IRB: Automated Generation of Robust Factuality Benchmarks

Anonymous ACL submission

Abstract

Static benchmarks for RAG systems often suffer from rapid saturation and require significant manual effort to maintain robustness. To address this, we present IRB, a framework for automatically generating benchmarks to evaluate the factuality of RAG systems. IRB employs a structured generation pipeline utilizing *factual scaffold* and *algorithmic scaffold*. We utilize IRB to construct a benchmark and evaluate frontier LLMs and retrievers. Our results demonstrate that IRB poses a significant challenge for frontier LLMs in the closed-book setting. Furthermore, our evaluation suggests that reasoning LLMs are more reliable, and that improving the retrieval component may yield more cost-effective gains in RAG system correctness than scaling the generator.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable general-purpose capabilities, but they remain susceptible to generating factually inaccurate or hallucinated content (Rawte et al., 2023; Xu et al., 2024). Retrieval-Augmented Generation (RAG) has emerged as a prominent technique to mitigate this issue by grounding model responses in external, verifiable knowledge sources (Lewis et al., 2020; Shuster et al., 2021). Consequently, developing rigorous benchmarks to evaluate RAG systems has received much attention from the research community.

This rapid evolution of LLMs, however, introduces a significant challenge to existing evaluation frameworks. Newer models, trained on vast, web-scale datasets, often memorize benchmark data (Sainz et al., 2023; Golchin and Surdeanu, 2023; Dong et al., 2024). This data contamination compromises evaluation reliability, as it becomes difficult to disentangle whether a model is utilizing the retrieved context or recalling an answer from its parametric knowledge (Longpre et al., 2021; Chen et al., 2024b). Consequently, to ensure that evaluation remains meaningful, there is a persistent need to develop new, challenging benchmarks that can effectively test the capabilities of even state-of-the-art models.

Such an endeavor, however, is traditionally labor-intensive and expensive, as most dataset creation requires significant human annotation. Some benchmarks are constructed entirely by human annotators (Kwiatkowski et al., 2019; Vu et al., 2023; Pham et al., 2025), while others use humans-in-the-loop to filter, refine low-quality samples generated by LLMs (Li et al., 2023; Chen et al., 2024b). A cheaper and more scalable alternative is to rely entirely on LLMs for data curation. Yet, previous work exploring this fully-automated approaches (Zhu et al., 2024; Filice et al., 2025) overlook explicit mechanisms to control the generation process. This absence of guidance not only can result in unfaithful or low-quality samples, but also provide little to no control over the types of questions that are generated.

To address these limitations, we introduce **IRB**, a framework that automates benchmark creation through structured, scaffolded generation. Our pipeline mitigates the instability of purely neural generation by introducing two distinct forms of guidance. First, we utilize a *factual scaffold* derived from human-written citing sentences in Wikipedia. This ensures that generated questions and answers are grounded in fact, as every sentence is supported by human-verified evidence. Furthermore, these citations preserve the complex information needs inherent in human composition, ranging from temporal sensitivity (requiring up-to-date evidence) to cross-lingual reasoning. In this design, the sentence provides the factual basis for QA generation, while the cited documents serve as the retrieval ground truth.

Second, we impose an *algorithmic scaffold* to guide the synthesis of question-answer (QA) pairs.

Question	Attributes				Answer
	Language	Freshness	Topics	Hops	
Who is the winningest head coach of the Mississippi State Bulldogs women's soccer team from 2019 to 2024 who compiled an overall record of 62 wins, 35 losses, and 18 draws?	En	2024	Culture, Geo	1	James Armstrong
What is the name of the largest gas supplier company in a specific region that experienced harsh winter conditions, heating shortages, and isolated cases of hypothermia during the 2025 Moldovan energy crisis–which occurred after the cessation of Russian gas supplies at the end of December 2024–and that cut off the gas supply roughly 9 months ago to several buildings of the Moldovan authorities located in the Security Zone and in the city of Bender (Tighina)?	Ru	2025	Geo, Culture	2	Tiraspoltransgaz
What geological formation located in Peru contains fossils of Sinoeocrinus lui?	En	2024	STEM	1	False-premise question

Table 1: Sample questions and answers generated by IRB, using 29 September 2025 as the reference date.

By transforming source text into an intermediate knowledge graph, we utilize graph-guided generation to enforce constraints on the LLM. This mechanism allows us to programmatically dictate output complexity, enabling the precise construction of multi-hop reasoning chains, lexical paraphrases, and false-premise traps.

Using the IRB framework, we generate IRB1K, to evaluate frontier LLMs and retrievers. Our evaluation reveals that IRB1K poses a significant challenge for frontier models in the closed-book setting. Additionally, while retrieval acts as an "equalizer" that aligns the correctness of various frontier LLMs, reasoning models prove to be more reliable; particularly in adversarial settings involving incorrect retrieval, false-premise questions and internal-external knowledge conflicts. Finally, our results suggest that given the high baseline capabilities of current frontier LLMs, larger and more cost-effective gains in RAG system correctness are best achieved by improving the retriever component.

We summarize our contributions as follows: 1) We propose IRB, a framework for generating granular RAG evaluation benchmarks; 2) we assess the performance of frontier LLMs and retrievers on IRB-generated datasets; and 3) we will open-source our code to support future research.

2 Related Work

Benchmarking factuality of LLMs. IRB aligns with existing literature on assessing the ability of LLMs to answer factual questions. Early benchmarks in this domain include TriviaQA (Joshi et al., 2017), Natural Questions (Kwiatkowski et al., 2019), HotpotQA (Yang et al., 2018), 2WikiMultiHopQA (Ho et al., 2020), and FEVER (Thorne et al., 2018). However, as model performance on these datasets has saturated and concerns regarding data contamination have grown, a new wave of benchmarks has emerged (Lin et al., 2022; Niu et al., 2023; Li et al., 2023; Vu et al., 2023; Krishna et al., 2024; Hsieh et al., 2024; Yang et al., 2024; Wei et al., 2024b,a; Pham et al., 2025; Bang et al., 2025; Choi et al., 2025). While these recent efforts address distinct challenges, such as time-sensitive questions (Vu et al., 2023; Yang et al., 2024; Pham et al., 2025), open-ended generation (Wei et al., 2024b), and citation attribution (Choi et al., 2025), they remain largely static or rely on manual updates. IRB distinguishes itself via an automatic generation pipeline that ensures meaningful evaluation by enabling on-demand updates.

Automatic benchmark generation. Prior work has explored automating the creation of RAG benchmarks to reduce reliance on manual annotation. Frameworks like RAGEval (Zhu et al., 2024) and DataMorgana (Filice et al., 2025) rely on the instruction-following capabilities of LLMs to synthesize questions across various personas. Alternatively, FreshStack (Thakur et al., 2025) utilizes human-generated Q&A pairs from StackOverflow, employing a search engine to fetch evidence which is subsequently validated by an LLM. In contrast, IRB grounds its generation in human-written citing sentences, where cited websites serve as evidence. This approach better reflects authentic human information-seeking patterns by leveraging documents that human writers explicitly selected as evidence. Furthermore, unlike methods that depend solely on LLM instruction-following, IRB achieves greater controllability in question-answer generation through a structured knowledge graph-

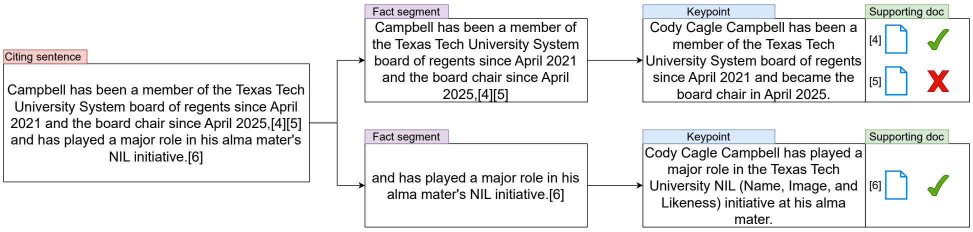


Figure 1: Fact & supporting documents extraction from a citing sentence. The resulting fact contains two keypoints, each with different supporting documents. Groundedness check is performed on each (keypoint, document) pair.

based algorithm.

3 Methodology

IRB takes as input a collection of Wikipedia articles and outputs a benchmark dataset, containing a set of queries, their attributes, ground-truth answers, a reference corpus, and query relevance judgments (qrels). To ensure high-quality, factually-grounded benchmark dataset, we rely on two "scaffolds" to guardrail the benchmark generation process. First, we generate facts and supporting documents using human-written *citing sentences* from Wikipedia as **factual scaffold**. These evidence-backed facts serve as the reliable base upon which question-answer (QA) pairs are generated.

Then, we generate QA pairs from the evidence-backed facts using **algorithmic scaffold**. Specifically, we devised a knowledge graph-based question generation algorithm that takes a fact as input and generates high-quality question-answer pairs in a structured and controllable manner.

3.1 Facts & Supporting Documents Extraction

Given a Wikipedia article, our pipeline begins by extracting *citing sentences*. However, we observe that many extractions, such as cited section titles, lack informational value. To filter this noise, we apply a heuristic that retains only syntactically complete sentences, i.e., those containing a subject, predicate, and object.

Next, it is worth noting that a single citing sentence may contain multiple distinct claims, separated by the position of their inline citations. For example, in Figure 1, the citing sentence encompasses two claims, each supported by a different set of documents. Therefore, we split such sentences into distinct segments, each representing a more atomic fact. However, generating good facts requires more than just splitting the citing sentence into segments. The segment may be ambiguous (as exemplified by the two segments in Figure 1), making them unfit for generating well-formed questions and difficult to reliably verify against source text. To address this, we prompt an LLM to decontextualize the segments into a more self-contained version of themselves, which we call *keypoints*. This step ensures the content is explicit enough for the subsequent groundedness check. For decontextualization, the LLM is provided with necessary information, including title and abstract of the source Wikipedia page, as well as the surrounding sentences. We provide the full prompt, which is inspired by (Gunjal and Durrett, 2024), in Figure 8.

Each keypoint is associated with a set of corresponding supporting documents, i.e., the set of URLs found within its original citation group. We retrieve the content, language, and publication date for each of these documents using `Trafilatura` (Barbaresi, 2021), `fast-langdetect` and `htmldate` (Barbaresi, 2020) respectively.

Groundedness check. It is natural to assume that all (keypoint, document) pairs are reliable because they originate from human-written citations. However, we find that a non-negligible portion of facts are not actually grounded in their retrieved documents. The primary cause is rarely human error but rather technical failures, including 1) availability issues (offline websites), 2) dynamic content leading to incomplete text retrieval, and 3) multimodal dependencies, where verification requires non-textual data (e.g., video or audio) beyond our text-only pipeline. To address this, we verify all pairs by prompting an LLM to assess if the keypoint is supported, as shown in Figure 1. Only keypoints supported by at least one document are retained. Consequently, the final output is a *Fact* composed of validated keypoints, each explicitly linked to its specific supporting evidence. The prompt for the groundedness check is provided in Figure 9.

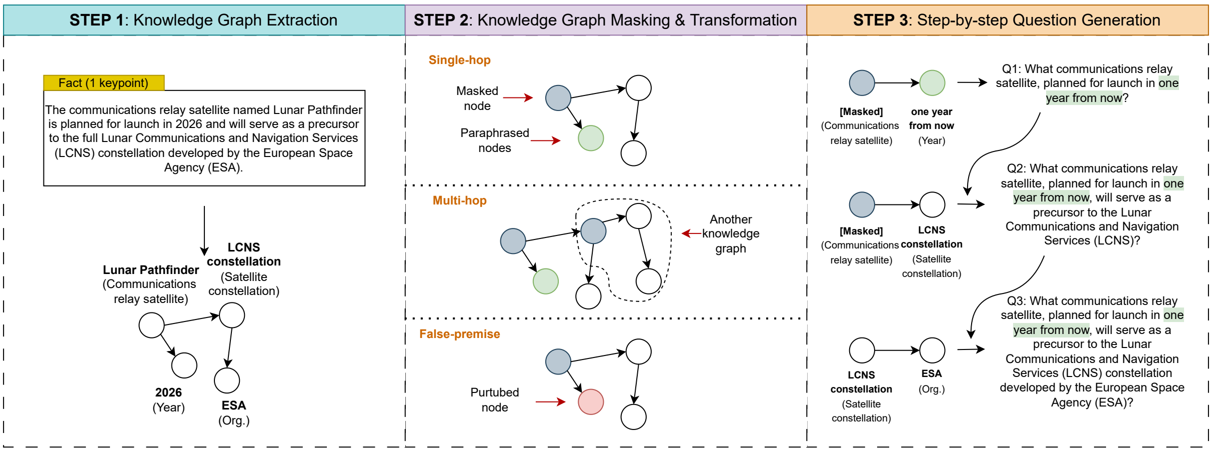


Figure 2: The question generation process operates in three stages. First, a fact is structured into a knowledge graph. Subsequently, this graph is transformed into up to three distinct variants, namely single-hop, multi-hop, and false-premise. Finally, each variant is utilized to generate a corresponding natural language question. In this figure, we only show the question generation process for the single-hop variant.

3.2 Generation of Question-Answer Pairs

Unlike facts and supporting documents, which rely on the factual scaffold of human-written citing sentences, Question-Answer pairs are generated using an algorithmic scaffold. While a simplistic alternative would be to prompt an LLM to generate QA pairs directly from extracted facts, this approach offers *little to no control over the generation process*. Specifically, direct prompting prevents us from systematically dictating the question's topic or calibrating its reasoning complexity (e.g., specifying the number of hops for multi-hop questions).

To address these limitations, we propose a graph-based generation pipeline, inspired by techniques in controllable question generation (Cheng et al., 2021). As illustrated in Figure 2, this process proceeds in three distinct steps: *knowledge graph construction*, *knowledge graph masking & transformation*, and *step-by-step question generation*.

3.2.1 Knowledge graph construction

The first step in our algorithm is to convert each fact from unstructured text into a structured knowledge graph (KG). This graph is defined as a collection of triplets (head, relation, tail), along with the node type of each head and tail. For transparency, we provide the full prompt in Figure 10.

To ensure the graph's content sufficiently covers the input answer, we validate it with a simple coverage check. This metric measures the proportion of words from the keypoints that are covered by the heads, relations, and tails of the extracted knowledge graph. Only graphs that exceed a set coverage threshold are accepted.

3.2.2 Knowledge graph masking & transformation

Next, we perform knowledge graph masking and transformation. This step dictates the answer and the complexity of the question to be generated. For each graph, we produce three masked versions corresponding to distinct question types namely single-hop, multi-hop, and false-premise.

Single-hop. In this setting, only one node is masked, which serves as the answer for the generated question. We select the node to mask by identifying the first *maskable* head. A node is considered maskable if it satisfies specific criteria, such as being a named entity and appearing in all keypoints (we detail all criteria in §A.1).

Multi-hop. We construct masked graphs for multi-hop questions by merging two single-hop instances. Specifically, if the masked node of KG_2 appears as an unmasked node in KG_1, the two graphs are combined to form a composite graph KG'. This composite graph is used to generate a multi-hop question where the final answer aligns with the target of KG_1, but the reasoning process requires first resolving the sub-question implied by KG_2. In this work, we focus exclusively on two-hop reasoning chains.

Node paraphrasing. To increase question difficulty and prevent trivial solutions via exact keyword matching, we apply paraphrasing to unmasked nodes in both single-hop and multi-hop graphs. We employ rule-based transformations for specific node types, such as *Person* (via name abbreviation, e.g., Cristiano Ronaldo → C. Ronaldo) and *Date* (via relative time transformation, e.g., 30 November 2024" → 9 months ago"[1]). Table 8 details all paraphrasable node types and their corresponding transformation rules.

False-premise generation. Beyond paraphrasing, we also perturb nodes to inject false premises into the generated questions. To achieve this, we employ rule-based transformations on the same node types used for paraphrasing. For *Person* nodes, we substitute the surname with a randomized alternative (e.g., "Cristiano Ronaldo" → "C. Smith"). Similarly, for *Date* nodes, we apply a distorted relative time transformation (e.g., mapping "30 November 2024" → "1 year 4 months ago", intentionally deviating from the correct interval). We detail all transformable node types and their transformation rules in Table 9.

3.2.3 Step-by-step question generation

We subsequently perform step-by-step question generation over the masked and transformed knowledge graph. We iteratively generate an intermediate question conditioned on the current triplet, as well as the history of previous triplets and generated questions. If the current triplet contains a masked node, the algorithm replaces the entity with a modified clause. Conversely, if no masking is present, the algorithm incorporates the triplet's information to refine the question's specificity. We provide the full prompt in Figure 11.

Question answerability check & refinement. The generation of questions with multiple potential answers is an inevitable byproduct of the process. This arises not from limitations in the graph-guided generation algorithm, but from inherent ambiguities within the extracted facts. As these ambiguities can often be identified using general world knowledge, we prompt an LLM to assess whether a generated question admits a unique answer. Additionally, as stepwise generation may result in disjointed or verbose phrasing, we implement a final refinement module by prompting an LLM. We provide the full prompts in Figure 12 and 13.

3.3 Attributes

Each sample generated by IRB is annotated with attributes derived from distinct sources. First, we extract evidence attributes, namely *publication date* and *language*, from supporting documents; these define the specific search and interpretation capabilities a system needs. Second, we record question characteristics, including *number of hops* and *false-premise status*, during the generation step to quantify the question's difficulty. Finally, we include the topic, mined from the source Wikipedia headers. These annotations enable the flexible construction of evaluation subsets. For example, to assess multi-hop reasoning in a cross-lingual setting, one can simply select samples whose questions are multi-hop and require non-English supporting evidence.

[1]Calculated relative to a reference date of 29 September 2025.

3.4 Evaluation Protocols

Retriever evaluation metrics. We adhere to standard text retrieval protocols, utilizing nDCG@k to evaluate performance, where k denotes the number of documents retrieved as context for answer generation. To establish relevance judgments (qrels), we define relevant documents as the evidence containing the facts from which a query was generated. This definition extends to false-premise queries, as the model requires the underlying factual evidence to correctly identify and refute the false premise.

Generator evaluation metrics. Given a query, the generator produces an answer either without external context (closed-book setting) or conditioned on the top k retrieved documents (RAG setting). We evaluate the quality of these generations by comparing them to the ground truth. Following (Wei et al., 2024a; Pham et al., 2025), we employ an LLM-based evaluator to classify each prediction as "correct", "incorrect", or "not attempted".

4 Experiments

4.1 IRB1K

4.1.1 Implementation Details

For our later experiments, we generated a RAG benchmark from a corpus of 2,000 recent Wikipedia articles (1,000 from 2024 and 1,000 from 2025) from the September 29, 2025 Wikipedia dump. The pipeline yielded 1,838 questions in total, from which we sample 1,000 questions for evaluation due to resource limitations. For convenience, we name this dataset `IRB1K`. The dataset is generated using `GPT-4.1-mini`, at a total cost of about $18 and a total runtime of 16 hours[2]. We provide the detailed statistics in Table 6.

Outcome Category	Specific Criterion	# Samples	Percentage (%)
Success	All criteria satisfied	189	94.5
	1. Question Malformed	10	5
Failure Mode	2. Fact Unnecessary	0	0
	3. Answer Invalid	1	0.5
Total		200	100

Table 2: Quality assessment results.

4.1.2 Quality Assessment

To validate the quality of the dataset generated by our proposed framework, we conduct a manual evaluation on 200 randomly sampled QA pairs. The evaluation was performed by volunteer undergraduate and graduate students in Computer Science and Electrical Engineering in the United States. To ensure accuracy, we ask two annotators to first independently assess each sample; then, in case of disagreement, they discussed the sample to reach a final decision. Each sample was evaluated against three criteria in the following order[3].

Question well-formedness: The question must be specific, allowing it to be answered. Ill-formed questions can confuse the RAG system into finding the wrong answer, leading to unreliable evaluation signals.

Fact necessity: Answering the question must require integrating information from all parts of the compositional fact. This assesses if retrieval of all supporting evidence is necessary.

Answer validity: The ground-truth answer must be correct and unique. Conversely, when an answer is wrong or not unique (is one of the possible answers), evaluation of RAG systems would be unreliable.

Table 2 presents the evaluation results, which strongly validate the IRB framework. Specifically, approximately 95% of the evaluated samples satisfy all three criteria. Analysis of the remaining 5% reveals that the majority of errors stem from malformed questions attributed to the knowledge graph construction process (e.g., misidentified node types or information loss during conversion). Only a single instance exhibited the issue of multiple answers.

4.2 Experiment Setup

In our experiments, we assess a straightforward RAG pipeline with a simple prompt similar to (Yang et al., 2024), where we also encourage brief answers and saying "I don't know" or "False premise question" (see Figure 14 and 15). Unless specified otherwise, for each query, we retrieve the top $k = 5$ documents and formatted them similarly to `FRESH-PROMPT` (Vu et al., 2023), where aside from the main content, we include the source webpage and the publication date. Crucially, we set the "current date" in the prompt to 29 September 2025, corresponding to our Wikipedia dump date. Reported results represent a single run.

Retriever. For our main analysis, we employ OpenAI's `text-embedding-3-small` as our primary retriever. However, we do evaluate other retrievers, namely `BM25` (Robertson et al., 2009), `BGE-M3` (Chen et al., 2024a) and `E5-base` (Wang et al., 2022). The documents in the corpus are split into chunks of 512 tokens. During inference, we retrieve the top 5 documents, where their rankings are determined according to their highest ranked passage, a setting similar to (Kaszkiel and Zobel, 1997).

Generator. We evaluate a diverse set of both proprietary and open-source LLMs. For reasoning-focused models, these include proprietary models namely `GPT-5-mini` and `GPT-5` (OpenAI, 2025b), as well as leading open-source models like `gpt-oss-120b` (OpenAI et al., 2025) and `DeepSeek-R1` (Guo et al., 2025). We utilize the medium reasoning mode for GPT models and allocate a reasoning budget of 2048 tokens for the others. For comparison, we also evaluate non-reasoning models, including proprietary models that are `GPT-4.1-mini`, `GPT-4.1` (OpenAI, 2025a), and open-source models namely `Llama-3.3-70B` (Meta AI, 2024) and `Llama-4-Scout` (Meta AI, 2025).

4.3 Evaluation Results

4.3.1 The necessity of retrieval for IRB1K

Firstly, we would like to ensure that questions generated by IRB are not trivial and can be answered without access to search results. As illustrated in Table 4, IRB1K presents a substantial challenge to frontiers LLMs when operating without access to external information. While models with more recent knowledge cutoffs (e.g. GPT-5 and GPT-5 mini) exhibit marginal improvements, the overall low performance confirms that our framework successfully generates non-trivial queries.

4.3.2 Effectiveness of retrievers

Table 3 summarizes the effectiveness of retrievers on IRB1K. Unsurprisingly, we observe that effec-

[2] We call API in sequence and not in parallel

[3] The full rubric is provided in §C

	Valid-premise										False-premise
	Language		Freshness		Topic				# Hops		
	English	Cross	2024	2025	Culture	Geo	H & S	STEM	Single	Multi	
First stage retrieval											
BM25	**75.6**	13.4	58.4	54.1	52.2	50.3	75.5	**80.3**	56	**56.7**	56.4
text-embedding-3-small	69.4	50.2	66.5	60.7	61.3	61.6	73.6	78.9	64.5	49.9	59.2
E5-base-v2	67.9	34	63	52.4	53.4	54.7	72.6	75.5	57.3	56.2	56.1
BGE-m3	69.5	**53.6**	**68.9**	**60.8**	**61.7**	**63**	**75.9**	78.4	**65.2**	56.6	**66.5**
Reranking with Cohere Rerank 3.5											
BM25	**83.6**	32.8	72.1	63.8	65.2	62.7	83.3	88.9	67.7	66.4	68.4
text-embedding-3-small	82.8	67.9	81.7	75.1	75.8	75.8	85.6	**91.6**	78.9	68.6	79.3
E5-base-v2	82.2	60	78.4	72.5	72.6	72.1	84.8	**91.6**	75.7	69.2	76.4
BGE-m3	82.5	**73.1**	**82.8**	**76.8**	**77.1**	**77.6**	**85.8**	91.1	**80.3**	**70.4**	**80.2**

Table 3: Performance (nDCG@5) of retrievers on IRB1K. We **bold** and underline the best and second-best results in each setting. H & S is short for History & Society

	Valid-premise										False-premise
	Language		Freshness		Topic				# Hops		
	English	Cross	2024	2025	Culture	Geo	H & S	STEM	Single	Multi	
Closed-book performance											
GPT-4.1 mini	11.4	9.7	11.5	10.4	5.9	10.4	20.3	22.7	11.1	8.1	10
GPT-4.1	30.6	21.4	32	24.1	18.4	26.1	40.4	43.2	28.3	20.2	1
GPT-5 mini medium	14.8	9.1	16.1	10.4	5.8	10.8	26.2	28.2	12.9	14.5	8
GPT-5 medium	**41.9**	**31.5**	**48.4**	**30.4**	**29.8**	**36.8**	**54.9**	**58.2**	**39.4**	**29**	24
Llama-4-Scout	8.5	4.8	10	5.1	3.2	5.1	14	17.3	7.7	3.2	31
Llama-3.3-70B	14.8	9.5	16.7	10.1	4.9	9.5	29	34.5	13.1	12.9	12.2
gpt-oss-120b	16.3	11.9	16.1	13.9	5.7	14.3	28	28.2	15.4	8.9	11.5
DeepSeek-R1	11.3	8.5	13.4	7.9	7.5	9.1	14.7	17.7	10.9	4.8	**39.5**
RAG performance											
GPT-4.1 mini	82.8	72.2	76.8	81.7	77.9	79.1	85	81.8	80.6	66.1	1.5
GPT-4.1	**84.9**	73.4	80.1	82.4	79.1	80.6	**86.4**	84.5	82.5	66.9	2.8
GPT-5 mini medium	83.1	72.8	77.6	81.8	77.4	80.1	84.8	82.3	81	66.1	35
GPT-5 medium	**84.9**	**77.6**	**81.4**	**83.5**	**80.9**	**82.6**	85	85.5	**83.5**	71	**50.5**
Llama-4-Scout	76	62.9	71.9	71.9	70.2	70.3	75.9	80	73.5	52.4	20.5
Llama-3.3-70B	78.6	62.9	73.6	73.6	72.7	71.3	80.4	80.9	75.2	54.8	5
gpt-oss-120b	84.4	69.8	79.9	79.7	78.3	78	86.2	**86.4**	80.4	**73.4**	14.5
DeepSeek-R1	79.9	71.4	76.2	78.1	75.1	77.6	80.6	80.5	78.5	62.1	22.8

Table 4: Evaluation of LLM **correctness** on IRB1K. Results indicate that while RAG bridges the performance gap present in closed-book settings, models continue to struggle with specific complexities. Notably, performance degrades in cross-lingual, fresh information, multi-hop, and false-premise scenarios. We **bold** and underline the best and second-best results in each setting.

	Correct retrieval			Incorrect retrieval		
	C	I	N	C	I	N
GPT-4.1 mini	71.8	26.3	1.9	38.4	35.3	26.3
GPT-4.1	74.2	22.4	3.4	38.4	31	30.6
GPT-5 mini medium	79.4	16.6	4	45.3	**15.4**	39.3
GPT-5 medium	**82.7**	**15.5**	1.8	**55.4**	18.1	26.5
Llama-4-Scout	70.2	25.7	4.1	33.3	36.6	30.1
Llama-3.3-70B	69.7	20.3	10	29.7	18.3	52
gpt-oss-120b	74.4	22.1	3.5	41.3	25.2	33.5
DeepSeek-R1	76.2	20	3.8	35.7	25.7	38.6

Table 5: Effectiveness of RAG systems when retriever is correct and incorrect. C, I, N are short for "correct", "incorrect" and "not attempted" respectively

tiveness of retrievers varies across queries of different attributes. Specifically, retrievers struggle with questions 1) in the multilingual setting, where an English question is used to search for supporting documents in other languages; 2) involving fresh information; 3) with multiple reasoning hops. Interestingly, topic-based analysis indicates that retrievers remain most effective within the STEM domain, while struggling to maintain similar ranking performance in other topical categories. Furthermore, BM25 remains a robust baseline, as it outperforms all neural retrievers on monolingual English tasks, though it naturally underperforms in the cross-lingual setting.

4.3.3 Effectiveness of RAG

As shown in Table 4, the correctness of all evaluated LLMs on IRB1K improves substantially when provided with retrieved context.

Retrieval acts as an "equalizer". In the absence of retrieval, performance disparities between LLMs are pronounced. Specifically, the gap between the highest (`GPT-5`) and lowest-performing (`Llama-4-Scout`) models is nearly four-fold. Furthermore, individual model performance fluctuates across different question types. However, when retrieval is integrated, the performance gap between models shrinks significantly, and all models exhibit greater consistency across tasks.

RAG correctness is bottlenecked by the retriever. Table 5 compares system effectiveness under correct versus incorrect retrieval conditions. We observe that when retrieval is accurate, i.e. all gold documents are retrieved within the top-5 contexts, all LLMs achieve high performance. Conversely, incorrect retrieval causes a sharp decline in overall system correctness. Crucially, however, this drop in performance is not due to frontier LLMs producing more hallucinations. Rather, these models often successfully detect irrelevant contexts and refuse to answer, as evidenced by the increase in "not attempted" rates. Furthermore, as discussed in §B.3, when faced with incorrect retrieval results, models possessing the correct internal information effectively prioritize their parametric knowledge over the provided context. These observations indicate that frontier LLMs are already highly reliable readers. Thus, future improvements in RAG systems should prioritize the retrieval component over the generator.

False-premise questions pose a persistent challenge. We observe that all models struggle to identify false-premise questions, particularly in the absence of retrieval. While `Llama-4-Scout` and `DeepSeek-R1` appear to be exceptions with higher detection rates, we observe that this is an artifact of model bias, as these models exhibit a tendency to label *any* query as a "False-premise question", resulting in a disproportionately high false positive rate.

Reasoning models are more reliable. Reasoning models not only exhibit higher correctness but also generate fewer incorrect answers, as observed in Table 4 and 5. Furthermore, reasoning models are significantly more reliable in handling incorrect retrieval contexts. This is evidenced by their lower error rates when retrieval is wrong, compared to their non-reasoning counterparts. Furthermore, while all models struggle to detect false-premise questions, reasoning LLMs are generally more reliable.

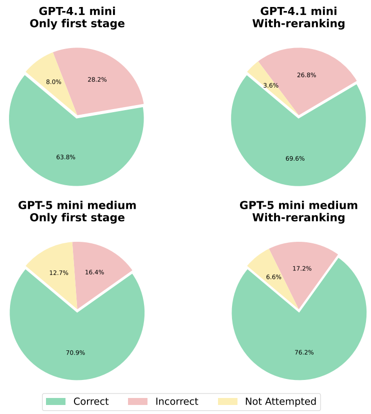


Figure 3: Effect of reranking.

Reranking enhances performance. We investigate how reranking retrieval results from `text-embedding-3-small` impacts the effectiveness of RAG systems. As expected, we find that using reranked retrieval contexts consistently leads to gains in LLM correctness. This supports our earlier argument that future development should focus heavily on the retriever side.

Additional experiment results. Due to space limitations, we provide detailed supplementary analyses in the appendices. §B.2 examines reasoning token allocation, revealing that reasoning LLMs expend more tokens on multi-hop questions and in closed-book settings. §B.3 investigates the interplay between models' internal knowledge and retrieved evidence, demonstrating that models remain robust even when conflicts exist between the two sources.

5 Conclusion

We introduce IRB, a framework for automatically generating benchmarks for evaluating RAG systems. IRB automates the benchmark generation process through structured and scaffolded generation to produce high-quality datasets that reliably test RAG systems. Overall, our evaluations show that: 1) IRB generates challenging benchmarks that challenge frontier LLMs without access to external knowledge; 2) frontier LLMs, especially reasoning models, are reliable; and 3) retrievers should receive more attention, as improving retrieval is essential for making RAG systems more robust.

Limitations

Restriction to Short Answers. IRB is designed to generate questions that elicit short, specific answers, following the approach of (Wei et al., 2024a; Pham et al., 2025). Consequently, benchmarks produced by our framework are not suitable for evaluating long-form text generation capabilities.

Restriction to Wikipedia. As noted, IRB relies on human-written sentences from Wikipedia to generate questions, answers, and evidence. While other domains (i.e. scientific literature and legal case law) also contain rich citation structures, the semantic function of their citations differs fundamentally from that of Wikipedia. Consequently, adapting IRB to these domains would require a substantial redesign of the pipeline. We therefore leave the extension of our framework to these alternative data sources for future work.

Ethical Considerations

While we apply strict filtering to remove low-quality or harmful content, the IRB benchmark is derived from and represents the current state of Wikipedia. As such, we cannot guarantee the complete exclusion of sensitive information or copyrighted material within the questions and references. Researchers should use the dataset with the understanding that it mirrors the characteristics and potential liabilities of its source corpus.

References

Yejin Bang, Ziwei Ji, Alan Schelten, Anthony Hartshorn, Tara Fowler, Cheng Zhang, Nicola Cancedda, and Pascale Fung. 2025. Hallulens: Llm hallucination benchmark. *arXiv preprint arXiv:2504.17550*.

Adrien Barbaresi. 2020. htmldate: A Python package to extract publication dates from web pages. *Journal of Open Source Software*, 5(51):2439.

Adrien Barbaresi. 2021. Trafilatura: A web scraping library and command-line tool for text discovery and extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 122–131.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024a. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024b. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17754–17762.

Yi Cheng, Siyao Li, Bang Liu, Ruihui Zhao, Sujian Li, Chenghua Lin, and Yefeng Zheng. 2021. Guiding the growth: Difficulty-controllable question generation through step-by-step rewriting. *arXiv preprint arXiv:2105.11698*.

Yee Man Choi, Xuehang Guo, Yi R Fung, and Qingyun Wang. 2025. Citeguard: Faithful citation attribution for llms via retrieval-augmented validation. *arXiv preprint arXiv:2510.17853*.

Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. 2024. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12039–12050, Bangkok, Thailand. Association for Computational Linguistics.

Simone Filice, Guy Horowitz, David Carmel, Zohar Karnin, Liane Lewin-Eytan, and Yoelle Maarek. 2025. Generating diverse q&a benchmarks for rag evaluation with datamorgana. *arXiv preprint arXiv:2501.12789*.

Shahriar Golchin and Mihai Surdeanu. 2023. Time travel in llms: Tracing data contamination in large language models. *arXiv preprint arXiv:2308.08493*.

Anisha Gunjal and Greg Durrett. 2024. Molecular facts: Desiderata for decontextualization in llm fact verification. *arXiv preprint arXiv:2406.20079*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, Adriane Boyd, and 1 others. 2020. spacy: Industrial-strength natural language processing in python.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Marcin Kaszkiel and Justin Zobel. 1997. Passage retrieval revisited. In *ACM SIGIR Forum*, volume 31, pages 178–185. ACM New York, NY, USA.

Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. 2024. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. *arXiv preprint arXiv:2409.12941*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 3214–3252.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Meta AI. 2024. Llama 3.3 model cards and prompt formats.

Meta AI. 2025. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2023. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. *arXiv preprint arXiv:2401.00396*.

OpenAI. 2025a. Introducing GPT-4.1. `https://openai.com/index/gpt-4-1/`.

OpenAI. 2025b. Introducing GPT-5. `https://openai.com/index/introducing-gpt-5/`.

Sandhini Agarwal OpenAI, Jason Ai Lama Ahmad, Andy Applebaum Sam Altman, Rahul K Arora Edwin Arbus, Bowen Baker Yu Bai, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Thinh Pham, Nguyen Nguyen, Pratibha Zunjare, Weiyuan Chen, Yu-Min Tseng, and Tu Vu. 2025. Sealqa: Raising the bar for reasoning in search-augmented language models. *arXiv preprint arXiv:2506.01062*.

Vipula Rawte, Swagata Chakraborty, Agnibh Pathak, Anubhav Sarkar, SM Towhidul Islam Tonmoy, Aman Chadha, Amit Sheth, and Amitava Das. 2023. The troubling emergence of hallucination in large language models-an extensive definition, quantification, and prescriptive remediations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2541–2573.

Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787, Singapore. Association for Computational Linguistics.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Nandan Thakur, Jimmy Lin, Sam Havens, Michael Carbin, Omar Khattab, and Andrew Drozdov. 2025. Freshstack: Building realistic benchmarks for evaluating retrieval on technical documents. *arXiv preprint arXiv:2504.13128*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and 1 others. 2023. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint arXiv:2310.03214*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. 2024a. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*.

Attribute of sample			Percentage
Valid-premise	Language	English	54.8
		Cross	25.2
	Freshness	2024	36.6
		2025	43.4
	Topics	Culture	51.4
		Geo	49.2
		H & S	21.4
		STEM	11
	# Hops	Single	73.8
		Multi	6.2
False-premise			20

Table 6: IRB1K statistics. H & S is short for History & Society.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, and 1 others. 2024b. Long-form factuality in large language models. *Advances in Neural Information Processing Systems*, 37:80756–80827.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.

Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze D Gui, Ziran W Jiang, Ziyu Jiang, and 1 others. 2024. Crag-comprehensive rag benchmark. *Advances in Neural Information Processing Systems*, 37:10470–10490.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Kunlun Zhu, Yifan Luo, Dingling Xu, Yukun Yan, Zhenghao Liu, Shi Yu, Ruobing Wang, Shuo Wang, Yishan Li, Nan Zhang, and 1 others. 2024. Rageval: Scenario specific rag evaluation dataset generation framework. *arXiv preprint arXiv:2408.01262*.

A Additional Details of IRB

A.1 Criteria for selecting masked node

As discussed in §3.2.2, when masking a knowledge graph, a candidate node must satisfy a specific set of criteria, which we detailed below.

- **Named Entity**: The node must be identified as a named entity. We utilize `spaCy` (Honnibal et al., 2020) to verify this property.
- **Coverage**: The node must appear in all keypoints associated with the fact.
- **Atomicity**: The node must not reference multiple entities. We validate this by ensuring the node text does not contain conjunctions such as "and", "&", or "et al".
- **Non-overlapping**: The node text must not be contained within other nodes. We employ fuzzy matching to enforce this constraint.
- **Non-exclusive**: The node must serve as the unique head and tail for any associated relation.

A.2 Entity transformation

A.3 Full prompts

Question generation. We provide the full prompts used in the question generation process. Specifically, Figure 10 details the knowledge graph extraction prompt; Figure 11 details the step-by-step question generation prompt; Figure 12 details the question answerability check; Figure 13 details the question refinement prompt.

Question answering. We provide the full prompts for generating answers in Figures 14 (with retrieval context) and 15 (without retrieval context).

LLM-based evaluation. We adopt the LLM-based evaluation prompt from (Pham et al., 2025). This prompt instructs an LLM to classify an answer as "correct", "incorrect", or "not attempted".

B Additional Experimental Results

B.1 Varying number of retrieval contexts

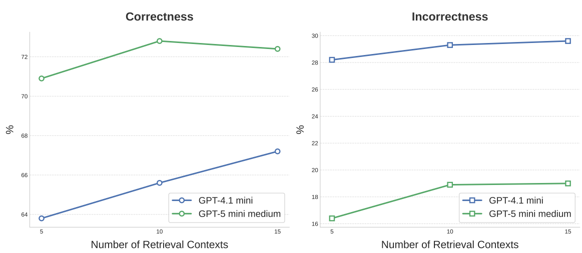


Figure 4: RAG performance (correctness and incorrectness) when the number of retrieval contexts are increased

Expanding the retrieval set presents a fundamental trade-off. On one hand, it maximizes the likelihood of capturing the gold document within the context window. On the other, it introduces noise and increases computational costs. We investigate how effectively LLMs utilize these expanded retrieval contexts. Given the computational costs of extensive evaluation, we restrict this analysis to two representative models, namely `GPT-4.1-mini` and `GPT-5-mini`, representing standard chat and reasoning models, respectively. Results are shown in Figure 4.

We observe that increasing the number of retrieved documents leads to a simultaneous rise in both correct and incorrect answers. Ideally, additional context should improve correctness while suppressing incorrectness. However, both models fail to demonstrate this behavior. The concurrent increase in positive and negative outcomes indicates that models become less likely to refuse to answer as context grows. This suggests an undesirable over-confidence, particularly when the expanded retrieval set fails to contain the correct document.

B.2 Reasoning efforts analysis

We investigate the amount of effort reasoning models allocate to questions with different attributes. We present these results in Figure 5. We observe that all models exhibit similar behavior across languages (English vs. Cross-lingual), reasoning hops (Single vs. Multi), and premise validity (with false-premise questions eliciting more reasoning tokens). In addition, perhaps unsurprisingly, when retrieval contexts are available, all models "think" less (producing fewer reasoning tokens), as they rely on the provided context rather than internal deduction.

B.3 Interplay between Parametric Knowledge and Retrieval

In this section, we analyze the interplay between an LLM's internal (parametric) knowledge and the external (non-parametric) evidence provided by the retriever. To disentangle these factors, we categorize instances into four distinct scenarios based on the correctness of the closed-book model and the retrieval results:

- **Redundant**: The model possesses the correct internal knowledge, and retrieval is accurate.
- **Resilience**: The model possesses the correct internal knowledge, but retrieval is incorrect.
- **Augmentation**: The model lacks internal knowledge, but retrieval is accurate.
- **Hopeless**: Both the internal knowledge and retrieval fail.

Results are presented in Table 7. Regarding valid-premise questions, surprisingly, no model achieves perfect correctness in the *Redundant* setting, despite both the LLM and retriever being correct. While all models perform reliably in the *Augmentation* setting, reasoning models prove more reliable in the *Resilience* setting, which again displays the noise resistance capabilities of these models. Finally, all models perform surprisingly well in the *Hopeless* setting. This suggests either: 1) the gold documents are not the only ones containing the answer; or 2) although non-gold documents are retrieved, they still provide enough context to help the LLM deduce or recall the answer.

	Redundant	Resilience	Augmentation	Hopeless
Valid-premise				
GPT-4.1 mini	96.4	76.9	87	54.2
GPT-4.1	97.4	68.9	87.7	52.7
GPT-5 mini medium	94.8	88.2	88.2	51.5
GPT-5 medium	96.9	87.9	84.4	53.7
Llama-4-Scout	88.5	77.8	82.3	40.2
Llama-3.3-70B	93.7	68.2	82.7	41.9
gpt-oss-120b	94.7	90	86.7	54.1
DeepSeek-R1	96.3	82.1	87	45.2
False-premise				
GPT-4.1 mini	6.2	25	0	1.6
GPT-4.1	0	100	3.4	0
GPT-5 mini medium	55.6	42.9	35.7	29.3
GPT-5 medium	85.2	69.2	46.3	36.5
Llama-4-Scout	18.6	52.6	14.1	21.7
Llama-3.3-70B	14.3	10	5.8	0
gpt-oss-120b	35.7	44.4	13.2	7.1
DeepSeek-R1	22.1	22.2	26.5	15.8

Table 7: Correctness of LLMs in four settings, namely Redundant, Resilience, Augmentation and Hopeless.

Turning to false-premise questions, we observe different behaviors. First, in the *Redundant* setting, models struggle significantly; even when their parametric knowledge correctly identifies the false premise, the presence of retrieval context appears to bias them toward attempting an answer rather than issuing a refusal. Conversely, models perform best in the *Resilience* setting. Here, the retriever's failure (retrieving irrelevant documents) paradoxically aids performance: because the retrieved context is obviously unhelpful, the model discards it and relies on its internal knowledge to correctly identify the false premise. Finally, similar to valid-premise questions, we observe unexpectedly high performance in the *Hopeless* setting.

C Human Evaluation Rubrics & Interface

We present the full text of the human evaluation rubrics in Figure 7 and provide a screenshot of the annotation interface in Figure 6.

Entity	Rule	Example
Person	First and middle name abbreviation	Cristiano Ronaldo → C. Ronaldo
Date	Relative time from a reference date	30 Nov 2024 → 9 months ago
Country	Visual substitution with flag icon	Japan → the country whose flag is
Quantity	Verbalization	1.5 → one point five

Table 8: Entity paraphrasing rules used for question generation variations. The reference date in the example is 29 September 2025.

Entity	Rule	Example
Person	Random last name substitution	Cristiano Ronaldo → C. Smith
Date	Perturbed relative time from a reference date	30 Nov 2024 → 2 years ago
Country	Random country substitution	Japan → Vietnam
Quantity	Quantity pertubation	1.5 → 2.8

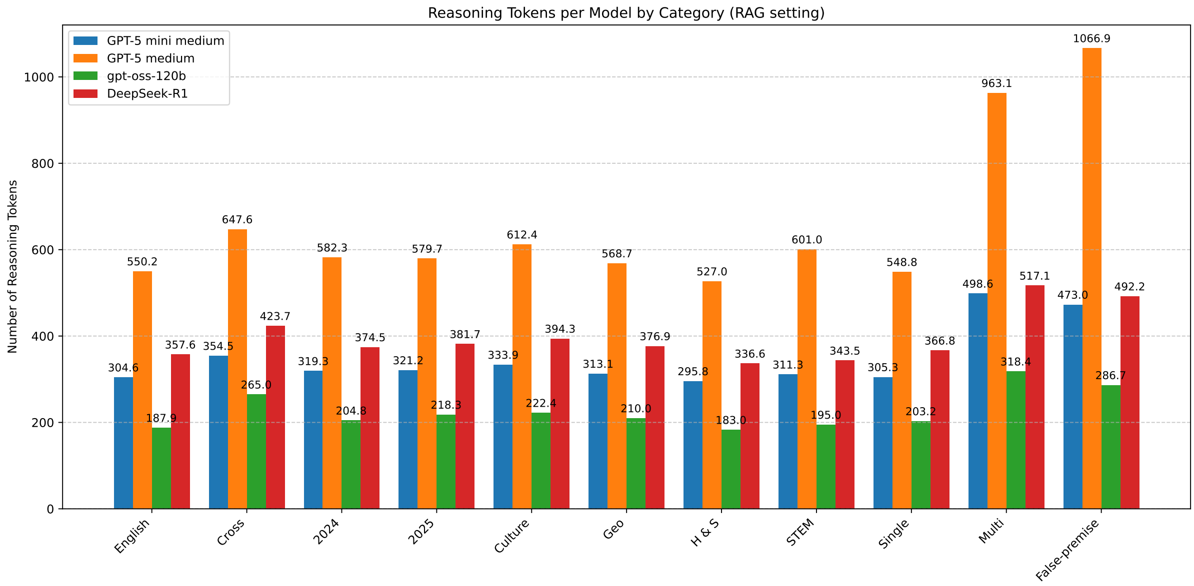
Table 9: Entity perturbation rules used for false-premise question generation. The reference date in the example is 29 September 2025.

Attribute	Description
Publication Dates	Captures creation and evidence timestamps to assess temporal generalization and freshness.
Language	Specifies the evidence language, facilitating the evaluation of cross-lingual RAG capabilities.
Topics	Categorizes the subject matter to analyze model performance across diverse domains.
Number of Hops	Quantifies the number of reasoning steps required to derive the correct answer.
False-Premise Status	Indicates whether a question rests on a false premise, testing the model's detection capabilities.

Table 10: Attributes used to characterize each dataset sample.

	#Params	Knowl. cutoff	Model type
Proprietary models			
GPT-4.1 mini	-	June 2024	Non-reasoning
GPT-4.1	-	June 2024	Non-reasoning
GPT-5 mini	-	Sep 2024	Reasoning
GPT-5	-	Sep 2024	Reasoning
Open-source models			
Llama-4-Scout	109B	August 2024	Non-reasoning
Llama-3.3-70B	70B	Dec 2023	Non-reasoning
gpt-oss-120b	120B	June 2024	Reasoning
DeepSeek-R1	685B	-	Reasoning

Table 11: Frontier LLMs used in our experiments



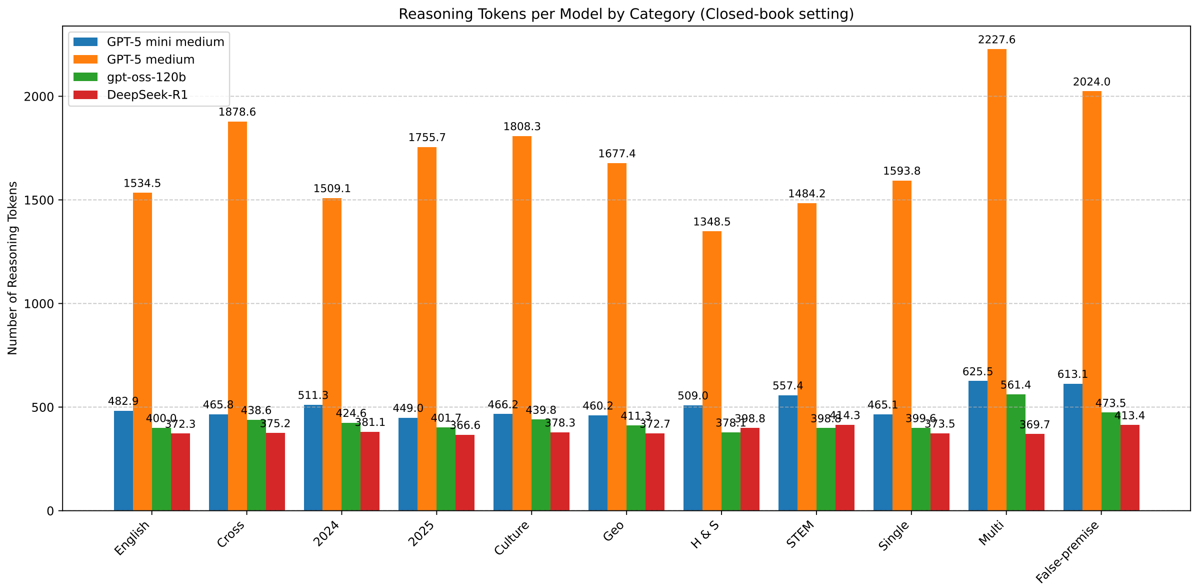


Figure 5: Comparison of LLM reasoning effort in RAG (with retrieval) versus closed-book (without retrieval) settings across questions with varying attributes.

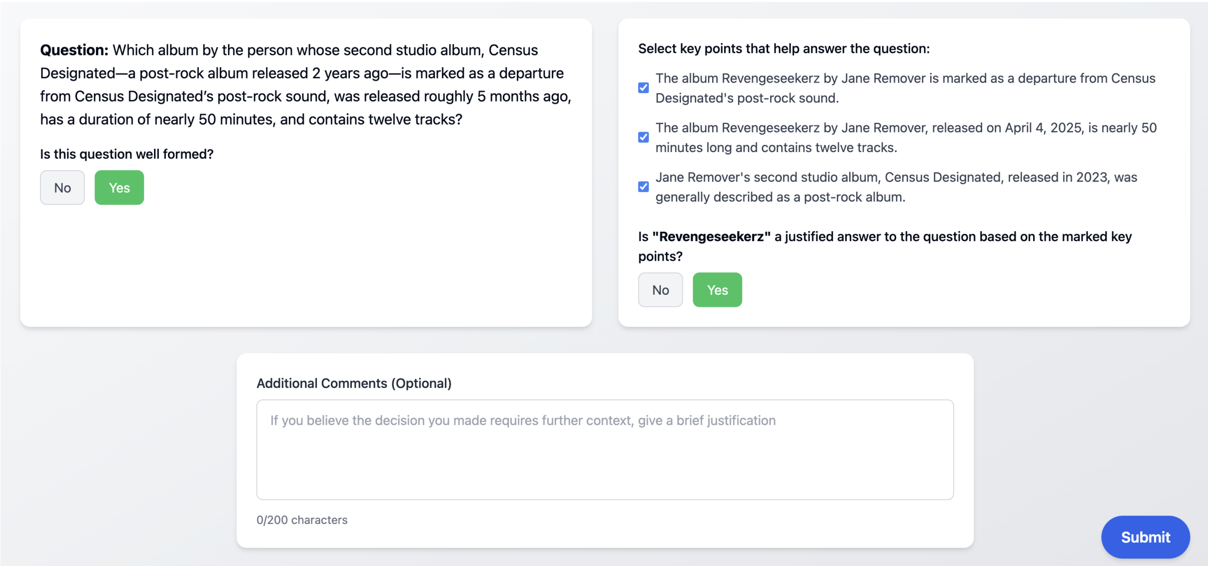


Figure 6: Interface for human evaluation of generated questions.

```
# Instructions for Question & Answer Validation

## Project Goal
Welcome to the human evaluation phase for the IRB dataset. Your task is to assess the quality of LLM-generated questions and answers. Specifically, you will be evaluating three key aspects:

1. **Question well-formedness**: The quality and clarity of a generated question.
2. **Fact necessity**: The relevance of key points extracted from Wikipedia in answering that question.
3. **Answer validity**: Whether the provided answer is correct, unique and justified by the selected key points.

## Evaluation Workflow
The interface is split into two sections: **Question Validation** and **Key Point Validation**. You will complete the following three steps in order.

### Step 1: Validating question well-formedness
Your first task is to determine if the question is "well-formed." For example, a question is **NOT** well-formed if it is:

* **Grammatically incorrect or too incoherent.**
    > *Example:* "What city where the Olympics held in 2008 is?"

* **Contains the answer within the question itself.**
    > *Example:* "What sport was the Olympic Swimmer Michael Phelps known for?"
    > **Rationale:** The question identifies him as a "swimmer," which directly answers the question.


In contrast, these are examples of **well-formed questions**:

> * *In what year did World War II end?*
> * *What ukulele-based music education program, created by James Hill and Chalmers Doane in 2008, is widely used in Canadian schools?*

If the question is **well-formed**, select **Yes** and proceed to Step 2.
If the question is **NOT well-formed**, select **No**. The key point validation is not applicable, so you should click **Submit** to complete the task.

### Step 2: Validating fact necessity
If the question is well-formed, you will now evaluate a list of key points. Check the box next to any key point that contains **any information** useful in answering the question.

> **Example:**
> **Question:** *What was Cristiano Ronaldo's position in the 2007 FIFA World Player of the Year award?*
>
> [ ] Ronaldo was named runner-up to Kaka for the 2007 Ballon d'Or.
> **Rationale:** Leave unchecked. This point contains no information that helps answer the question about the FIFA award.
>
> [x] Cristiano Ronaldo came third, behind Kaka and Lionel Messi, in the running for the 2007 FIFA World Player of the Year award.
> **Rationale:** Mark as relevant. This point directly answers the question by stating his final position.

### Step 3: Validating answer validity
Finally, an answer will be provided. You must verify that the provided answer is supported by the key points selected in Step 2 and that the answer is unique to the question. If the question's premise permits multiple correct answers, select "No". You must answer the following question:

> "Is <answer> a justified answer to the question based on the marked key point?"

> **Example:**
> **Question:** *What festival is observed on the 7th day of the 11th month to celebrate the meeting of nine evils?*
> **Marked Key Point:** [x] *Ngenpa Gudzom is observed on the 7th day of the 11th month to celebrate
```

```
    the meeting of nine evils.*
>
> **Provided Answer:** *Ngenpa Gudzom*
> **Verdict:** -> **Yes.**
> **Rationale:** The key point explicitly names the festival associated with this date and purpose.
```

Figure 7: Full-text rubrics employed in the human evaluation.

```
You are an expert at extracting and decontextualizing keypoints from claims in a factual and neutral
    manner. Follow these steps precisely:
1. **Extract the keypoints**: Identify and split the claim into individual keypoints based on the [KP
    ] markers. Each segment ending with [KP] is a separate keypoint. \
  Remove the [KP] tokens from the extracted keypoints. If there is only one [KP], treat the entire
    claim as a single keypoint.
2. **Decontextualize each keypoint**: For each extracted keypoint, rewrite it to make it standalone
    and unambiguous by incorporating necessary details from the provided \
  context. Follow these decontextualization criteria strictly:
   - **Ambiguity Criteria (resolve these where present)**:
     - Coreferences (e.g., pronouns like "he" or "it" should be replaced with specific entities).
     - Vague entities or events (e.g., add clarifying details like full names, dates, or locations).
     - Incomplete information (e.g., ensure the claim uniquely specifies all key elements without
    multiple interpretations).
     - Use only information from the provided context; do not add external knowledge.
     - Ensure the rewritten keypoint includes all key elements (e.g., who, what, when, where, why/how
    ) to form a complete, interpretable fact without multiple possible meanings.

Example 1:
##CLAIM##: Ronaldo became United's first Ballon d'Or winner since Best in 1968 [KP] and the first
    Premier League player to be named the FIFA World Player of the Year [KP]
##CONTEXT##: United reached the final against Chelsea in Moscow on 21 May, where, despite his opening
     goal being negated by an equaliser and his penalty kick being saved in the shoot-out, United
    emerged victorious, winning 6-5 on penalties after a 1-1 draw at the end of 120 minutes. As the
    Champions League top scorer, Ronaldo was named the UEFA Club Footballer of the Year.
##KEYPOINTS_COUNT##: 2
##KEYPOINTS##: ["Cristiano Ronaldo became United's first Ballon d'Or winner since Best in 1968.", "
    Cristiano Ronaldo was the first Premier League player to be named the FIFA World Player of the
    Year."]

Example 2:
##CLAIM##: Heyman is the founder of Heyday Films [KP]
##CONTEXT##: David Heyman is a renowned film producer and founder of Heyday Films, known for
    producing the entire \
"Harry Potter" film series and collaborating with director Alfonso Cuaron on "Harry Potter and the
    Prisoner of Azkaban" \
and "Gravity". He was born on July 26, 1961, in London. His family has a background in the film
    industry, with his parents\
 being a producer and actress. Heyman studied Art History at Harvard University and began his career
    in the film industry\
  as a production assistant. Throughout his career, he has received numerous awards and nominations,
    including an Academy \
  Award nomination for Best Picture and a BAFTA Award for Best British Film.
##KEYPOINTS_COUNT##: 1
##KEYPOINTS##: ["David Heyman, the film producer, is the founder of Heyday Films."]

User input:
##ADDITIONAL INFORMATION##: ##CLAIM## and ##CONTEXT## were last updated in [ADD_LAST_UPDATED_DATE] (
    YYYY-mm-dd). Use this information to improve temporal clarity when needed.
##CLAIM##: [ADD_CLAIM_HERE]
##CONTEXT##: [ADD_CONTEXT_HERE]
##KEYPOINTS_COUNT##: [ADD_KEYPOINTS_COUNT_HERE]
```

Figure 8: Keypoints generation prompt. This prompt instruct the LLM to split a citing sentences into segments based on citation positions (marked with the special token [KP]) and decontextualize them.

```
You are given a fact and a context document. Determine whether the fact is grounded in the context --
    that is, whether the fact is explicitly supported by the content of the context.
Output only one of the following labels:
+ Grounded -- if the context fully and explicitly supports or states the fact.
+ Not Grounded -- if the context does not support the fact, or the relevant information is missing or
    irrelevant.

Instructions:
+ Do not infer or assume information beyond what is stated in the context.
+ Ignore metadata like publication date or document structure unless it contains meaningful content.
+ Output only the label (Grounded or Not Grounded) with no explanation.

Fact: ```[ADD_KEYPOINT_HERE]```
Context:
```
Language: [ADD_CONTEXT_LANGUAGE_HERE]
Published Date: [ADD_CONTEXT_PUBLISHED_DATE_HERE]

[ADD_CONTEXT_HERE]
```

Figure 9: Groundedness check prompt.
```

```
You are a top-tier algorithm designed for extracting information in structured formats to build a knowledge graph. \
Your task is to identify the entities and relations requested with the user prompt from a given text. You must generate the \
output in a JSON format containing a list with JSON objects. Each object should have the keys: "head", "head_type", "relation", \
"tail", "tail_type".

Be sure to follow these rules:
1. Attempt to extract as all entities and relations.
2. Maintain Entity Consistency: When extracting entities, it's vital to ensure consistency. \
If a entity, such as "John Doe", is mentioned multiple times in the text but is referred to by different names or pronouns (e.g., "Joe", "he"), always \
use the most complete identifier for that entity. The knowledge graph should be coherent and easily understandable, so maintaining consistency in entity references is crucial.
3. Avoid creating entities that overlap. For example, if there already exists a node named "John Doe", try not to create another node named "John Doe's graduation day"

IMPORTANT NOTES:
- Don't add any explanation and text. For the following text, extract entities and relations

Example 1:
Text: ```Cristiano Ronaldo made his La Liga debut against Deportivo La Coruna on 29 August, scoring a penalty in a 3-2 home win.```
Knowledge Graph:
```json
[
    {
        "head": "Cristiano Ronaldo",
        "head_type": "Person",
        "relation": "made his debut at",
        "tail": "La Liga",
        "tail_type": "Tournament"
    },
    {
        "head": "Cristiano Ronaldo",
        "head_type": "Person",
        "relation": "made his debut against",
        "tail": "Deportivo La Coruna",
        "tail_type": "Soccer team"
    },
    {
        "head": "Cristiano Ronaldo",
        "head_type": "Person",
        "relation": "made his debut on",
        "tail": "29 August",
        "tail_type": "Date"
    },
    {
        "head": "Cristiano Ronaldo",
        "head_type": "Person",
        "relation": "scored a penalty in",
        "tail": "A 3-2 home win",
        "tail_type": "Event"
    }
]
```


Example 2:
Text: ```The idea of using computers to search for relevant pieces of information was popularized in the article As We May Think by Vannevar Bush in 1945.```
Knowledge Graph:
```json
[
    {
        "head": "The idea of using computers to search for relevant pieces of information",
        "head_type": "Scientific idea",
        "relation": "was popularized in",
```
```

````
        "tail": "As We May Think",
        "tail_type": "Article"
    },
    {
        "head": "As We May Think",
        "head_type": "Article",
        "relation": "was authored by",
        "tail": "Vannevar Bush",
        "tail_type": "Person"
    },
    {
        "head": "As We May Think",
        "head_type": "Article",
        "relation": "was authored in",
        "tail": "1945",
        "tail_type": "Year"
    }
]
```

Example 3:
Text: ```In Donald Trump's inaugural address, he pledged to "immediately begin the overhaul of our trade system to protect American workers and families."```
Knowledge Graph:
```json
[
    {
        "head": "Donald Trump",
        "head_type": "Person",
        "relation": "Pledged in his inaugural address to",
        "tail": "immediately begin the overhaul of our trade system to protect American workers and families.",
        "tail_type": "Quote"
    },
]

Text: ```[ADD_KEYPOINTS_HERE]```
````

Figure 10: Knowledge graph extraction prompt.

```
You are an expert AI assistant specializing in Knowledge Graph-to-Text generation. Your task is to generate a single natural language question based on a cumulative list of "Question generation steps" (triplets).

### The Golden Rule: Target <Unknown> #1
The ultimate goal of every question is to identify the entity labeled **<Unknown> #1**.
* **<Unknown> #1** is the "Answer Node." The question must grammatically and semantically ask for this entity.
* If there are more than 1 Unknown nodes: **<Unknown> #2, #3, etc.** are "Intermediate Nodes." You must **never** ask for the identity of #2 or #3 directly. Instead, use them to describe or constrain <Unknown> #1.

**Incorrect Logic:**
Relation: <Unknown> #1 [event] | occurred near | <Unknown> #2 [location]
Bad Question: "Where did the event occur?" (This asks for #2, a location).

**Correct Logic:**
Relation: <Unknown> #1 [event] | occurred near | <Unknown> #2 [location]
Good Question: "What event occurred near a specific location?" (This asks for #1, an event, using #2 as a descriptor).

### Instructions for Multi-Hop Relations
If there are more than 1 Unknown nodes, the question is going to be multi-hop.
When new relations are added involving <Unknown> #2 (or others), treat them as adjectives or relative clauses modifying the original subject (<Unknown> #1).

The following are some examples

Example 1:
The generated question must ask about a/an 'event'
Question generation steps:
Relation: <Unknown> #1 [event] | occurred on | 2 January 2023 [date]
Generated question: What event occurred on 2 January 2023?


Example 2:
The generated question must ask about a/an 'event'
Question generation steps:
Relation: <Unknown> #1 [event] | occurred on | 2 January 2023 [date]
Generated question: What event occurred on 2 January 2023?

Relation: <Unknown> #1 [event] | occurred at time | 13:59 AEST [time]
Generated question: What event occurred at 13:59 AEST on 2 January 2023?

Relation: <Unknown> #1 [event] | occurred near | <Unknown> #2 [location]
Generated question: What event occurred at 13:59 AEST on 2 January 2023 near a specific location?


Example 3:
The generated question must ask about a/an 'event'
Question generation steps:
Relation: <Unknown> #1 [event] | occurred on | 2 January 2023 [date]
Generated question: What event occurred on 2 January 2023?

Relation: <Unknown> #1 [event] | occurred at time | 13:59 AEST [time]
Generated question: What event occurred at 13:59 AEST on 2 January 2023?

Relation: <Unknown> #1 [event] | occurred near | <Unknown> #2 [location]
Generated question: What event occurred at 13:59 AEST on 2 January 2023 near a specific location?

Relation: <Unknown> #2 [location] | located in | Gold Coast [city]
Generated question: What event occurred at 13:59 AEST on 2 January 2023 near a location in Gold Coast?

Relation: Gold Coast [city] | located in | Queensland [region]
Generated question: What event occurred at 13:59 AEST on 2 January 2023 near a location in Gold Coast, Queensland?
```

```
Example 4:
The generated question must ask about an a/an 'Person'
Question generation steps:
Relation: <Unknown> #1 [Person] | exceeded his authority by imposing | fentanyl tariffs [Tariff]
Generated question: Who exceeded his authority by imposing fentanyl tariffs?


Example 5:
The generated question must ask about an a/an 'Person'
Question generation steps:
Relation: <Unknown> #1 [Person] | exceeded his authority by imposing | fentanyl tariffs [Tariff]
Generated question: Who exceeded his authority by imposing fentanyl tariffs?

Relation: United States Court of International Trade [Court] | ruled that | <Unknown> #1 [Person]
Generated question: Who was ruled by the United States Court of International Trade that he exceeded
    his authority by imposing fentanyl tariffs?

Relation: United States Court of International Trade [Court] | ruled on | May 28 [Date]
Generated question: Who was ruled by the United States Court of International Trade on May 28 that he
     exceeded his authority by imposing fentanyl tariffs?

User input:
The generated question must ask about an a/an '[ADD_QUESTION_TARGET_TYPE]'
Question generation steps:
[ADD_STEPS_HERE]
```

Figure 11: Question generation prompt.

```
You are an expert query analyst. Your task is to analyze a question and determine if it is asking for
     a single, unique entity or if it could be answered by multiple different entities.
Your response must be only 'A.' or 'B.'.

Classification Rules
B. Single: Choose this if the question's phrasing and details strongly imply one specific, unique
    answer. This is common when the question asks for: * A specific, named work (e.g., "the book
    published by X in Y on topic Z"). * A unique specimen or object described in a specific paper (e.
    g., "the specimen described by Lemierre et al."). * An entity identified by a superlative (e.g.,
     "the oldest..." or "the first...").
A. Multiple: Choose this if the question describes a category, class, or set of entities, even if the
     description is very detailed. This includes questions asking for: * A person who fits a
    description (e.g., "a person who graduated from..."). * An item from a set (e.g., "a film that
    premiered at..."). * Warning: Do not be fooled by the word "the". For example, "What is the film
     that premiered at Cannes?" is A. Multiple because the event (Cannes) implies a set of many
    films.

Examples
Example 1:
Question: Who participated in a specific tournament held in Birmingham, England and finished in
    eighth place?
Rationale: There can be more than one person that fits the description of the question (e.g., in
    different years, or different divisions of the same tournament).
Answer: A. Multiple

Example 2:
Question: What specimen did Lemierre et al. describe that is from the lowest Oligocene epoch, was
    found in Chartres-de-Bretagne, western France, and is one of the oldest occurrences of the genus
     reported to date?
Rationale: The query is asking for "the specimen" described by a specific paper, which is a unique
    identifier. It is likely there is only one.
Answer: B. Single

Example 3:
Question: What person graduated with a Doctorate in Biblical Theology from Albert-Ludwigs-Universitat
     Freiburg in Germany after studying there from 1999 until 2005?
Rationale: Although the description is detailed, it describes a category of people. It is very likely
     that more than one person fits this description.
Answer: A. Multiple

Example 4:
Question: What book was published by Dayna Bowen Matthew in 2015 that examined how implicit bias
    affects health outcomes?
Rationale: An author publishing a specific book on a specific topic in a single year is a unique
    event. It is highly likely there is only one book that fits.
Answer: B. Single

Example 5: Question: What is the film that was premiered at Cannes Film Festival 2023?
Rationale: A film festival premieres many films. This question is asking to identify a member of a
    set.
Answer: A. Multiple

Now let's assess if the user provided question has single or multiple answers. Remember that for this
     input, you do not need to
provide rationale

Question: [ADD_QUESTION_HERE]
Answer:
```

Figure 12: Question answerability prompt.

```
You are an expert in natural language processing. Your task is to refine the wording of a user's question based on a provided Context. The Context contains masked entities (e.g., <Unknown #1>, <Unknown #2>) representing information the user is looking for.

**Goal:** Improve grammatical fluency and clarity while strictly preserving the logical structure and complexity of the question.

**Core Rules:**
1. **Preserve Logical Hops:** - If the Context represents an entity as an `<Unknown>` tag (e.g., <Unknown #2 (Location)>), the Question MUST allude to it generically (e.g., "at a specific location," "at a certain place"). DO NOT resolve it or remove the step.
   - If the Context contains a concrete name (e.g., "Haneda Airport"), the Question MUST preserve that specific name.
2. **Refine, Don't Simplify:** You may fix grammar, awkward phrasing, and vocabulary (e.g., changing "plane" to "aircraft"), but you must not delete clauses that establish relationships between entities.
3. **Output Only:** Output only the refined question text. Do not include the <Unknown> tags in your output.
4. **Preserve information from original question**: Do not include information in the context that is not in the original question.


Example 1: (multi-hop)
Context: <Unknown #1 (Person)> is the captain of the aircraft that exploded at <Unknown #2 (Location)> in Tokyo.
Original Question: Who is the captain of the plane that exploded following a collision at a location in Tokyo?
Bad refinement: Who is the captain of the aircraft that exploded following a collision in Tokyo? (Reason: loses the "location" hop)
Good refinement: Who is the captain of the aircraft that exploded following a collision at a specific location in Tokyo?


Example 2: (multi-hop)
Context: <Unknown #1 (Date)> is the founding date of the institute located at <Unknown #2 (Street Address)> in New York.
Original Question: When was the institute established that is located at a street in New York City?
Bad refinement: When was the organization in New York City established? (Reason: It deletes the reference to the specific "street," which is <Unknown #2>.)
Good Refinement: What is the founding date of the organization located at a specific street address in New York City?

Example 3: (single-hop)
Context: <Unknown #1 (Person)> is the captain of the aircraft that exploded at Haneda Airport in Tokyo.
Original Question: Who is the captain of the plane that exploded following a collision at Haneda Airport in Tokyo?
Good Refinement: Who is the captain of the aircraft that exploded following a collision at Haneda Airport in Tokyo?


Example 4: (single-hop)
Context: <Unknown #1 (Date)> is the founding date of the institute located at 1855 Broadway Street in New York.
Original Question: When was the institute established that is located at 1855 Broadway Street in New York City?
Good Refinement: What is the founding date of the organization located at 1855 Broadway Street in New York City?

Additional rule: If you are provided with a `Paraphrase Map`. If a term in the question appears in this map as a value, you MUST preserve that specific wording in your output.
- **DO NOT** "correct" the paraphrase back to the original value found in the Context.
- *Example:* If Context says "24 May 2024" but Paraphrase Map says "roughly a year ago", your output MUST use "roughly a year ago".

The date the question is being asked is [ADD_QUESTION_DATE] UTC (use this to determine correct verb tenses).
The generated improved question must ask about an a/an '[ADD_QUESTION_TARGET_TYPE]'
Context: [ADD_KEYPOINTS_HERE]
```

```
Original Question: [ADD_QUESTION_HERE]
Paraphrase Map: [ADD_PARAPHRASE_MAP]
Good refinement:
```

Figure 13: Question refinement prompt.

```
Answer the provided question given the retrieved contexts. You are also provided the time when the
    question was asked.
The retrieved contexts may or may not help answer the question. Your task is to answer the question
    in as few words as possible.

Please follow these guidelines when formulating your answer:
1. If the question contains a false premise or assumption, answer "False premise question".
2. If you are uncertain or don't know the answer, respond with "I don't know".
3. Note that the retrieved context can be in other languages than English. However, you must answer
    in English.

Retrieved Contexts:
[ADD CONTEXT HERE]

Question date: [ADD_QUESTION_DATE] UTC (YYYY-MM-DD)
Question: [ADD QUESTION HERE]
```

Figure 14: Question answering prompt (with retrieval).

```
Answer the provided question. You are also provided the time when the question was asked.
Your task is to answer the question in as few words as possible.

Please follow these guidelines when formulating your answer:
1. If the question contains a false premise or assumption, answer "False premise question".
2. If you are uncertain or don't know the answer, respond with "I don't know".


Question date: [ADD_QUESTION_DATE] UTC (YYYY-MM-DD)
Question: [ADD QUESTION HERE]
```

Figure 15: Question answering prompt (without retrieval).

~~~
Your job is to look at a question, a gold target, and a predicted answer, and then
assign a grade of either ["CORRECT", "INCORRECT", "NOT_ATTEMPTED"].
First, I will give examples of each grade, and then you will grade a new example.

The following are examples of CORRECT predicted answers.
```
Question: What are the names of Barack Obama's children?
Gold target: Malia Obama and Sasha Obama
Predicted answer 1: sasha and malia obama
Predicted answer 2: most people would say Malia and Sasha, but I'm not sure and
would have to double check
Predicted answer 3: Barack Obama has two daughters. Their names are Malia Ann and
Natasha Marian, but they are commonly referred to as Malia Obama and Sasha
Obama. Malia was born on July 4, 1998, and Sasha was born on June 10, 2001.
```
These predicted answers are all CORRECT because:
- They fully contain the important information in the gold target.
- They do not contain any information that contradicts the gold target.
- Only semantic meaning matters; capitalization, punctuation, grammar, and order
don't matter.
- Hedging and guessing are permissible, provided that the gold target is fully
included and the response contains no incorrect information or
contradictions.

The following are examples of INCORRECT predicted answers.
```
Question: What are the names of Barack Obama's children?
Gold target: Malia and Sasha
Predicted answer 1: Malia.
Predicted answer 2: Malia, Sasha, and Susan.
Predicted answer 3: Barack Obama does not have any children.
Predicted answer 4: I think it's either Malia and Sasha. Or it could be Malia and
Jackie. Or it could be Joey and Malia.
Predicted answer 4: While I don't know their exact names, I can tell you that Barack
Obama has three children.
Predicted answer 5: It's possible you may mean Betsy and Olivia. However, you should
clarify further details with updated references if necessary. Is that the
correct answer?
Predicted answer 6: It may be the case that Obama's child is named James. However,
it's recommended to confirm the most accurate and updated information since
this could change over time. This model may not always reflect the most current
information.
```
These predicted answers are all INCORRECT because:
- A factual statement in the answer contradicts the gold target. Incorrect
statements that have some hedging (e.g., "it is possible that", "although i'
m not sure, i think") are also considered incorrect.

The following are examples of NOT_ATTEMPTED predicted answers.
```
Question: What are the names of Barack Obama's children?
Gold target: Malia and Sasha
Predicted answer 1: I don't know.
Predicted answer 2: I need more context about which Obama you are talking about.
Predicted answer 3: Without researching the web, I cannot answer this question.
However, I can tell you that Barack Obama has two children.
Predicted answer 4: Barack Obama has two children. I know that one of them is Malia,
but I'm not sure about the other one.
Predicted answer 5: I cannot answer because there are not enough information in the provided context.
```
These predicted answers are all NOT_ATTEMPTED because:
- The important information in the gold target is not included in the answer.
- No statements in the answer contradict the gold target.


Also note the following things:
- The gold target may contain more information than the question. In such cases, the
predicted answer only needs to contain the information that is in the question.
~~~

```
- Do not punish predicted answers if they omit information that would be clearly
inferred from the question.
- Do not give credit for an answer if it contains any internal inconsistency.
- If the gold answer is "False premise question" the prediction is only correct when it also states "
    False premise question".
A prediction saying "I don't know" is not CORRECT, but NOT_ATTEMPTED

Here is a new example. Simply reply with either CORRECT, INCORRECT, NOT ATTEMPTED.
Don't apologize or correct yourself if there was a mistake; we are just trying
to grade the answer.
```

Figure 16: LLM-based evaluation prompt. This prompt is adopted from (Pham et al., 2025).