["<label> (<Q-ID>)", "<predicate>
(<P-ID>)", "<label> (<Q-ID>)"]
Triples: {triples}
Your task is to determine whether this subgraph is sufficient to support a
challenging and non-trivial question for a knowledge graph question answering
(KGQA) benchmark.
Guidelines:
1. Reasoning Depth
   • Prefer questions requiring at least 2-hop reasoning.
   • Avoid generic topics or subclass chains-focus on instance-level,
     specific entities.
   • Use factual constraints (e.g., date, affiliation) only when needed to
     disambiguate the answer or add meaningful specificity.
   • Do not over-constrain-include only what is necessary to yield a specific
     answer.
2. Entity Selection and Expansion
   • Focus on concrete, instance-level entities (e.g., Q7186), not types like
     Q5 (human) or Q11424 (film).
   • Avoid generic classes like ‘‘scientist’’, ‘‘award’’, or ‘‘event’’, and
     relations like ‘‘subclass of’’ or ‘‘instance of’’.
   • Prefer entities and paths supporting deeper reasoning-e.g., affiliations,
     recognitions, or spatiotemporal links.
3. Difficulty
   • Encourage inverse relations, comparative logic, date/number filters, or
     set membership.
   • Ensure the answer cannot be derived from general knowledge alone.
   • The subgraph must contain all supporting information to answer the
     question.
4. Naturalness
   • Phrase the question as a fluent, self-contained query a user might ask.
   • Avoid references to the input format (e.g., "triples", "given data").
   • Do not use phrases like:
     – "from the given data"
     – "among these entities"
     – "listed here"
5. Clarity
   • The question must be unambiguous and logically imply a unique, specific
     answer.
   • Avoid vague or underspecified language.
```

```
Output Format:
If the graph is not sufficient, return: { "sufficient": false, "candidate":
[<QID>, ..., <QID>] }
If sufficient, return: { "sufficient": true, "question": "<natural-language
question>", "answer": ["<answer-label (QID)>", "..."], "proof": [ ["<label
(QID)>", "<predicate (PID)>", "<label (QID)>"], ... ] }
Return strict JSON only - no commentary.
```

Few-shot Example

```
Example 1: Triples: [ ["Johann Martin Schleyer (Q12712)", "nominated for
(P1411)", "Nobel Peace Prize (Q35637)"], ["International VolapÕ0fck Academy
(Q3358168)", "founded by (P112)", "Johann Martin Schleyer (Q12712)"], ["Johann
Martin Schleyer (Q12712)", "place of birth (P19)", "Oberlauda (Q885402)"] ]
Output: { "sufficient": true, "question": "Who among the nominees for the
Nobel Peace Prize was also the founder of International VolapÕ0fck Academy?",
"answer": ["Johann Martin Schleyer (Q12712)"], "proof": [ ["Johann Martin
Schleyer (Q12712)", "nominated for (P1411)", "Nobel Peace Prize (Q35637)"],
["International VolapÕ0fck Academy (Q3358168)", "founded by (P112)", "Johann
Martin Schleyer (Q12712)"] ] }

Example 2: Triples: [ ["Karakalpakstan (Q484245)", "capital (P36)",
"Nukus (Q489898)"], ["Karakalpakstan (Q484245)", "shares border with (P47)",
"Mangystau Region (Q238931)"], ["Karakalpakstan (Q484245)", "official language
(P37)", "Karakalpak (Q33541)"], ["Karakalpakstan (Q484245)", "country (P17)",
"Uzbekistan (Q265)"] ]
Output: { "sufficient": false, "candidate": ["Q33541", "Q489898", "Q238931"]
}

Example 3: Triples: [ ["Astronomy and Astrophysics (Q752075)", "publisher
(P123)", "EDP Sciences (Q114404)"], ["Astronomy and Astrophysics (Q752075)",
"editor (P98)", "Thierry Forveille (Q46260676)"], ["Zeitschrift für Astrophysik
(Q3575110)", "followed by (P156)", "Astronomy and Astrophysics (Q752075)"] ]
Output: { "sufficient": true, "question": "What astronomical journal,
published by EDP Sciences and edited by Thierry Forveille, succeeded Zeitschrift
für Astrophysik as its immediate follower?", "answer": ["Astronomy and
Astrophysics (Q752075)"], "proof": [ ["Astronomy and Astrophysics (Q752075)",
"publisher (P123)", "EDP Sciences (Q114404)"], ["Astronomy and Astrophysics
(Q752075)", "editor (P98)", "Thierry Forveille (Q46260676)"], ["Zeitschrift
für Astrophysik (Q3575110)", "followed by (P156)", "Astronomy and Astrophysics
(Q752075)"] ] }
```

This structured prompt design ensures consistent generation of high-quality, diverse, and well-specified question-answer pairs for the benchmark dataset. The prompt template balances multiple objectives: maintaining reasoning complexity, ensuring natural language quality, and guaranteeing answer specificity. By enforcing these requirements through explicit rules and format specifications, we enable systematic generation of challenging yet well-formed KGQA instances.

D.2 SPARQL Validation Prompt

The validation component uses a focused prompt template for verifying and refining SPARQL queries. This streamlined prompt specifically addresses query correction when execution results differ from intended answers, ensuring both syntactic correctness and semantic alignment.

The validation process operates on the principle of iterative refinement. When a generated SPARQL query fails to return expected results or returns empty result sets, the validation component engages a lightweight language model to diagnose and correct the query. This approach recognizes that initial query generation may suffer from syntactic errors, incorrect entity identifiers, or inappropriate query structure that prevents successful execution against the Wikidata endpoint.

The prompt template emphasizes simplicity and executability in query revision. Rather than attempting complex query transformations, the validation component focuses on ensuring that revised queries use only essential triple patterns and avoid unnecessary complexity that might introduce additional

failure points. The template specifically discourages the use of optional clauses and filter conditions unless they are strictly necessary for answering the question, as these constructs often lead to query execution failures or unexpected empty results. Additionally, the validation process enforces structural requirements that ensure compatibility with the Wikidata query service. All revised queries must terminate with a single SERVICE wikibase:label clause to retrieve human-readable English labels for entities, maintaining consistency with the expected output format. The prompt also mandates syntactic validity and direct executability at the official Wikidata SPARQL endpoint, ensuring that corrected queries can be verified immediately. The output format maintains strict JSON formatting requirements to facilitate automated processing. The validation component returns only the corrected SPARQL query without additional commentary or explanation, enabling seamless integration into the broader validation pipeline. This focused approach allows for rapid iteration and correction when initial query generation produces non-executable or semantically misaligned queries.

Through this validation mechanism, KGQAGen ensures that all retained question-answer pairs are grounded in verifiable SPARQL queries that can be executed against the knowledge base. This constraint provides a strong guarantee of answer correctness and enables ongoing validation as the underlying knowledge graph evolves over time.

Prompt

```
You are given a SPARQL query over Wikidata that returned no results.
Question: {question}
Original SPARQL: {sparql}
Your task is to revise the query so that it returns valid results from Wikidata.
Revision Guidelines:
• Use only essential triple patterns. Avoid OPTIONAL and FILTER clauses unless
  strictly necessary.
• The query must end with a single SERVICE wikibase:label clause to retrieve
  English labels.
• Ensure the query is syntactically valid and directly executable at https:
  //query.wikidata.org.
Output Format: Return a single JSON object in the exact format below - no
commentary, no markdown:
{
  "correct_sparql": "<REVISED SPARQL QUERY HERE>"
}
```

This structured prompt design ensures consistent generation of high-quality, diverse, and well-specified question-answer pairs for the benchmark dataset. The prompt template balances multiple objectives by maintaining reasoning complexity, ensuring natural language quality, and guaranteeing answer specificity. By enforcing these requirements through explicit rules and format specifications, we enable systematic generation of challenging yet well-formed KGQA instances.

E A Case Study of `KGQAGen-10k`

An audit of 300 randomly sampled question–answer pairs from the entire 10,787-instance `KGQAGen-10k` revealed 11 defective cases shown Table 4, a rate of error of 3.6%. Although this figure is relatively low, these instances expose recurring weaknesses in the generation and verification pipeline that warrant attention. The issues fall into three broad categories: self-answering prompts, hallucinated or incomplete relations, and errors inherited from the source knowledge graph.

The first issue, **self-answering questions**, was evident in items 4555 and 6931, where the question text directly includes the target answer. For example, asking about a subclass that 'has the same meaning as the English-language word 'city' leaves little ambiguity about the expected answer. Because our current verification process only checks that the SPARQL query returns at least one result overlapping the proposed answer set, it overlooks this form of lexical leakage and accepts the examples as valid.

The second and more prevalent category involves **hallucinated or incomplete knowledge**. In six cases (IDs 8318, 1105, 1164, 1529, 1825, and 10469), the model generated questions based on nonexistent or incoherent relationships, such as attributing architectural roles to historical political

figures. In these situations, the LLM still produces syntactically valid queries, sometimes by leveraging loosely related property paths, allowing the verifier's overlap check to pass despite clear semantic errors. In a complementary failure mode, item 2297 exemplifies incomplete annotations: While multiple mathematicians satisfy the described criteria, only one is listed in the answer set. Since the verifier stops when it finds a match, it does not detect incompleteness.

A third source of error originates not from the generation process but from the **underlying knowledge graph itself in Wikidata [61]**. Items 9572 and 2046 illustrate this point: one references an entity mistyped as a product, the other includes an unlabeled identifier. Our pipeline implicitly treats Wikidata typing and labeling as authoritative, so these issues remain undetected unless caught during manual review.

The major cause of these verification failures lies in the limited scope of the current safeguard. The answer veridation and refinement (detailed in Section 4.3) of our `KGQAGen` checks are whether the SPARQL query compiles and whether its result set overlaps the proposed answer. Although effective in filtering out broken or irrelevant queries, this approach does not account for key semantic and structural issues. It does not detect leakage of lexical answers, does not require precise predicate alignment, does not check the joint coherence of query constraints, and does not validate type or label accuracy within the knowledge graph.

To address these gaps and further improve the quality of the data set beyond the current validation rate of 96.3%, we plan a series of targeted upgrades. First, we will implement a lexical filter to reject questions that contain their own answers. Second, we will enforce stricter predicate and type constraints within the generated queries to check against hallucinated relations. Third, we will introduce answer completeness audits through closure tests that ensure the full set of valid answers is captured. Finally, we will cross-check critical entity labels and types against alternative KG snapshots to catch inconsistencies and improve robustness across knowledge versions.

Table 4: KGQAGen-10k Error Analysis: Each case is displayed with concise issue diagnosis.

Field	Content
ID	4555
Question	What subclass of 'city or town' has both a GND ID and is identified as the same concept as the English-language word 'city'?
Answer	city
Issue	**Self-answering:** The question is trivial; the answer is explicitly stated in the question itself.
ID	8318
Question	Who is the architect of Estadio Nacional de Costa Rica that was also a successful candidate in multiple Costa Rican general elections?
Answer	Ricardo Jiménez Oreamuno
Issue	**Hallucinated knowledge:** The question implies an architect relationship that does not exist; Ricardo Jiménez Oreamuno was a politician, not an architect.
ID	1105
Question	Which person who has been an owner of the Shroud of Turin is not a human individual, but rather a dynastic house?
Answer	Geoffroi de Charny, House of Savoy, Jeanne de Vergy, pope
Issue	**Hallucinated knowledge:** Geoffroi de Charny and House of Savoy are individuals, not dynastic houses.
ID	1164
Question	Which person, who held citizenship in the Ming, Qing, and short-lived Zhou dynasties, was both the father of Wu Yingxiong and the spouse of both Chen Yuanyuan and Empress Zhang, and led a revolt known as the Revolt of the Three Feudatories?
Answer	Kangxi Emperor, Kingdom of Tungning, Qing dynasty, Wu Sangui, Zheng Jing
Issue	**Hallucinated knowledge:** Zheng Jing is not the spouse of Chen Yuanyuan.
ID	2297
Question	Which mathematician both lent his name to the principle maintained by WikiProject Mathematics and is explicitly named as its discoverer or inventor?
Answer	Johann Peter Gustav Lejeune Dirichlet
Issue	**Hallucinated knowledge:** Many mathematicians have principles named after them; the annotation is incomplete and not unique.
ID	1825
Question	Which concept, characterized as 'unusual', is named after 'unusual', is the main subject of an entity named after 'unusual', and is also cited as a partially coincident concept by 'rarity'?
Answer	frequency, scarcity, unusual
Issue	**Hallucinated knowledge:** None of the ground truths are the subject of "World's Weirdest Animals". Its main subject is "creature".
ID	10469
Question	Which musical artist is associated with the genre that is said to be the same as "vintage" and has also performed in the genre represented by 'retro style'?
Answer	Adam Tsarouchis, Ahmad Bersaudara, Anna Jantar, Gloomwood, Nina Shatskaya, Type-B, VCTR-SCTR, Wieczór na dworcu w Kansas City
Issue	**Hallucinated knowledge:** Wieczór na dworcu w Kansas City is not a musical artist, but a retro style song.
ID	1529
Question	Which artist created a work that depicts the astronomical event discovered by Pierre Gassendi, and has as its main subject the transit of Mercury?
Answer	Mercury Passing Before the Sun
Issue	**Hallucinated knowledge:** Mercury Passing Before the Sun is the artwork, not the artist. The artist is Giacomo Balla.
ID	6931
Question	Which type of underwater vehicle is classified as both a subclass of submersible and shares this property with bathyscaphe, narco-submarine, and Osprey-class submersible?
Answer	Osprey-class submersible, bathyscaphe, bathysphere, narco-submarine, submersible drilling rig
Issue	**Self-answering:** The answer is trivially the same as the question's subject; provides no substantive challenge.
ID	9572
Question	Which company has produced a product used for flow measurement that includes a flow meter as a part?
Answer	Sage Metering
Issue	**Wikidata Mislabel:** The product of Sage Metering is labelled as "flow measurement" on Wikidata, but "flow measurement" is a task, not a product.
ID	2046
Question	Which tool that is a subclass of both 'physical tool' and is connected with 'level staff', is in turn a subclass of something that has the shape of a cylinder and is different from 'Rod'?
Answer	"Q9397141"
Issue	**Wikidata Mislabel:** The entity "Q9397141" doesn't contain the natural language label.

F Experimental Details

This section provides comprehensive details about our experimental setup, including model configurations, training protocols, and evaluation procedures.

F.1 Model Specifications

We evaluate three categories of models on `KGQAGen-10k`: (1) **Pure Language Models**, (2) **KG-RAG Systems**, and (3) **LLMs with Supporting Graphs**. These categories reflect increasing levels of external knowledge integration, ranging from purely parametric reasoning to symbolic augmentation and perfect-evidence setups.

Pure Language Models These models rely entirely on their internal parametric memory and are evaluated in a zero-shot setting, without any KG subgraph or retrieval. Their performance reflects inherent reasoning capability, factual coverage, and generalization.

- **LLaMA-3.1-8B-Instruct** [19]: An 8B instruction-tuned model from Meta's LLaMA 3.1 series. It is optimized for following task-specific instructions and shows improved reasoning performance compared to earlier LLaMA versions.
- **LLaMA2-7B** [59] : A general-purpose 7B model trained on publicly available data, serving as a foundational open-weight baseline for reasoning without instruction tuning.
- **Mistral-7B-Instruct-v0.2** [2]: An instruction-following model based on Mistral-7B with a 32k context window and standard attention. It is designed for accurate and efficient long-context reasoning.
- **GPT-4o-mini** [1]: A compact version of GPT-4o offering reduced latency and strong language understanding performance, suitable for real-time applications.
- **GPT-4** [1]: OpenAI's flagship model known for robust multi-step reasoning, long-context understanding, and generalization across a wide array of tasks.
- **DeepSeek-Chat** [11]: A dialogue-oriented LLM developed by DeepSeek and fine-tuned for task completion and conversational fluency aligned with human feedback.
- **GPT-4o** [1]: A unified multimodal model capable of handling text, image, and audio inputs. We use it in a text-only setup to assess its advanced reasoning capabilities.
- **GPT-4.1** [39]: An updated variant of GPT-4 that improves long-context performance, factual grounding, and consistency in complex prompt execution.

KG-RAG Systems Knowledge Graph Retrieval-Augmented Generation (KG-RAG) systems incorporate structured symbolic evidence from a KG to assist reasoning and answer generation. These models access retrieved subgraphs at runtime and vary in how they integrate retrieved content—either as conditioning input or through decoding constraints.

- **RoG (LLaMA2-7B)** [32]: Fine-tuned on `KGQAGen-10k`'s training split, using annotated supporting subgraphs to supervise faithful reasoning path generation for answer and explanation prediction.
- **GCR (LLaMA-3.1 + GPT-4o)** [33]: Fine-tuned its path generation module with supporting subgraphs, leveraging a KG-Trie to constrain decoding and using GPT-4o for final answer synthesis.
- **ToG (GPT-4o)** [55]: Adapted to Wikidata by replacing Freebase API calls with SPARQL queries and evaluated zero-shot, interactively exploring the KG and generating answers from the retrieved subgraph without fine-tuning or parameter adjustment.
- **PoG (GPT-4o)** [9]: Applied as a zero-shot prompting-based agent that dynamically decomposes, explores, and self-corrects over Wikidata for each test question, without any dataset-specific fine-tuning.

LLM with Supporting Subgraph To estimate the upper bound of KG-augmented QA performance, we provide models with the gold supporting subgraph used during data generation. This simulates a perfect-retrieval setting, where the model receives all and only the minimal evidence needed to answer correctly. These experiments assess whether models can effectively reason over structured KG input when retrieval is assumed to be ideal.

- **LLaMA2-7B (w/ SP)** [59]: The model is provided with the gold subgraph and asked to generate the answer. This tests the reasoning capacity of a smaller open-weight model under ideal symbolic input.
- **GPT-4o (w/ SP)** [1]: The same setup as above, but with GPT-4o as the base model. This configuration reflects an upper-bound for KG-RAG systems when both retrieval and reasoning are ideal.

F.2 Beyond Exact Match: Introducing LASM

While the limitations of exact match evaluation in KGQA are well-recognized [48, 62], few works have proposed principled solutions. To address this gap, we introduce LLM-Assisted Semantic Match (LASM), a novel evaluation scheme that goes beyond surface-level equivalence by leveraging the semantic understanding capabilities of large language models.

The core idea of LASM is to use an LLM verifier to assess semantic similarity between predicted and ground truth answers. When a model's prediction fails the exact string match, LASM invokes a GPT-4o-mini judge to determine whether the prediction is semantically equivalent to the gold answer. This approach enables LASM to properly credit models for generating meaningfully correct responses that traditional metrics would overlook due to syntactic or lexical variation. To quantify the impact of LASM, we compare model performance on the FreebaseQA dataset [23] under both exact match and LASM evaluation. As shown in Table 5, LASM yields substantial improvements across all key metrics, including accuracy (+5.3%), Hit@1 (+5.3%), and F1 (+5.0%). These gains demonstrate the effectiveness of semantic matching in capturing valid model predictions that exact match misses.

Scoring	Accuracy	Hit@1	F1	Precision	Recall
Exact Match	90.39	90.39	88.08	87.08	90.39
LASM	95.72	95.67	93.12	92.04	95.65

Table 5: FreeBaseQA results with GPT-4o. LASM consistently recovers semantically correct predictions missed by exact match, leading to substantial metric improvements.

Beyond offering a more robust and nuanced assessment of model outputs, LASM has important implications for the development and evaluation of KGQA systems. By rewarding models for semantic correctness rather than rigid string matching, LASM promotes the development of systems that prioritize meaning over surface form. Moreover, as a fully automated method that does not rely on dataset-specific rules or annotations, LASM is readily applicable to any KGQA benchmark, enabling more meaningful cross-dataset comparisons.

In summary, LASM represents a principled and generalizable approach to overcoming the limitations of traditional exact match evaluation in KGQA. By incorporating semantic awareness through LLM-based similarity judgments, LASM provides a more reliable and nuanced assessment of model performance, paving the way for the development of more robust question answering systems. As we will demonstrate through extensive experiments in Section 5, LASM offers a valuable tool for evaluating and advancing the state of the art in KGQA.

F.3 Experimental Setup

We evaluate all models on KGQAGen-10k using a standardized split of 8,629/ 1,079/1,079 train/eval/test. For KG-RAG systems, we adapt each model to work with Wikidata by replacing their original knowledge base interfaces with SPARQL queries to the Wikidata endpoint.

Training and Inference Protocols RoG [32] employs a planning-retrieval-reasoning pipeline where LLaMA2-7B first generates candidate relation paths, which are then matched against the

knowledge graph using constrained breadth-first search. The retrieved reasoning paths, combined with the original question, guide the model to generate both answers and explanations. We fine-tune the entire pipeline on our training split, using the supporting subgraphs from dataset construction as supervision for faithful path generation.

GCR [33] enforces graph faithfulness through constrained decoding. Prior to inference, we construct a KG-Trie index that efficiently captures all valid reasoning paths within a fixed hop limit. During generation, a fine-tuned LLaMA-3.1 model produces candidate paths under strict KG-Trie constraints, ensuring only valid graph traversals. These candidates are then passed to GPT-4o for inductive reasoning and answer synthesis. Similar to RoG, we leverage our supporting subgraphs for training the path generation component.

In contrast, ToG [55] and PoG [9] operate without fine-tuning, treating the LLM as an agent that interactively explores the knowledge graph. ToG constructs reasoning trees by iteratively selecting relations and entities based on question semantics, while PoG enhances this with adaptive planning and self-correction mechanisms. For both models, we implement direct Wikidata integration, allowing them to dynamically query the knowledge base during inference without dataset-specific training.

Evaluation Metrics We evaluate each KGQA system using 4 complementary Hit@1, Precision, Recall, and F1—under two answer-matching schemes: Exact Match (EM) and LASM. **EM** considers a prediction correct only if the model's answer set exactly matches the ground-truth set after basic normalization, which includes lowercasing and alphabetically sorting answers. This is a strict string-level comparison that does not account for synonyms, paraphrases, or other forms of semantic equivalence. Hit@1 measures whether the model's top-ranked answer appears in the ground-truth set. Precision, Recall, and F1 capture the degree of set overlap: Precision reflects the proportion of predicted answers that are correct, Recall captures the proportion of ground-truth answers that are retrieved, and F1 is their harmonic mean—together highlighting whether a model tends to over- or under-generate. **LASM** extends this evaluation by replacing literal comparison with a GPT-4o-mini verifier that determines whether the predicted and ground-truth answers are semantically equivalent. We then recompute all five metrics based on this semantic agreement. This two-tiered protocol offers a comprehensive view of model performance, balancing surface-level exactness with meaning-level correctness.