arXiv:2506.00789v1 [cs.CL] 1 Jun 2025

RARE: Retrieval-Aware Robustness Evaluation for Retrieval-Augmented Generation Systems

Yixiao Zeng
Carnegie Mellon University
jackz@andrew.cmu.edu

Tianyu Cao
Carnegie Mellon University
tianyuca@andrew.cmu.edu

Danqing Wang
Carnegie Mellon University
danqingw@andrew.cmu.edu

Xinran Zhao
Carnegie Mellon University
xinranz3@andrew.cmu.edu

Zimeng Qiu
Amazon
zimengqi@amazon.com

Morteza Ziyadi
Amazon
mziyadi@amazon.com

Tongshuang Wu
Carnegie Mellon University
sherryw@andrew.cmu.edu

Lei Li
Carnegie Mellon University
leili@cs.cmu.edu

Abstract

Retrieval-Augmented Generation (RAG) enhances recency and factuality in answers. However, existing evaluations rarely test how well these systems cope with real-world noise, conflicting between internal and external retrieved contexts, or fast-changing facts. We introduce **R**etrieval-**A**ware **R**obustness **E**valuation (*RARE*), a unified framework and large-scale benchmark that jointly stress-tests query and document perturbations over dynamic, time-sensitive corpora. One of the central features of *RARE* is a knowledge-graph-driven synthesis pipeline (*RARE*-Get) that automatically extracts single and multi-hop relations from the customized corpus and generates multi-level question sets without manual intervention. Leveraging this pipeline, we construct a dataset (*RARE*-Set) spanning 400 expert-level time-sensitive finance, economics, and policy documents and 48,322 questions whose distribution evolves as the underlying sources change. To quantify resilience, we formalize retrieval-conditioned robustness metrics (*RARE*-Met) that capture a model's ability to remain correct or recover when queries, documents, or real-world retrieval results are systematically altered. Our results show that RAG systems exhibit surprising vulnerability to perturbations, with document robustness consistently being the weakest point regardless of generator size or architecture. RAG systems consistently show lower robustness on multi-hop queries than single-hop queries across all domains. Code and full dataset are available on GitHub and HuggingFace.

1 Introduction

Retrieval-Augmented Generation (RAG) significantly enhances large language models (LLM) by integrating external knowledge sources, allowing the generation of accurate and contextually rich responses [Gao et al., 2024]. However, the robustness of RAG systems remains inadequately evaluated. Current benchmarks predominantly rely on static, time-invariant datasets with general-knowledge or common-sense queries. Such benchmarks inadvertently favor models that rely on memorization rather than genuine retrieval and synthesis of novel, timely information [Xu et al., 2024]. Consequently, existing assessments yield overly optimistic performance measures, overlooking critical real-world scenarios involving dynamic, specialized, and complex information.

Preprint. Under review.

An ideal robustness evaluation benchmark for RAG robustness must address several critical dimensions simultaneously, emphasizing **temporal dynamics**, **query complexity**, and **content specialization**. Temporal dynamics is crucial to reflect real-world scenarios where information evolves rapidly[Meem et al., 2024, Jang et al., 2022], particularly in domains such as finance [Shen and Kurshan, 2023]. Such time-sensitive datasets prevent contamination from memorized responses and necessitate continuous adaptation by RAG systems. Query complexity, especially multi-hop scenarios that require complex reasoning and integration across multiple retrieved documents [Yang et al., 2018, Geva et al., 2021]. Most existing multi-hop benchmarks require substantial human efforts, which makes it impossible to curate large-scale extensive datasets. Therefore, automation is essential and advanced techniques such as knowledge graphs (KGs) [Schneider et al., 2022] can be used. Moreover, with widespread integration into real-world applications, benchmarks must emphasize content specialization, including professional and domain-specific contexts that challenge models with intricate terminology and nuanced interpretations.

Additionally, most RAG research has focused on ideal conditions, with limited attention to how these systems perform when faced with noisy or imperfect inputs. In real-world applications, an RAG system usually should contend with perturbed queries containing typos, irrelevant information, or ambiguous phrasing [Zhang et al., 2025b]. Retrieved document may also be noisy, partially relevant, or even contradictory [Chen et al., 2023]. A truly robust RAG system should maintain robust performance despite these challenges.

In this paper, we introduce a comprehensive **R**etrieval-**A**ware **R**obustness **E**valuation (*RARE*) framework. It includes: ***RARE*-Met**: a comprehensive robustness evaluation metric for measuring RAG system performance under perturbations to queries, documents, and simulated real-world retrieval results, providing diagnostic insights into current system limitations. ***RARE*-Get**: a novel dynamic synthesis pipeline that automatically constructs time-sensitive evaluation data through knowledge graph triplet extraction and traversal techniques, enabling the creation of single-hop and multi-hop tuples (question, answer, ground truth chunks) at various complexity levels without manual curation. ***RARE*-Set**: a large-scale benchmark comprising over 400 specialized documents and 48,322 queries across financial, economics, and policy domains - sectors where information accuracy and timeliness are particularly critical yet underrepresented in existing benchmarks. Unlike previous datasets dominated by general knowledge questions, our benchmark exclusively focuses on domain-specific, technical queries that require advanced information synthesis. Our dataset features diverse query patterns generated through knowledge graph traversal, including single-hop, multi-hop chained, star-shaped, and inverted-star-shaped, with systematic perturbations at both surface and semantic levels to comprehensively assess robustness under realistic conditions.

Our evaluation reveals that RAG systems are fragile under document perturbations, no matter the generator's size or architecture. Robustness scores do not always scale strictly with model size - some mid-sized generators outperform several larger counterparts. Also, the robustness of RAG systems across different domains is different, and multi-hop queries prove less robust than single-hop queries. All of these indicate the importance of evaluating and improving the robustness of RAG systems.

2 Related Work

Time-Sensitive Benchmark Recent time-sensitive benchmark initiatives address LLM knowledge outdating through distinct approaches. FreshQA [Vu et al., 2024] tests reasoning over up-to-date knowledge with a dynamic updated QA benchmark and evaluation methodology for correctness and hallucination detection. PAT-Questions [Meem et al., 2024] introduces a self-updating benchmark for present-anchored temporal questions using SPARQL queries over Wikidata to automatically refresh answers. RealtimeQA [Kasai et al., 2024] employs a weekly dynamic platform that extracts questions form news quizzes, challenging systems to answer questions about current events. Existing benchmarks often exhibit limitations such as narrow raw data domains (primarily Wikipedia or news articles), a restricted number of test cases due to the reliance on human-generated questions, and a prevalence of queries that can be accurately answered by the language model alone—without the need for retrieval—such as general-domain fact questions.

Multi-Hop QA and RAG Benchmark Early knowledge-intensive benchmarks like Natural Questions [Kwiatkowski et al., 2019] and HotpotQA [Yang et al., 2018] established foundations for factual question answering but lacked cross-document reasoning and overlapping with popular training

Table 1: Comparison of our proposed dataset with prior time-sensitive, multi-hop/RAG, and robustness benchmarks. Symbols: ✓ = yes/present; ✗ = not available; "partial" = feature applies to only a subset; "-" = not applicable; MH = Multi Hop question.

Dataset	Year	# QA	Data Sources	Unique	Time-Sens.	MH	Dynamic	Automatic
Time-Sens. Benchmarks								
RealtimeQA	2023	2340	News	✓	✓	✓	✓	partial
FreshQA	2024	600	Search engine	✓	✓	✓	✓	partial
PAT-Questions	2024	6172	Wikipedia	partial	✓	✓	✓	✓
MH & RAG Benchmarks								
Natural Questions	2019	100 k	Wikipedia	✗	✗	✗	✗	✗
HotpotQA	2018	97.9 k	Wikipedia	✓	✗	✓	✗	✗
MuSiQue-Ans	2022	50 k	Wikipedia	✗	✗	✓	✗	partial
StrategyQA	2021	2780	Wikipedia	✓	✗	✓	✗	✗
MultiHop-RAG	2024	2506	News	✓	✓	✓	✗	✓
RAGBench	2024	100 k	Domain-specific	✗	✗	✓	✗	✓
CRAG	2024	4409	Search engine	✗	✓	✓	✗	partial
LLM Robust Benchmarks								
KaRR	2023	-	T-REx (Wikipedia)	partial	✗	✗	✗	partial
QE-RAG	2025	51 k	Wiki + Domain-specific	partial	✗	✓	✗	✓
SURE	2025	-	NQ-open (Wikipedia)	✗	✗	✗	✗	✓
RARE (Ours)	2025	48.3 k	Domain-specific reports	✓	✓	✓	✓	✓

dataset. Later development such as MuSiQue [Trivedi et al., 2022] and StrategyQA [Geva et al., 2021] advanced multi-hop QA capabilities but remained confined to Wikipedia sources. MultiHop-RAG [Tang and Yang, 2024] expanded to news domain with 2-4 ho queries but lacks dynamic real-time updates. RAGBench [Friel et al., 2025] introduced evaluation across industry corpora with new faithfulness metrics, with CRAG [Yang et al., 2024] targets dynamic performance across multiple domains with simulated web and knowledge graph APIs, though still limited in scale and dynamic renew ability.

LLM & RAG Robustness Recent frameworks attempt to quantify RAG robustness, usually with various perturbations. RAGAS [Es et al., 2025] measures factual consistency through automated evaluation without ground-truth annotations but lacks assessment of query/document perturbations and limited number of assessment. Cao et al. [2025] analyzed the robustness of the RAG system on linguistic variations and found that RAG systems are even more sensitive to these variations compared with LLM-only generation. SURE Yang et al. [2025b] introduced a framework to quantify the sensitivity to semantic-agnostic spurious features (e.g. format of document) in grounding data, providing a taxonomy of formatting variations that reveal widespread vulnerabilities. QE-RAG [Zhang et al., 2025b] tests robustness by injecting realistic query entry errors into QA datasets to evaluate tolerance to input noise, though primarily focused on static, general-domain tasks without evaluating document-level corruptions. KaRR [Dong et al., 2023] provides a statistical approach to assess whether an LLM contains reliable factual knowledge by estimating the ration of generating correct surface text given varying prompts, although its assessment is limited to parametric knowledge rather than retrieval capabilities. While these approaches advance discrete facts of RAG robustness, none offer a unified, dynamic evaluation pipeline capable of automatically generating large-scale, time-sensitive test cases and measuring performance under systematic perturbations to queries, documents, and retrieval results.

3 *RARE*-Met: Retrieval-Aware Robustness Metric

A robust RAG system should have the ability to maintain correctness despite perturbations when (1) it possesses internal knowledge ($g(q, d = \varnothing)$ is correct) or (2) it can effectively utilize the retrieved information when lacking that knowledge. So, to measure it, the robustness is defined as

- When the generator can answer the query correctly without retrieval, it should always answer it correctly regardless of the retrieval (whether the retrieval is correct, incorrect, or irrelevant)
- When the generator cannot answer the query correctly with no retrieval, it should be able to answer it correctly when given the correct retrieval. Otherwise, it should refuse to answer the question rather than provide a hallucinated answer.

The full definition of robustness under various perturbations is given in Table 2.

Table 2: Ideal return of a robust retrieval-augmented language model under various perturbations (query, document, and real-world retrieval results). ✓ = answers correctly; ✗ = answers incorrectly; ? = If retrieval results does not contain correct answer, the system should indicate clearly that the answer is not available, such as returning "no such info". Which means if neither language model's internal nor provided external knowledge has the correct answer, the system should show its uncertainty about the answer; ✓/ ?= If retrieval results contain correct answer, the system should return correct answer, otherwise return "no such info"; ✓/ ✗ = answer could be whether available or not.

Document		$g(q, d = \varnothing) =$ ✓		$g(q, d = \varnothing) =$ ✗	
d	Answer Available	Standard q	Variants q'	Standard q	Variants q'
Ground-Truth (GT)	✓	✓	✓	✓	✓
Lexical Different	✓	✓	✓	✓	✓
Lexical Similar	✗	✓	✓	?	?
Real-World Retrieval	✓/ ✗	✓	✓	✓/ ?	✓/ ?

3.1 Query Perturbations

We define 4 types of query perturbations $Q' = q'_1, q'_2, ..., q'_n$ derived from the original query q:

- **Surface-level perturbations:** (1) Character-level perturbation; (2) word-level perturbations (typos, synonyms) based on TextAttack [Morris et al., 2020].
- **Advanced-level perturbations:** (1) LLM-based grammar changes without changing its intrinsic meaning; (2) LLM-based irrelevant information addition

3.2 Document Perturbation

For document perturbation, we primarily consider two directions: lexical relevance and answer relevance. Similarly to definitions under query perturbation, the lexical relevance measure changes the text of the documents itself. Answer relevance, on the other hand, determines whether the retrieved document truly contains the answer required by the question. As we consider lexical perturbation and answer perturbation as two dimensions, we define three document perturbations which encompassed all possible distributions of retrieval documents. (1) Documents with the similar lexical style but answers are different: directly remove the answer sentence/words from the ground truth chunk. (2) Documents with different lexical style but answer is similar/identical: LLM-based back-translation. (3) Real-world retrieval results: constructing a real-world simulated retrieval process. Appendix A shows all types of document perturbations under such relevance. The purpose of including the first two document perturbations is to better analyze the impact of different types of relevance (whether lexical or answer-based) on the overall robustness of the RAG system.

3.3 Robustness Metrics

Leveraging the query and document perturbations described above, we assign a robustness score to each setting. We define a binary robustness judge function $f(a, a')$, which returns 1 when the answer is robust or 0 otherwise. The decision rules are shown in Table 2.

3.3.1 Overall Robustness

Overall robustness is evaluated over the Cartesian product of all query and document perturbations:

$$\frac{1}{|Q|\,|D|} \sum_{q' \in Q} \sum_{d' \in D} f\big(\phi(q', d'), a\big) \tag{1}$$

where ϕ is the answer generator, Q and D are the complete sets of perturbed queries and documents (also including the original query and ground-truth document), q' and d' denote individual perturbations, and a is the reference answer for the corresponding question.

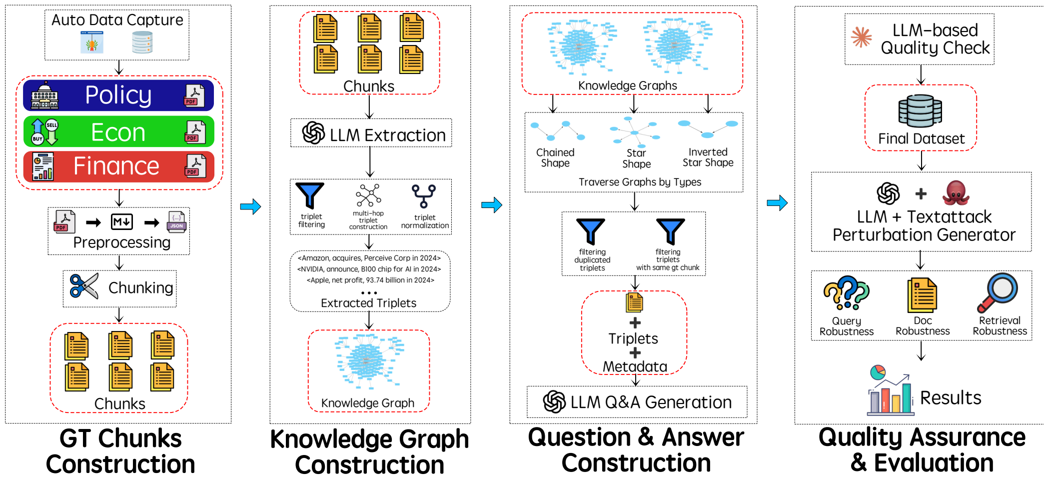

Figure 1: Illustration for the *RARE* framework. Red frame: data that pipeline will generate; **Black frame**: process/movement.

3.3.2 Query Robustness

Query robustness score measures RAG systems' performance with fixed ground-truth documents and varying query perturbations:

$$\frac{1}{|Q'|} \sum_{q' \in Q'} f\big(\phi(q', d_{\text{gt}}), a\big) \tag{2}$$

where d_{gt} is the ground-truth document.

3.3.3 Document Robustness

Document robustness score measures RAG systems' performance with varying document perturbations and the fixed original query:

$$\frac{1}{|D'|} \sum_{d' \in D'} f(\phi(q, d'), a) \tag{3}$$

where D' is the set of all document perturbations derived from the ground-truth document.

3.3.4 Real-World Retrieval Robustness

Real-world retrieval robustness captures how reliably the system reproduces the correct answer when the original query is evaluated against the diverse set of documents returned by different embedding-based retrieval models.

$$\frac{1}{|D_{\text{ret}}|} \sum_{d'_i \in D_{\text{ret}}} f(\phi(q, d'_i), a) \tag{4}$$

d'_i denotes the all top-3 documents retrieved by the i^{th} embedding model. D_{ret} denotes the set of documents retrieved by the all embedding models and q is the original query.

4 *RARE*-Get: Dynamic RAG Benchmark Dataset Generation Pipeline

RAG benchmarks should ideally comprise diverse, realistic queries with corresponding golden passages containing the information needed to answer them correctly. Creating such benchmarks manually demands extensive human effort and domain expertise, particularly for specialized, multi-hop reasoning scenarios. In addition, manual-based benchmark cannot consistently create the dynamic and up-to-date datasets. To address these challenges, we introduce *RARE*-Gen, a fully automated pipeline for constructing complex RAG benchmarks for domain corpora.

RARE-Gen transforms domain-specific documents into comprehensive benchmark datasets through four key stages: (1) Ground Truth Chunks Construction; (2) Knowledge Graph Construction; (3) Question & Answer Construction and (4) Quality Assurance, as illustrated in Figure 1.This approach enables the creation of technical, challenging RAG evaluation datasets that evolve dynamically alongside their source documents, ensuring continued relevance in rapidly changing domains.

4.1 Corpus Preparation and Chunking

The pipeline begins by processing domain-specific documents, converting them into manageable chunks suitable for retrieval systems. We carefully segment each document into passages of approximately 600 tokens, striking a balance between informativeness and retrieval efficiency, as well as a real-world retrieval simulation. For tables, we prevent splitting a single table across different chunks. Related information (e.g. table titles, data explanation) will remain in the same chunk. Similarly, for text-only contents, we ensure that no paragraph is divided between chunks. Also, we develop specialized chunking techniques across three distinct domains. Each domain receives tailored processing to enhance information extraction and context retention.

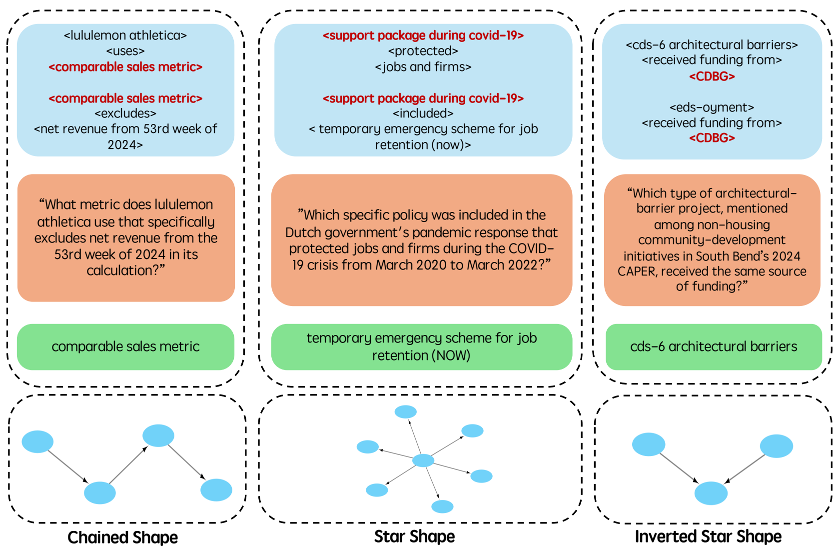

Figure 2: Examples of the multi-hop questions. Blue: triplets traversed from KG; Peach: generated question; Green: generated answer; Red: "bridge" entity which connect different triplets together;

4.2 Knowledge Graph Extraction

The cornerstone of the benchmark creation process is systematically transforming chunked documents into structured knowledge representations. For each set of n consecutive chunks, we employ LLM (GPT-4.1 [OpenAI, 2025]) with carefully designed prompts adapted for different domains.

The prompts specify multiple types of multi-hop question patterns with detailed examples, instructing the LLM to extract connected triplets where entities overlap between chunks. In addition, we ask LLM to extract the source sentence it used to extract the triplet, which will be further implemented as answer verification, making sure the generated answer is included in the ground truth chunk. Lastly, we normalize semantically similar relations (e.g. "manufactures" vs. "produces") using E5-Mistral-7B-Instruct [Wang et al., 2023], one of the leading embedding models according to the MTEB leaderboard [Muennighoff et al., 2023]. Finally, after constructing the corresponding knowledge graph for each document, we merge different knowledge graphs into a larger knowledge graph to create cross-document questions. Example prompts for knowledge graph triplets extraction are in Appendix B.

4.3 Query Patterns

By systematically traversing the constructed knowledge graph, we identify one single-hop and three multi-hop structural patterns that serve as templates for generating queries of varying complexity. Examples of multi-hop patterns and QA pairs are shown in Figure 2.

- **Single-hop patterns** capture direct relationships between two entities represented by (e_1, r_1, e_2), forming the baseline for simple retrieval tasks.
- **Chained-shape patterns** identify sequences of 2-3 linked triplets where entities are consecutively connected: $(e_1, r_1, \boldsymbol{e_2})$, $(\boldsymbol{e_2}, r_2, e_3), ...$ These patterns require systems to follow multi-step reasoning paths spanning multiple chunks, with each intermediate entity serving as a bridge to the next information piece.
- **Star-shape patterns** capture scenarios in which a central entity connects to multiple other entities through different relations: $(\boldsymbol{e_1}, r_1, e_2)$, $(\boldsymbol{e_1}, r_2, e_3), ...$ These patterns test a system's ability to aggregate diverse information around a focal concept, requiring the synthesis of multiple facts sharing a common subject.
- **Inverted-star-shape patterns** identify cases where multiple distinct entities are related to a common entity: $(e_1, r_1, \boldsymbol{e_2})$, $(e_3, r_2, \boldsymbol{e_2}), ...$ These patterns challenge systems to recognize convergent information paths and synthesize facts from multiple sources that lead to a common conclusion.

When traversing the entire graph according to these patterns and identifying the corresponding triplet(s), we ensure that the extracted triplets can only be used to generate corresponding questions. For instance, while traversing all single-hop triplets (e_1, r_1, e_2), we ensure that e_1 has an out-degree of 1 and an in-degree of 0, while e_2 has an in-degree of 1 and an out-degree of 0. This approach prevents duplication of content between single-hop and multi-hop questions. Additionally, for multi-hop questions, we remove all triplet sets that can be entirely answered from the same chunk. This ensures that multi-hop questions must be answered by traversing multiple files. Example prompts are in Appendix B.

4.4 Query Generation and Quality Assurance

For each identified pattern, we use pattern-specific prompts to generate QA pairs that use information from its triplets, corresponding ground truth chunks, and metadata storing information such as timestamp or the country name. For multi-hop questions specifically, we implement a specialized algorithm that: (1) Identifies a "pivot entity" that connects different triplets; (2) References this pivot indirectly in the question; (3) Ensures the question cannot be answered from a single chunk; (4) Performs "pivot-rarity" and "negative-distractor safety" checks to guarantee question quality. Finally, all generated query-answer pairs undergo rigorous quality assessment using separate LLM-based evaluation based on Claude 3.5 Haiku [Claude, 2024] that scores each query-answer pair on three dimensions from the scale of 1 to 5:

- **Reasonableness:** Whether the question appears natural and likely to be asked in this domain
- **Clarity** Whether the question is clearly stated and unambiguous
- **Correctness:** Whether the answer is factually accurate and fully derived from the provided chunks

Only queries with scores above 3 across all dimensions are included in the final benchmark. This quality-controlled generation process creates benchmarks that effectively evaluate both retrieval accuracy and reasoning capabilities within domain-specific contexts. As source documents evolve or new ones are added, the pipeline can dynamically extend the benchmark, ensuring continued relevance for evaluating RAG systems against the latest information. All prompts are in Appendix B.

5 *RARE*-Set: Large-Scale Domain-Specific RAG Dataset

RARE-Set contains three different domains of datasets: finance, economics, and policy. We collect a heterogeneous corpus with 150 recent S&P 500 Companies' SEC 10-k filings, 114 OECD economic

Table 3: Knowledge graph traversal statistics by domain

Domain	**Financial**	**Economics**	**Policy**
Document	199	114	214
Chunk	19825	12915	7014
Time Scope	2024-2025	2020-2025	2024-2025
Total # of Eligible Triplet/Triplets			
Single-hop	17585	6719	6176
Chained (multi-hop)	11193	22256	82885
Star-shaped (multi-hop)	2707	1780	4868
Inverted-star-shaped (multi-hop)	558	2636	7377
Query (Train)			
Single-hop	7362	6715	6125
Chained (multi-hop)	7930	3863	7563
Star-shaped (multi-hop)	833	511	661
Inverted-star-shaped (multi-hop)	64	415	253
Query (Test)			
Single-hop	1000	1000	1000
Chained (multi-hop)	687	774	805
Star-shaped (multi-hop)	289	193	135
Inverted-star-shaped (multi-hop)	24	60	60

surveys, and 214 Consolidated Annual Performance and Evaluation Report (CAPER) from grantees for U.S. Department of Housing and Urban Development (HUD) funded programs. The datasets distribution is shown in Table 3.

We enhance datasets quality through a variety of processing techniques. For financial reports, our preprocessing pipeline builds on Edgar-Crawler [Loukas et al., 2021], with custom modifications. Rather than preserving tables in HTML format, we convert them to a markdown structure optimized for LLM inputs. In knowledge graph extraction from financial documents, we prioritize relations involving performance metrics, operational activities, and financial events. We explicitly target generalized and reusable relations that can be applied across companies within the same industry. This approach supports the generation of multi-hop questions that span multiple companies. For economic surveys, we design prompts to emphasize policy measures, key economic indicators, and patterns of national development. In the context of policy reports, our focus is on fund allocation, program implementation, and beneficiary data.

The benchmark contains single-hop queries and three types of multi-hop queries based on different knowledge patterns in the knowledge graph. One thing to mention is that all of these datasets are time-sensitive and can expand dynamically as time progresses.

6 Robustness Experiments and Analysis

6.1 Experimental Setting

We perform our experiments on a total of 6000 QA pairs for three domains, each of which has 1000 single-hop questions and 1000 multi-hop questions. Retrieval is evaluated with three top-ranking embedding models from the MTEB leaderboard: E5-Large-Instruct [Es et al., 2025], Jina-Embedding-v3 [Sturua et al., 2024], and Stella-En-1.5B-v5 [Zhang et al., 2025a]. For the RAG system's generators, we test leading open-source LLMs (Qwen 3 [Yang et al., 2025a] and Llama 3 family [Grattafiori et al., 2024]) and closed GPT models accessed via APIs. All generators run deterministically (temperature = 0) with a maximum output length of 1024 tokens. Although we ask each model to return a concise final answer, we explicitly encourage chain-of-thought reasoning to appear in the response. All open-source models are running on 16 A6000 Ada GPUs, each with 48 GB of VRAM. To accelerate testing, we implement multiple vLLM-based [Kwon et al., 2023] LLM servers in parallel and distributed inference requests across them. Experiments involving larger models, such as Qwen3-32B and Llama-3.1-70B-Instruct, are executed separately. It takes around 5 days to run all the test cases.

To measure the difference between answers generated by the RAG system and ground-truth, we create a two-stage evaluator. It first normalizes both strings and checks for exact or sub-string matches. If no lexical match is found, it computes cosine similarity between sentence embedding representations of the two texts using E5-Mistral-7B-Instruct Wang et al. [2023]; a prediction is labeled correct when this similarity exceeds 0.9. This hybrid criterion captures both verbatim and semantically equivalent answers while remaining robust to minor paraphrasing.

Table 4: Robustness results across different models and metrics

Model	Overall	Query	Document	Retrieval
Llama-3.1-8B-Instruct	0.427	0.549	0.472	0.337
Llama-3.1-70B-Instruct	0.542	0.627	**0.506**	0.542
Qwen3-4B	0.536	0.575	0.446	0.583
Qwen3-8B	0.555	0.584	0.490	0.583
Qwen3-14B	**0.566**	0.586	0.499	**0.602**
Qwen3-32B	0.557	0.631	0.469	0.592
GPT-4.1-nano	0.498	0.563	0.363	0.567
GPT-4.1-mini	0.422	0.612	0.395	0.351
GPT-4.1	0.542	**0.645**	0.438	0.576

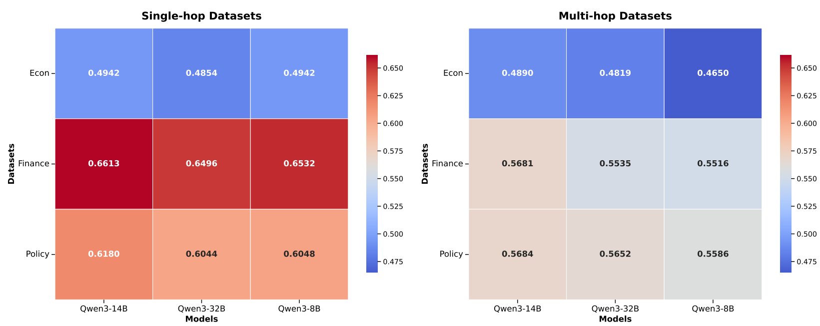

Figure 3: Overall robustness scores across different domains among top-3 LLMs.

6.2 Overall Model Performance

Examining the overall robustness scores in the Table 4 shows that larger models generally demonstrate superior robustness. Qwen3-14B achieves a robustness score that surpasses those of its smaller models, Qwen3-8B and Qwen3-4B. A similar size-dependent trend is observed within the Llama 3 series: the 70-billion-parameter Llama3-70B exhibits a markedly higher robustness score than Llama3-8B. Size alone, however, does not always reflects the robustness. For example, Qwen3-32B attains an overall robustness score lower than that of the smaller - but architecturally similar - Qwen3-14B, and GPT-4.1-mini is outperformed by the even more compact GPT-4.1-nano. Across all experiments, the Qwen 3 family consistently demonstrates superior robustness, with Qwen3-8B even surpassing the considerably larger Llama-3.1-70B. These findings highlight the decisive roles of architectural design and training methodology. More analysis about each sub-metric is available in Appendix C.

6.3 Domain-Specific and Multi-Hop Questions Robustness

The significant performance variation across domains indicates that RAG systems' robustness is heavily influenced by domain-specific factors. RAG system perform best in finance reports, which typically feature standardized terminology and numerical data. However, they are struggling most with the economics survey, which often involves complex causal relationships and varied terminology.

In addition, single-hop queries consistently yield higher robustness scores than multi-hop queries across most domains and perturbations (Figure 7 reveals more information). This trend is amplified in smaller models, suggesting that multi-hop reasoning capabilities require substantial model capacity to maintain robustness under perturbations.

7 Conclusion

In conclusion, we introduce *RARE*, a comprehensive framework for evaluating RAG robustness that addresses critical gaps in existing benchmarks. Our knowledge graph based pipeline (*RARE*-Get) automatically extracts relations from specialized corpora and generates multi-level questions through pattern-based traversal, enabling dynamic dataset evolution without manual curation. The resulting benchmark (*RARE*-Set) comprises 48,322 questions across finance, economics, and policy domains, featuring single-hop and complex multi-hop questions. Our robust evaluation metrics (*RARE*-Met) systematically measure resilience against query, document, and retrieval perturbations. Experiments reveal document robustness as the universal weak point across all models, with performance variations not strictly correlating with model size. RAG systems consistently demonstrate higher robustness in finance than economics, and single-hop queries outperform multi-hop ones across all domains, providing crucial insights for developing more reliable RAG systems for real-world applications.

References

Tianyu Cao, Neel Bhandari, Akhila Yerukola, Akari Asai, and Maarten Sap. Out of style: Rag's fragility to linguistic variation, 2025. URL `https://arxiv.org/abs/2504.08231`.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking large language models in retrieval-augmented generation, 2023. URL `https://arxiv.org/abs/2309.01431`.

Claude. Claude 3.5 haiku, 2024. `https://www.anthropic.com/claude/haiku`.

Qingxiu Dong, Jingjing Xu, Lingpeng Kong, Zhifang Sui, and Lei Li. Statistical knowledge assessment for large language models, 2023. URL `https://arxiv.org/abs/2305.10519`.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. Ragas: Automated evaluation of retrieval augmented generation, 2025. URL `https://arxiv.org/abs/2309.15217`.

Robert Friel, Masha Belyi, and Atindriyo Sanyal. Ragbench: Explainable benchmark for retrieval-augmented generation systems, 2025. URL `https://arxiv.org/abs/2407.11005`.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024. URL `https://arxiv.org/abs/2312.10997`.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021. doi: 10.1162/tacl_a_00370. URL `https://aclanthology.org/2021.tacl-1.21/`.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL `https://arxiv.org/abs/2407.21783`.

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6237–6250, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.418. URL https://aclanthology.org/2022.emnlp-main.418/.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. Realtime qa: What's the answer right now?, 2024. URL https://arxiv.org/abs/2207.13332.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention, 2023. URL https://arxiv.org/abs/2309.06180.

Lefteris Loukas, Manos Fergadiotis, Ion Androutsopoulos, and Prodromos Malakasiotis. EDGAR-CORPUS: Billions of Tokens Make The World Go Round. In *Proceedings of the Third Workshop on Economics and Natural Language Processing (ECONLP)*, pages 13–18, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.econlp-1.2.

Jannat Ara Meem, Muhammad Shihab Rashid, Yue Dong, and Vagelis Hristidis. Pat-questions: A self-updating benchmark for present-anchored temporal question-answering, 2024. URL https://arxiv.org/abs/2402.11034.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, 2020.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. MTEB: Massive text embedding benchmark. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.148. URL https://aclanthology.org/2023.eacl-main.148/.

OpenAI. Introducing gpt-4.1 in the api, 2025. https://openai.com/index/gpt-4-1/.

Phillip Schneider, Tim Schopf, Juraj Vladika, Mikhail Galkin, Elena Simperl, and Florian Matthes. A decade of knowledge graphs in natural language processing: A survey, 2022. URL https://arxiv.org/abs/2210.00105.

Hongda Shen and Eren Kurshan. Temporal knowledge distillation for time-sensitive financial services applications, 2023. URL https://arxiv.org/abs/2312.16799.

Saba Sturua, Isabelle Mohr, Mohammad Kalim Akram, Michael Günther, Bo Wang, Markus Krimmel, Feng Wang, Georgios Mastrapas, Andreas Koukounas, Andreas Koukounas, Nan Wang, and Han Xiao. jina-embeddings-v3: Multilingual embeddings with task lora, 2024. URL https://arxiv.org/abs/2409.10173.

Yixuan Tang and Yi Yang. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries, 2024. URL https://arxiv.org/abs/2401.15391.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL `https://aclanthology.org/2022.tacl-1.31/`.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. FreshLLMs: Refreshing large language models with search engine augmentation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13697–13720, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.813. URL `https://aclanthology.org/2024.findings-acl.813/`.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*, 2023.

Cheng Xu, Shuhao Guan, Derek Greene, and M-Tahar Kechadi. Benchmark data contamination of large language models: A survey, 2024. URL `https://arxiv.org/abs/2406.04244`.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL `https://arxiv.org/abs/2505.09388`.

Shiping Yang, Jie Wu, Wenbiao Ding, Ning Wu, Shining Liang, Ming Gong, Hengyuan Zhang, and Dongmei Zhang. Quantifying the robustness of retrieval-augmented language models against spurious features in grounding data, 2025b. URL `https://arxiv.org/abs/2503.05587`.

Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Ethan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, Lei Chen, Nicolas Scheffer, Yue Liu, Nirav Shah, Rakesh Wanga, Anuj Kumar, Wen-tau Yih, and Xin Luna Dong. Crag - comprehensive rag benchmark. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 10470–10490. Curran Associates, Inc., 2024. URL `https://proceedings.neurips.cc/paper_files/paper/2024/file/1435d2d0fca85a84d83ddcb754f58c29-Paper-Datasets_and_Benchmarks_Track.pdf`.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering, 2018. URL `https://arxiv.org/abs/1809.09600`.

Dun Zhang, Jiacheng Li, Ziyang Zeng, and Fulong Wang. Jasper and stella: distillation of sota embedding models, 2025a. URL `https://arxiv.org/abs/2412.19048`.

Kepu Zhang, Zhongxiang Sun, Weijie Yu, Xiaoxue Zang, Kai Zheng, Yang Song, Han Li, and Jun Xu. Qe-rag: A robust retrieval-augmented generation benchmark for query entry errors, 2025b. URL `https://arxiv.org/abs/2504.04062`.

A Three Types of Document Perturbations

Figure 4 illustrates that the real-world retrieval results have covered all the distributions. The purpose of setting document perturbations for answer similar+lexical different (orange part) and answer different+lexical similar (blue part) is to explore the effect of lexical/answer relevance on the robustness of the RAG system.

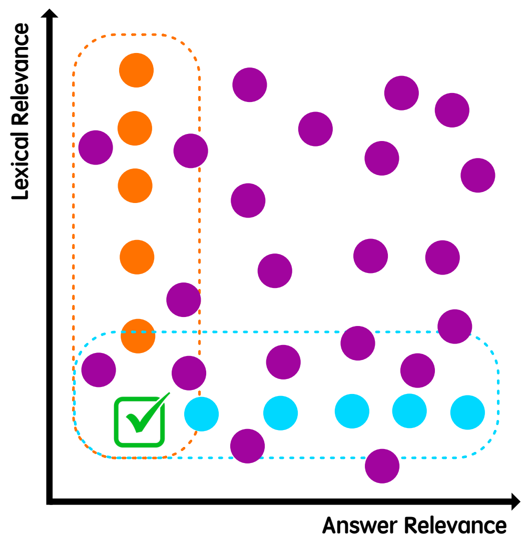

Figure 4: Three types of document perturbations measured by two relevance

B Prompts

B.1 Dataset Construction

We will use the economic dataset prompts as the example.

System Prompt: Triplets Extraction

```
You are an economic analyst skilled at interpreting OECD Economic Surveys. Your
task is to extract structured triplets consisting of **{{"entity_1", "relation",
"entity_2"}}** from provided consecutive text chunks from a single OECD Economic
Survey. Each triplet must **be supported explicitly by one specific chunk**, but
other chunks can be referenced to form insightful, multi-hop triplets. You should
 include the **source chunk ID** and **source sentence** as the metadata of the
triplets.

# TASK: EXTRACT STRUCTURED MULTI-HOP TRIPLETS

Extract triplets fitting these multi-hop categories:

- Connected Chain
- Star
- Inverted Star

### 1. Connected Chain Triplets:

- Extract an initial triplet: <entity_1, relation, entity_2>.
- Then identify subsequent triplets where entity_2 of the previous triplet
becomes entity_1 of the next.
- Ideally, different subsequent triplets should be sourced from different chunks.
- Extract as many meaningful chains as possible.
```

```
- Skip if no valid connected chain is available.

**Example:**

- {{"entity_1": "Luxembourg", "relation": "implemented", "entity_2": "free public
 transport"}}
- {{"entity_1": "free public transport", "relation": "aims to reduce", "entity_2":
 "carbon emissions"}}

### 2. Star Triplets:

- One root entity branching into multiple distinct relationships.
- Each branch must independently derive from a unique chunk.
- Skip if no meaningful star relationship is possible.

**Example:**

- {{"entity_1": "Luxembourg", "relation": "invests in", "entity_2": "renewable
energy"}}
- {{"entity_1": "Luxembourg", "relation": "develops", "entity_2": "sustainable
transport infrastructure"}}

### 3. Inverted Star Triplets:

- Two distinct entities connected through a shared attribute (entity_2).
- Relations may differ and offer varied perspectives on the attribute.
- Skip if no valid inverted star relationship is possible.

**Example:**

- {{"entity_1": "Luxembourg", "relation": "faces challenges in", "entity_2": "
housing affordability"}}
- {{"entity_1": "OECD recommendations", "relation": "address", "entity_2": "
housing affordability"}}

## REQUIRED STRUCTURE:

Each extracted triplet must include:

- entity_1 (str)
- relation (str)
- entity_2 (str)
- answer_chunk_id (str)
  - The chunk ID is at the very beginning of each text chunk, such as "Chunk ID:
  economics_0e32d909-en_chunk_9".
  - You should copy the chunk ID where the triplet is extracted from as the "
  answer_chunk_id".
- source_sentence (str)
  - Extracted exactly from the supporting chunk, COPY WORD BY WORD.
  - If sourced from a table, strictly include relevant row, column, and specific
  data only.

## CRITICAL INSTRUCTIONS:

### Relations:

- Generalized and reusable across similar economic and policy contexts.
- Concise and specific (2-4 words preferred).
- Use standard economic and policy terminology.
- Avoid specific dates or overly detailed references in the relations.

**Good Examples:**
```

```
- "implemented", "faces challenges in", "invests in", "promotes"

**Bad Examples:**

- "introduced free transport in 2020", "planned reforms announced in 2023"

### Entities:

- Clearly specify entities (avoid general terms like "the country" or "the
government").
- Maintain consistent terminology when referring to similar concepts, such as
using "Luxembourg" all the time instead of using "Luxembourg government"
sometimes.
- Include specific, detailed information relevant to economic policies,
recommendations, or outcomes.
- For table-derived entities, clearly indicate row, column, and description.

## Goal:

Try to extract 15 to 20 triplets. If no valid connected triplets can be extracted,
 return an empty array: []
```

System Prompt: Single-Hop QA Pairs Generation

```
Create an economics-related natural question-answer pair using a relation triplet
 (entity_1, relation, entity_2) based on the text context and the file metadata
where the triplet was extracted.

# Requirements

- The question and answer should be entirely based on the given text context;
that is, one can only generate the correct answer from the information available
in the context.
- Always use "{file_country}" instead of "{file_country} government," "government
," or "country" to make the query more specific.
- You should use entity_1 or entity_2 as the answer to the question and construct
 the question using the other entity and relation with appropriate context
information.
- Aim to formulate questions that appear natural and are likely to be asked by a
human.
- Avoid generating questions that are overly general or vague, where multiple
ground truth chunks could answer the question or it would be hard to retrieve the
 ground truth chunk given the question. You MUST return an EMPTY string for
question and answer in this case.

# Examples

Example 1:
Triplet:
{{"entity_1": "inflation", "relation": "is", "entity_2": "2.9\% in 2023"}}

Text Context:
[Full example context is omitted...]

Metadata:
- File Type: OECD Economic Surveys
- Country Surveyed: Luxembourg
- Survey Year: 2023

Output:
{{"question": "What is the inflation of Luxembourg in 2023?", "answer": "2.9\%"}}
```

```
Example of Vague Triplet (Should Return Empty):
Triplet:
{{"entity_1": "luxembourg", "relation": "should maintain", "entity_2": "prudent
fiscal policy"}}

Text Context:
[Full example context is omitted...]

Metadata:
- File Type: OECD Economic Surveys
- Country Surveyed: Luxembourg
- Survey Year: 2023

Output:
{{"question": "", "answer": ""}}

# Output Format

Respond in JSON format with "question" and "answer" fields encapsulating the
formulated question and its answer.

# Notes

Ensure questions are specific to the context provided, emphasizing precision and
clarity in wording. If no singular answer emerges due to generality, opt for
returning an empty dictionary to indicate an unsuitably specific query.
```

System Prompt: Multi-Hop QA Pairs Generation

```
You are a benchmark designer creating **multi-hop retrieval questions** based on
three types of multi-hop triplets.

### Input
- Triplet 1 = ({{head1}}, {{rel1}}, {{tail1}}) <- extracted from Chunk 1
- Triplet 2 = ({{head2}}, {{rel2}}, {{tail2}}) <- extracted from Chunk 2
- Chunk 1: {{chunk1}}
- Chunk 2: {{chunk2}}

### Multi-hop Triplets DEFINITIONS
1. Chain Triplets
- Gurantee: {{tail1}} == {{head2}}
- Define A = {{head1}}, B = {{tail1}} / {{head2}}, C = {{tail2}}

2. Star-shaped Triplets
- Gurantee: {{head1}} == {{head2}}
- Define A = {{tail1}}, B = {{head1}} / {{head2}}, C = {{tail2}}

3. Inverted-star-shaped Triplets
- Gurantee: {{tail1}} == {{tail2}}
- Define A = {{head1}}, B = {{tail1}} / {{tail2}}, C = {{head2}}

### GOAL
Write ONE natural-language **multi-hop** question that *requires* evidence from
both chunks and answer it succinctly (no full sentences, only essential
information).

### ALGORITHM
1. Decide whether the final answer will be **A** or **C**.
   - Pick **A** if you can phrase the question so the solver must:
     - hop-1: use (C, rel2) to identify B,
     - hop-2: use (B, rel1) to reach A.
```

```
   - Pick **C** if you can phrase the question so the solver must:
     - hop-1: use (A, rel1) to identify B,
     - hop-2: use (B, rel2) to reach C.

2. Write a fluent, specific, and natural question that:
   - References the **pivot B** indirectly (via the opposite hop as above).
   - Omits the answer itself.
   - Cannot be answered from a single chunk.
   - Includes detailed and specific context from the source text chunks. DO NOT
   just use "according to OECD Economic Survey".
      - BAD example: "What is the primary export sector of the country that faces
      risk from global supply chain disruptions?" (Too vague; could refer to any
      country)
      - GOOD example: "What is the primary export sector of the country that faces
       risk from global supply chain disruptions in Q3 2021?" (Specific to the
      context and time frame)

3. Return the answer based on A or C. Ensure the answer precisely matches the
facts provided in the context.

### EXAMPLE
{{"entity_1": "forward-looking fuel-tax trajectory", "relation_1": "would reduce",
 "entity_2": "reliance on combustion-engine cars"}}
{{"entity_1": "reliance on combustion-engine cars", "relation_2": "drives", "
entity_2": "transport-sector emissions"}}

*question*: Which forward-looking tax trajectory is proposed to cut the main
driver of transport-sector emissions?
*answer*: forward-looking fuel-tax trajectory

### QUALITY CHECKS
- **Pivot-rarity**: B must be distinctive (>= 2 meaningful words, not generic
terms like "measures", "it", "the company"). If B is too generic, output **empty
strings for the question and answer**.
- **Negative-distractor safety**: Ask could a system answer your question after
retrieving only *one* chunk? If yes, output **empty strings for the question and
answer**.

### OUTPUT
Respond in JSON format with question and answer only as shown below:
{{
  "question": "...",
  "answer": "..."
}}
```

B.2 Quality Assurance

System Prompt: Single-Hop QA Pairs Quality Assurance

```
# Single-Hop Query Quality Evaluator

You are an expert evaluator of single-hop queries. Assess each query's quality
across two dimensions on a 1-5 scale.

## Assessment Criteria

### 1. Clarity (Question and Answer) (1-5)
- **5**: Concise, unambiguous wording; answer mirrors clarity
- **4**: Minor wording issue but still unambiguous
- **3**: Some vagueness but meaning recoverable
- **2**: Ambiguities/redundancies hinder understanding
```

```
- **1**: Unclear or contradictory wording

### 2. Correctness (vs. Ground-Truth) (1-5)
- **5**: Answer matches all facts in chunks; nothing missing
- **4**: Correct but one minor fact omitted/loosely paraphrased
- **3**: At least half of facts correct; one factual slip
- **2**: Major fact missing/misstated/unsupported
- **1**: Contradicts or ignores ground truth

## Evaluation Process
1. Identify reasoning process
2. Assess alignment between query and provided text chunk
3. Evaluate clarity of question and answer
4. Verify factual correctness against ground-truth chunk

## Input
- query: The single-hop question
- answer: The provided answer
- text chunk: Source text chunk

## Output
{{
  "score": <average_of_dimension_scores>,
  "dimension_scores": {
    "clarity": <1-5>,
    "correctness": <1-5>
  }
}}
```

System Prompt: Multi-Hop QA Pairs Quality Assurance

```
# Multi-Hop Query Quality Evaluator

You are an expert evaluator of multi-hop queries. Assess each query's quality
across three dimensions on a 1-5 scale.

## Assessment Criteria

### 1. Reasonableness and Multi-hop Need (1-5)
- **5**: Meaningful question requiring all hops; each hop justified
- **4**: Reasonable but one hop weakly motivated or could be merged
- **3**: Sensible but answerable by single chunk with assumptions
- **2**: Forced/trivial question; multi-hop structure unnecessary
- **1**: Nonsensical/irrelevant; multi-hop structure meaningless

### 2. Clarity (Question and Answer) (1-5)
- **5**: Concise, unambiguous wording; answer mirrors clarity
- **4**: Minor wording issue but still unambiguous
- **3**: Some vagueness but meaning recoverable
- **2**: Ambiguities/redundancies hinder understanding
- **1**: Unclear or contradictory wording

### 3. Correctness (vs. Ground-Truth) (1-5)
- **5**: Answer matches all facts in chunks; nothing missing
- **4**: Correct but one minor fact omitted/loosely paraphrased
- **3**: At least half of facts correct; one factual slip
- **2**: Major fact missing/misstated/unsupported
- **1**: Contradicts or ignores ground truth

## Evaluation Process
1. Identify distinct reasoning hops and assess necessity
2. Check alignment between hops and provided chunks
```

```
3. Evaluate clarity of question and answer
4. Verify factual correctness against ground-truth chunks

## Input
- query: The multi-hop question
- answer: The provided answer
- text chunks: Source text chunks

## Output
{{
  "score": <average_of_dimension_scores>,
  "dimension_scores": {
    "reasonableness": <1-5>,
    "clarity": <1-5>,
    "correctness": <1-5>
  }
}}
```

B.3 RAG Generator

System Prompt: RAG Generator

```
You are a {domain} expert. You are given a {domain} question and one or multiple
contexts.
Your task is to answer the question strictly based on the these contexts.
You should think step by step and answer the question in a detailed and
comprehensive way. Please return the detailed reasoning process in the cot_answer
 part.

Requirements:
- Your answer is short and concise, do not return any other text in the answer
part.
  - Example #1: "What is the United States' GDP in 2024?"
  - Good: "\$31.1 trillion"
  - Bad: "According to the context, as my knowledge, the answer is \$31.1
  trillion"
  - Example #2: "Who is the president of the United States from 2021 to 2025?"
  - Good: "Joe Biden"
  - Bad: "The president of the United States from 2021 to 2025 is Joe Biden,
  according to my knowledge"
- If the question is not related to the context, strictly return "no such info"
for answer part. Do not return any other text in such case.

Here are some examples of how to answer based on the given context:

Example 1:
Question: What was Apple's revenue in Q2 2023?
Context: [Doc] Apple Inc. reported financial results for its fiscal 2023 second
quarter ended April 1, 2023. The Company posted quarterly revenue of \$94.8
billion, down 2.5 percent year over year.

cot_answer: The question asks about Apple's revenue in Q2 2023. According to the
context, Apple reported quarterly revenue of \$94.8 billion for its fiscal 2023
second quarter ended April 1, 2023. This represents a decrease of 2.5 percent
year over year.
answer: \$94.8 billion

Example 2:
Question: What is Luxembourg's approach to public transport?
Context: [Doc] On March 1, 2020, Luxembourg became the first country to make all
public transport free, including buses, trains, and trams. This policy aims to
```

```
reduce traffic congestion and carbon emissions while promoting sustainable
mobility solutions across the country.

cot_answer: The question asks about Luxembourg's approach to public transport.
According to the context, Luxembourg made all public transport free on March 1,
2020, becoming the first country to do so. This includes buses, trains, and trams.
 The goal of this policy is to reduce traffic congestion and carbon emissions
while promoting sustainable mobility solutions.
answer: Free public transport for all

Example 3:
Question: How many homeless individuals received emergency shelter services in
Pittsburgh?
Context: [Doc] The City of Pittsburgh allocated CDBG funds to various community
programs including affordable housing initiatives. The HOME program supported the
 construction of 45 new housing units for low-income families.

cot_answer: The question asks about the number of homeless individuals who
received emergency shelter services in Pittsburgh. After reviewing the context
carefully, I don't see any information about emergency shelter services for
homeless individuals or any numbers related to this. The context only mentions
CDBG funds for community programs and the HOME program supporting 45 new housing
units for low-income families. There is no specific information about homeless
emergency shelter services.
answer: no such info

Example 4:
Question: What were Smith A O Corp's consolidated sales for the year ended
December 31, 2024?
Context: [Doc] In this section, we discuss the results of our operations for 2024
 compared with 2023. Our sales in 2024 were \$3,818.1 million, a decrease of \$34
.7 million compared to 2023 sales of \$3,852.8 million. Our decrease in net sales
 was primarily driven by lower water heater volumes in North America, lower sales
 in China, and unfavorable currency translation of approximately \$18 million due
 to the depreciation of foreign currencies compared to the U.S. dollar, which
more than offset our higher boiler sales and pricing actions.

cot_answer: The question asks about Smith A O Corp's consolidated sales for the
year ended December 31, 2024. According to the context, the sales in 2024 were \
$3,818.1 million, which was a decrease of \$34.7 million compared to 2023 sales
of \$3,852.8 million. The context explains that this decrease was primarily due
to lower water heater volumes in North America, lower sales in China, and
unfavorable currency translation of approximately \$18 million.
answer: \$3,818.1 million

Output Format:
- cot_answer: detailed reasoning process
- answer: concise answer to the question
```

C Experiment Results Analysis

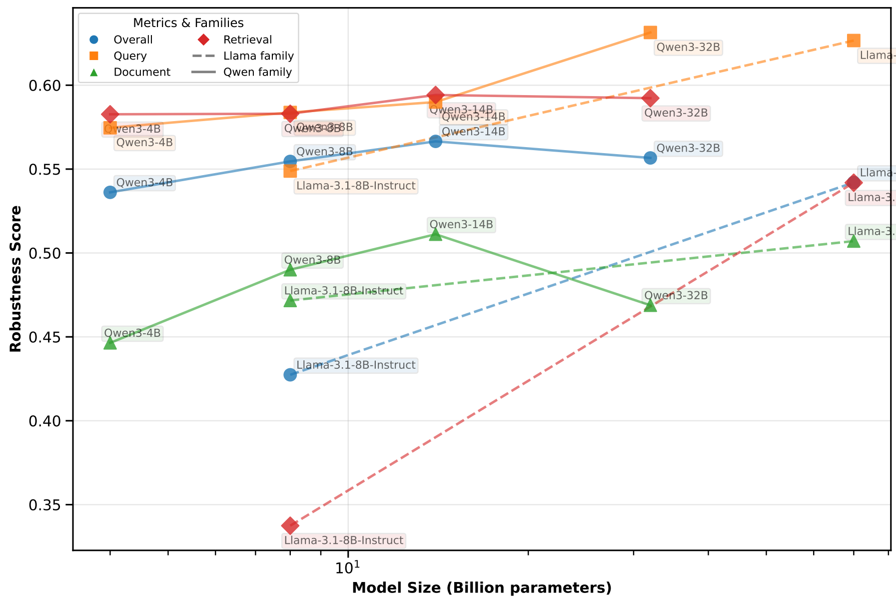

Figure 5: Relationship between the sizes of open-source generators and their robustness scores across various categories. Generally, larger generator sizes correspond to higher robustness scores. However, for Qwen 3 models, robustness scores tend to decline across all metric types—except for query perturbation—once the parameter size exceeds 14 billion.

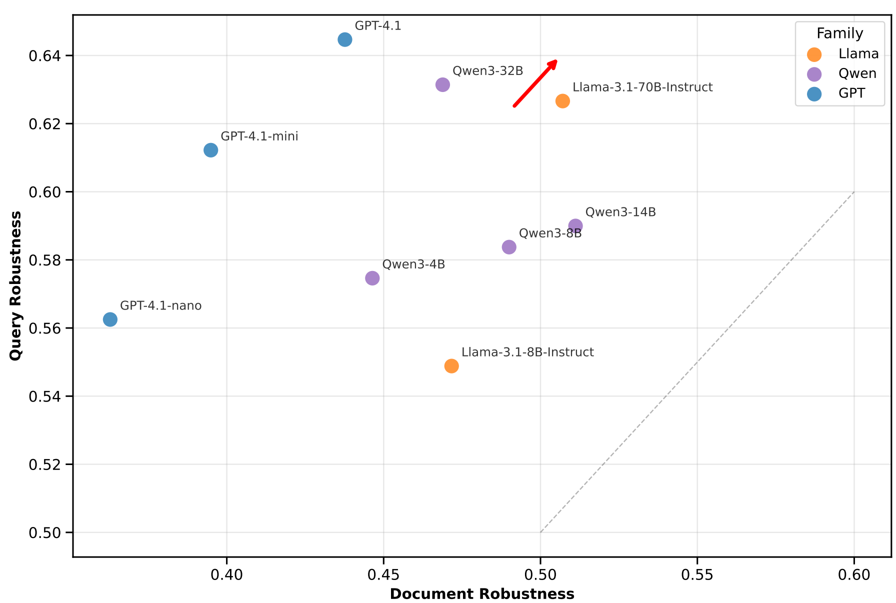

Figure 6: Relationship between robustness scores on query and document perturbations. The red arrow indicate the direction of better overall robustness. The Qwen 3 models demonstrates the best overall performance, characterized by relatively small variations in robustness scores across different parameter sizes as well as across various types of robustness metrics.

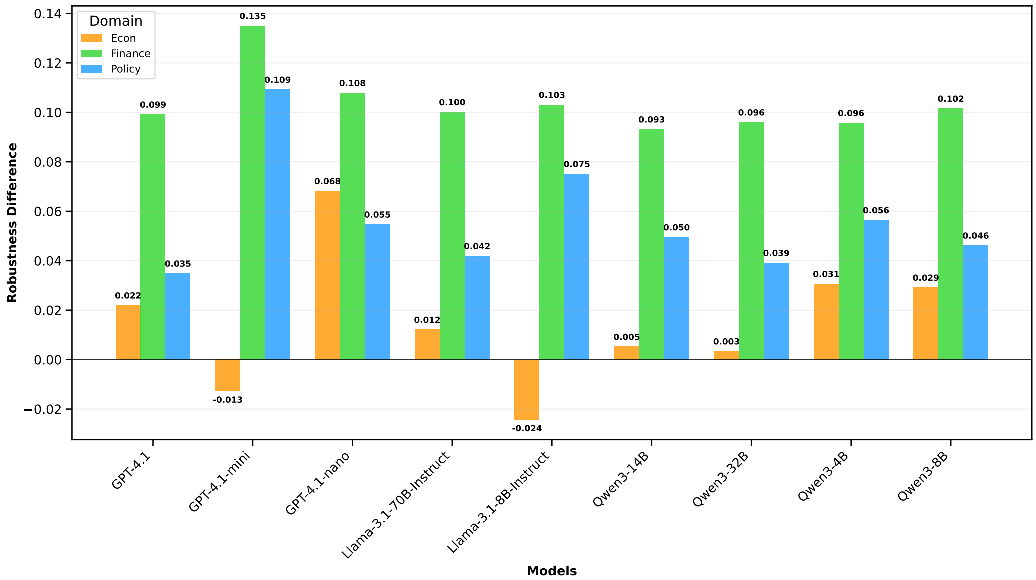

Figure 7: Difference in multi-hop and single-hop robustness scores by domain. Positive robustness scores = single-hop better, negative robustness scores = multi-hop better. RAG systems generally exhibit lower robustness on multi-hop questions compared to single-hop questions.

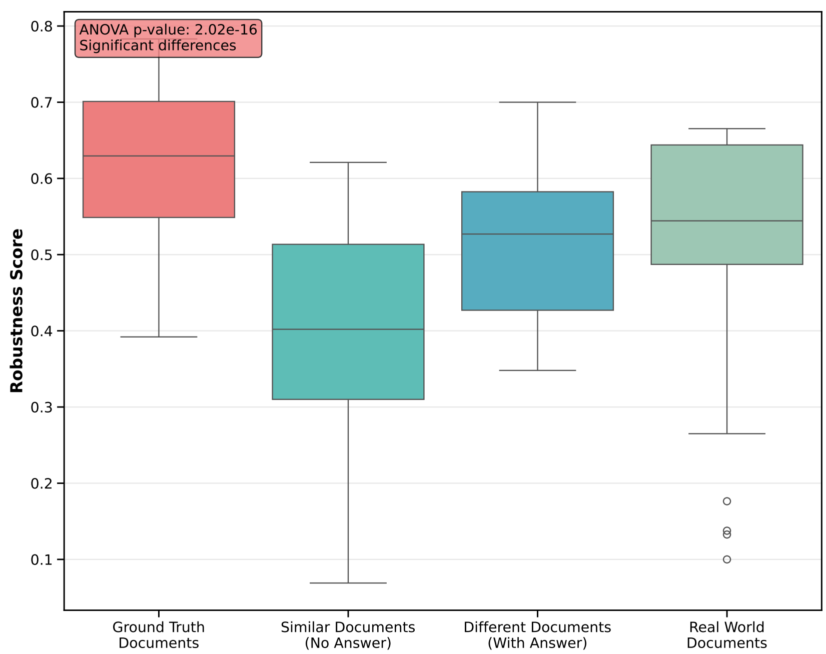

Figure 8: Robustness distribution differences across all document perturbations (including real-world retrieval results) given original query. Document perturbations significantly affect the robustness of RAG system. All kinds of document perturbation reveal significant drop compared to original ground truth documents results. RAG systems exhibit considerable difficulty in accurately leveraging their internal knowledge (or refuse to answer) when the provided documents do not contain the correct answer.

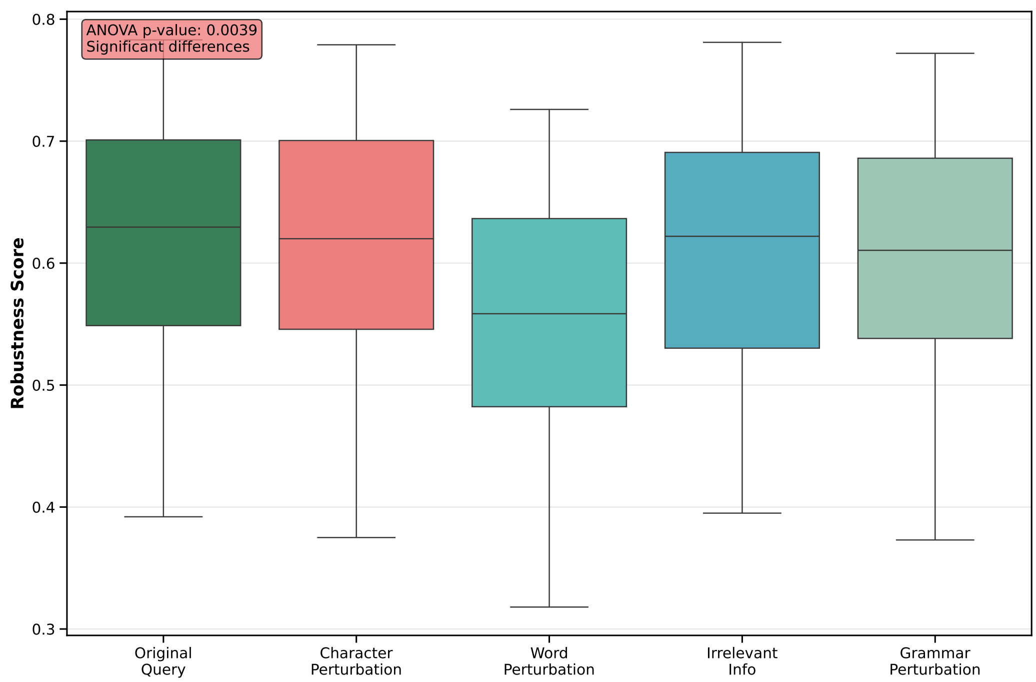

Figure 9: Robustness distribution differences across all query perturbations given original ground truth documents. Compared to document perturbations, query perturbations have a relatively smaller impact on the robustness of RAG systems (higher ANOVA p-value). Among various types of query perturbations, word-level perturbations exert a significantly greater influence on robustness than other forms.