CondAmbigQA: A Benchmark and Dataset for Conditional Ambiguous Question Answering

Anonymous ACL submission

Abstract

Users often assume that large language models (LLMs) share their cognitive alignment of context and intent, leading them to omit critical information in question-answering (QA) and produce ambiguous queries. Responses based on misaligned assumptions may be perceived as hallucinations. Therefore, identifying possible implicit assumptions is crucial in QA. We propose Conditional Ambiguous Question-Answering (CondAmbigQA), a benchmark comprising 200 ambiguous queries and condition-aware evaluation metrics[1]. Our study pioneers the concept of "conditions" in ambiguous QA tasks through retrieval-based annotation, where conditions represent contextual constraints or assumptions that resolve ambiguities. The retrieval-based strategy uses retrieved Wikipedia fragments to identify possible interpretations for a given query as its conditions and annotate the answers accordingly. Experiments show that models considering conditions before answering improve answer accuracy by 19%, with an additional 5% gain when conditions are explicitly provided. These results underscore the value of conditional reasoning in QA, offering researchers tools for rigorous ambiguity resolution evaluation.

1 Introduction

Large language models (LLMs) have made remarkable progress in question answering (QA). However, these advanced models remain prone to generating unreliable responses, especially in ambiguous contexts, with hallucinations being a primary concern (Ji et al., 2023). This issue stems from a fundamental misalignment between user expectations and model capabilities, where LLMs often misinterpret queries due to limited ability to infer human-like context through common-sense reasoning (Banerjee et al., 2024). Ambiguity in QA is particularly problematic because human communication relies highly on shared background knowledge and implicit cognitive frameworks, often omitting mutual contexts that are not universally recognized outside specific environments. In addition, language itself is inherently ambiguous, as people prefer concise expressions over exhaustive ones (Wasow et al., 2005). As a result, users typically approach QA systems with implicit assumptions, which shape their intent but are not explicitly conveyed in their queries. Since models lack direct access to these assumptions, responses may be logically sound with the query's literal wording yet misaligned with user expectations. To bridge this gap, we approximate these assumptions by leveraging retrieval to surface possible interpretations, which are formalized as explicit conditions.

We argue that identifying and addressing these implicit assumptions is key to disambiguation, ensuring that generated responses are accurate and aligned with user expectations. Current research focuses on improving model reasoning, expanding context length, and enhancing retrieval and the use of relevant information (Ding et al., 2024; Sun et al., 2024; Petroni et al., 2024). Techniques such as Chain-of-Thought (CoT) prompting, reinforcement learning (RL) (Wei et al., 2022; Ahmadian et al., 2024), and human preference alignment (Ji et al., 2024) enhance model capabilities, yet they do not explicitly resolve ambiguity.

This paper introduces **Cond**itional **Ambig**uous **Q**uestion-**A**nswering (CondAmbigQA), a novel framework that tackles ambiguity by incorporating explicit conditions. To approximate the implicit assumptions underlying ambiguous queries, we use a retrieval-based strategy to surface diverse contextual constraints from external knowledge sources (e.g., Wikipedia). These constraints, defined as "conditions," represent contextual prerequisites that clarify plausible interpretations and pinpoint the correct answer. Unlike existing datasets that at-

[1]The CondAmbigQA dataset and evaluation codes are provided in Data and Software sections of the submission.

tempt to enumerate all possible answers based on human knowledge, our framework focuses on identifying key conditions that distinguish a question from similar ones. We design a human-LLM interactive annotation process where GPT-4o assists in refining condition-answer pairs, significantly reducing annotation cost and minimizing subjectivity.

Using CondAmbigQA, we develop an experimental protocol to evaluate models on both condition identification and conditional answer generation. Our results demonstrate that incorporating explicit conditions into answer generation improves response quality compared to standard retrieval-augmented generation (RAG) methods (Lewis et al., 2020). Larger models, such as GPT-4o and GLM4-Plus, outperform smaller models in both condition adherence and answer quality. Additionally, we introduce a metric for citation generation, further enhancing answer reliability. Our main contributions are as follows:

- We are the first to identify implicit conditions as the root cause of ambiguity in QA tasks and propose a framework for disambiguation through explicit condition representation.
- We introduce CondAmbigQA, a novel condition-based framework that structures QA responses around identified conditions, ensuring clarity and relevance in context-specific answers.
- We adopt a human-LLM interactive annotation process that uses GPT-4o to assist in generating condition-answer pairs, significantly reducing annotation costs and maintaining high-quality data.
- Our experiments highlight the importance of condition in QA. When models consider possible conditions during reasoning, they achieve substantial improvements in answer generation accuracy.

2 Related Work

Recent advances in LLMs for QA have primarily focused on alignment through CoT prompting, process supervision, and RL (Brown et al., 2020; Bai et al., 2022). However, these alignment strategies often embed human-biased reward signals, prioritizing expected outcomes over probabilistic reasoning (Hewitt et al., 2024). Agent-based approaches (Zhu et al., 2023) generally lack specialized mechanisms for disambiguation (Park et al., 2023). RAG-based methods have shown promise in improving factual accuracy through retrieval (Lewis et al., 2020; Gao et al., 2023b), but they do not directly address ambiguity arising from implicit assumptions. Self-RAG (Asai et al., 2024) enhances answer reliability through self-reflection, and CRAG (Yan et al., 2024) employs trained evaluators for retrieved documents, but neither explicitly models conditions to resolve multiple interpretations. CoT techniques like retrieval-augmented thought (Wang et al., 2024b) support extended reasoning capabilities. However, these methods often assume the correctness of reasoning steps, potentially amplifying misalignments with user intent (Es et al., 2024).

The evaluation of LLM responses presents unique challenges, as traditional metrics, such as ROUGE and BLEU, fail to capture the complexity of modern outputs. While some research proposes using LLMs as evaluators (Zheng et al., 2023; Yu et al., 2024), this approach risks introducing additional hallucinations (Liu et al., 2023). New frameworks such as G-Eval (Wei et al., 2022), self-evolving benchmarks (Wang et al., 2024a), LiveBench (White et al., 2024), and MixEval (Ni et al., 2024) have emerged to address these limitations. However, establishing unbiased, domain-specific evaluation metrics remains challenging (Gehrmann et al., 2021; Magesh et al., 2024).

In ambiguous QA, existing research has made important advances but faces critical limitations. AmbigQA (Min et al., 2020) rewrites ambiguous questions to capture possible answers; however, its reliance on human annotators introduces bias and fails to codify the implicit conditions driving various interpretations. ASQA (Stelmakh et al., 2022) extends this work by generating long-form answers to cover multiple interpretations, but its annotation process can lead to logical inconsistencies, particularly when linking different answer components. While ALCE (Gao et al., 2023a) enhances credibility through Wikipedia citations, it fails to address the fundamental challenge of implicit ambiguity within queries. Recent approaches like APA (Kim et al., 2024) attempt to detect ambiguity by using an agent to prompt users for clarification, but this method's heavy dependence on model internal biases may inadvertently guide users toward biased or unintended choices. BeaverTails (Ji et al., 2024) leverages human preference, but this approach can

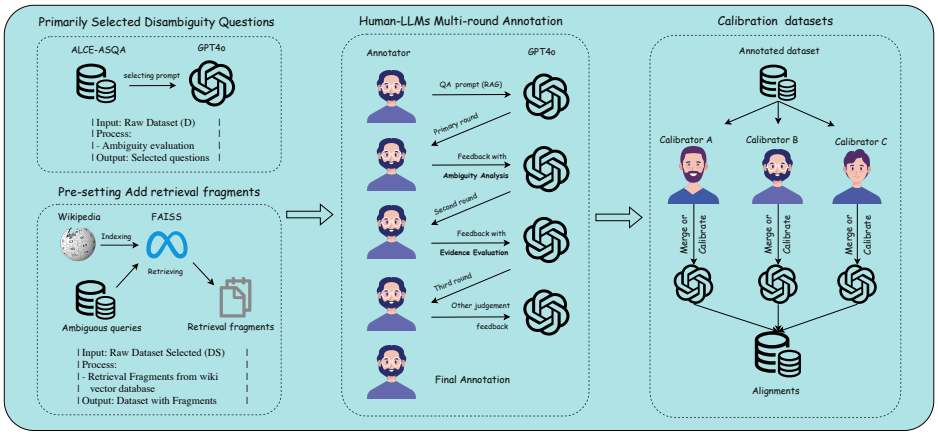

Figure 1: Annotation workflow adopted in CondAmbigQA dataset construction.

amplify annotation biases and may become outdated as underlying knowledge sources evolve.

Unlike prior works that either rewrites queries (AmbigQA, ASQA) or detects ambiguity *post hoc* (APA), our method systematically identifies implicit assumptions by structuring responses around explicit conditions. This approach ensures that retrieved contexts serve as an interpretative guide in reasoning. Additionally, our condition-aware evaluation provides a more precise metric for measuring ambiguity resolution effectiveness.

3 Dataset Construction and Overview

This section presents our dataset and its construction process. We first define the concept of "condition" in LLMs and then provide a comprehensive overview of the dataset.

3.1 Definition of "Condition"

We formally define a **condition** as *a set of contextual constraints that must be satisfied for an answer to be considered correct within a particular scope*. Conditions naturally emerge in RAG systems when retrieved documents provide different but valid grounds for an answer. The need for conditions arises when users pose questions that yield multiple valid answers (Qian et al., 2024), necessitating clarification. For example, the question "when did US currency leave the gold standard?" yields multiple valid answers due to the progressive transition in monetary policy. Some may cite the 1933 suspension during the Great Depression, others the 1968 repeal of gold reserve requirements, and still others the 1971 Nixon Shock. The conditions clarify why multiple answers exist by explicitly identifying the underlying constraints, allowing users to understand the entire historical progression rather than focusing on a single date.

3.2 Dataset Composition and Structure

Our dataset, CondAmbigQA, consists of 200 annotated instances derived from the ALCE-ASQA[2] dataset (Gao et al., 2023a), which originates from AmbigNQ[3] (Min et al., 2020). Each instance contains a user query, retrieved document fragments from Wikipedia[4], and a structured set of condition-answer-citation triples. The components are formally organized as:

$$\begin{aligned}&\texttt{Query}|\{\texttt{RetrievalDocs}\}:\\&\quad\{(\texttt{Condition}_1,\texttt{Answer}_1,\{\texttt{Citation}_1^1,\dots\}),\\&\quad(\texttt{Condition}_2,\texttt{Answer}_2,\{\texttt{Citation}_2^1,\dots\}),\\&\quad\dots\}.\end{aligned}$$

This structure represents a significant advancement over traditional datasets by incorporating retrieved documents and explicit conditions, enabling a more fine-grained evaluation of ambiguity resolution.

3.3 Annotation Process and Guidelines

Figure 1 depicts our annotation workflow, which integrates human expertise with LLM capabilities to construct a robust dataset. Identifying conditions from retrieval results and consistently summarizing key contextual factors is a highly tedious task for human annotators, making the annotation inherently complex and labor-intensive. To address this challenge, we leverage LLMs' superior text comprehension abilities to streamline annotation

[2] https://huggingface.co/datasets/princeton-nlp/ALCE-data
[3] https://huggingface.co/datasets/sewon/ambig_qa
[4] https://huggingface.co/datasets/wikimedia/wikipediahttps://huggingface.co/datasets/sewon/ambig_qa

Dataset	Retrieval Included	Complete Answer	Advanced Reasoning	Ambiguity Resolution
CondAmbigQA	✓	✓	✓	✓
ASQA (Stelmakh et al., 2022)	✗	✓	✓	✓
AmbigNQ (Min et al., 2020)	✗	✗	✗	✓
ALCE (Gao et al., 2023a)	✓	✓	✗	✗
Multihop-RAG (Tang and Yang, 2024)	✓	✗	✓	✗
NaturalQuestions (Kwiatkowski et al., 2019)	✓	✗	✗	✗
TriviaQA (Joshi et al., 2017)	✗	✗	✗	✗
ELI5 (Fan et al., 2019)	✓	✓	✓	✗
TruthfulQA (Lin et al., 2022)	✗	✓	✓	✗

Table 1: Comparison of CondAmbigQA with other datasets.

while maintaining human oversight. LLMs can efficiently process retrieved contexts and generate initial condition summaries in a consistent manner, significantly reducing the cognitive load on human annotators and minimizing subjectivity. However, careful human validation is still needed, particularly when distinguishing subtle variations leading to different answers (Geva et al., 2019).

The annotation team comprises four PhD candidates and two senior researchers specializing in NLP. The first phase involves an initial screening to identify genuinely ambiguous questions. By analyzing both the questions and their corresponding long-form answers from ALCE-ASQA (detailed in Appendix B), GPT-4o filters out cases where ambiguity does not lead to meaningfully different answers. This ensures that human annotators focus on cases where ambiguity is truly impactful.

We adopt a three-round annotation process, where GPT-4o and human annotators iteratively refine the annotations. In the first round, GPT-4o processes each query using predefined dataset-construction prompts to draft initial condition-answer pairs. Annotators then leverage LLMs to analyze these pairs and validate their ambiguity using the corresponding prompts. In the final round, the LLM maps these condition-answer pairs to supporting citations from retrieved passages. Human annotators independently review the LLM-generated responses, focusing on reasoning coherence, logical soundness, and citation accuracy. If additional information or clarification is necessary for more precise tuples, the annotators reject the current output and provide feedback for calibration. If no further refinement is required, the tuples are accepted as final. The complete set of prompts provided to annotators is listed in Appendix C.

To ensure data quality, regular team meetings are held to collectively discuss difficult cases, resolve ambiguities, and maintain consistency across annotations. The final dataset integrates multiple annotators' insights while preserving logical coherence and eliminating redundancy.

Through this three-round process, GPT-4o generates satisfactory condition-answer-citation pairs for 40% of cases without modification. With two additional rounds of expert feedback and calibration, this percentage increased to 85%, indicating that although LLMs can handle a substantial portion of the task, human expertise remains essential for handling more complex cases. The finding also suggests that this is a meaningful and challenging research problem, highlighting the need for further studies in condition-guided ambiguity resolution.

The current dataset size of 200 instances reflects a trade-off between quality and scale. While fully manual annotation takes at least 30 minutes per query, our LLM-assisted approach reduces this time to an average of 10 minutes, significantly improving efficiency. However, annotation remains resource intensive and intellectually demanding due to the need of extensive review and cross-checking. We plan to extend the dataset to 500 instances in the coming months and release it publicly to encourage contributions from broader research community.

3.4 Dataset Features and Advantages

CondAmbigQA provides a framework for assessing ambiguous QA, incorporating key features that enable systematic evaluation, as outlined in Table 1.

First, **retrieval-included** annotations ensure that different models are evaluated under consistent background information when addressing ambiguity. The retrieved fragments not only provide evidence for answers but also serve as sources for extracting conditions. This feature allows for assessing how well models utilize contextual information to ground their reasoning. Second, CondAmbigQA

is designed to ensure **complete answers** by providing explicit condition-answer-citation pairings. Unlike datasets that force a single answer, our structure enables the evaluation of multiple interpretations grounded in conditions, ensuring that answers are both comprehensive and contextually appropriate. Third, the dataset requires **advanced reasoning** by presenting scenarios that demand nuanced condition identification and answer generation. This challenges models to engage in deeper logical reasoning, encouraging them to generate well-grounded responses based on external information. Finally, CondAmbigQA emphasizes **ambiguity resolution**, explicitly capturing possible clarifications for ambiguous questions. This allows for a structured evaluation of how effectively models recognize, interpret, and resolve ambiguity by interpreting distinct possible meanings.

Compared to other datasets like ASQA and AmbigNQ, CondAmbigQA's unique features makes it particularly well-suited for benchmarking models on ambiguous QA.

Data Sources and Licensing

CondAmbigQA is built upon AmbigNQ (Min et al., 2020), which is distributed under the CC BY-SA 3.0 license. Context passages are retrieved from Wikipedia under the same license, allowing for reproduction and distribution with appropriate attribution. To maintain consistency with these data sources, we will release our dataset under the CC BY-SA 4.0 license.

4 Experimental Design

In this section, we describe the experimental protocol for the CondAmbigQA benchmark. Our approach decomposes ambiguous question answering into two sequential stages: (1) disambiguation through explicit condition identification and (2) conditional answer and citation generation.

4.1 Evaluation Metrics

To quantitatively assess model performance at each stage, we employ a multi-metric evaluation framework. Let (M) denote the model output and (G) the corresponding ground truth. We define G-Eval (Liu et al., 2023) to measure the quality of output relative to the reference, following criteria similar to those in (Yao et al., 2024; Liu et al., 2023), as implemented in the `DeepEval` package[5]. Four metrics

[5] https://github.com/confident-ai/deepeval

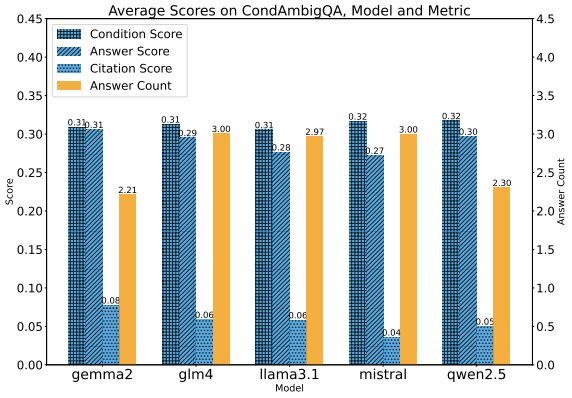

Figure 2: Model performance on four metrics.

are defined, with detailed prompts provided in Appendix D, which describe the instructions used for large language models to generate relevant outputs.

Condition Score quantifies the quality of condition identification by comparing the model's extracted conditions against the ground truth conditions. It assesses both the completeness and clarity of the extracted conditions. The G-Eval framework evaluates whether the model has accurately identified and clearly articulated all relevant conditions from the input.

Answer Score evaluates the factual accuracy and contextual relevance of generated answers by comparing the model's answers against the ground truth answers. The G-Eval framework assesses whether the responses are factually correct and appropriately address the identified conditions.

Citation Score measures source attribution accuracy, which is defined as follows:

$$\textit{Citation Score}(M, G) = \frac{|\{c \in M.\text{citations}\} \cap \{c \in G.\text{citations}\}|}{|\{c \in G.\text{citations}\}|}, \quad (1)$$

where c refers to individual citations. $M.\text{citations}$ represents the citations generated by the model M, and $G.\text{citations}$ represents the ground truth citations. The numerator counts the shared citations between M and G, and the denominator is the total number of citations in G. This score reflects the model's accuracy in attributing citations relative to the ground truth.

Answer Count captures the discrepancy in the number of generated answers.

4.2 Experimental Protocol

The experiment protocol comprises two settings. In the primary setting, each model is provided with a query Q along with the retrieved passages P, and is required to (i) extract disambiguating condi-

tions from P, and (ii) generate answers based on the extracted conditions, supported with citations. The outputs are then evaluated using the aforementioned metrics. This end-to-end evaluation assesses the model's ability in both condition identification and conditional answer generation. Additionally, models are supplied with ground truth conditions alongside Q and P in an alternative setting. By comparing the performance of the model-generated and ground truth conditions, we quantitatively assess the impact of explicit condition guidance on answer generation quality and citation accuracy.

4.3 Baseline Models and Deployment

We evaluate the CondAmbigQA benchmark using five open-source language models with comparable parameter scales, i.e., LLaMA3.1 (8B) (Dubey et al., 2024), Mistral (7B) (Jiang et al., 2023), Gemma (9B) (Team et al., 2024), GLM4 (9B) (GLM et al., 2024), and Qwen2.5 (7B) (Yang et al., 2024). All models are deployed via the `Ollama` framework using default sampling parameters and an 8K context window. The models are prompted according to the instructions described in Appendix D.

Model	Condition Score	Answer Score	Citation Score
Mistral	0.316 ± 0.116	0.272 ± 0.137	0.036 ± 0.116
Qwen2.5	$\mathbf{0.317}$ ± 0.103	0.297 ± 0.159	0.050 ± 0.134
Gemma2	0.309 ± 0.111	$\mathbf{0.306}$ ± 0.135	$\mathbf{0.077}$ ± 0.173
GLM4	0.313 ± 0.110	0.295 ± 0.153	0.059 ± 0.151
LLaMA3.1	0.305 ± 0.103	0.276 ± 0.136	0.058 ± 0.144
Average	0.312	0.289	0.056

Table 2: Main experiment scores.

5 Experimental Results

This section presents the experimental evaluation on the CondAmbigQA benchmark. The results reveal varying performance levels across models.

5.1 Condition Generation Performance

The results summarized in Table 2 show that, for condition scores, the models exhibit similar performance (ranging from 0.305 to 0.317). Qwen2.5 achieved the highest score of 0.317 ($\sigma = 0.103$), with Mistral and GLM4 scoring slightly lower at 0.316 ($\sigma = 0.116$) and 0.313 ($\sigma = 0.110$), respectively. This clustering of scores suggests that, despite architectural differences, current LLMs exhibit comparable capabilities in identifying and proposing potential conditions. However, these scores still leave significant room for improvement, as the ability to identify disambiguating conditions is not yet fully optimized across models.

In condition generation, we observed that models often struggle to fully capture the context. For instance, when asked, "when did US currency leave the gold standard?", Gemma2 generated conditions focusing on "abandonment of the gold standard in the early 20th century" (score $= 0.37$), which only captures the initial phase of the transition without addressing critical later developments. Meanwhile, LLaMA3.1's response emphasized the Great Depression era suspension but failed to articulate the distinction between temporary suspension and final abandonment (score $= 0.48$). These examples demonstrate that while models can identify individual historical events, they share common limitations in capturing the complete progression of policy changes over time, as reflected in their condition evaluation scores rarely exceeding 0.5.

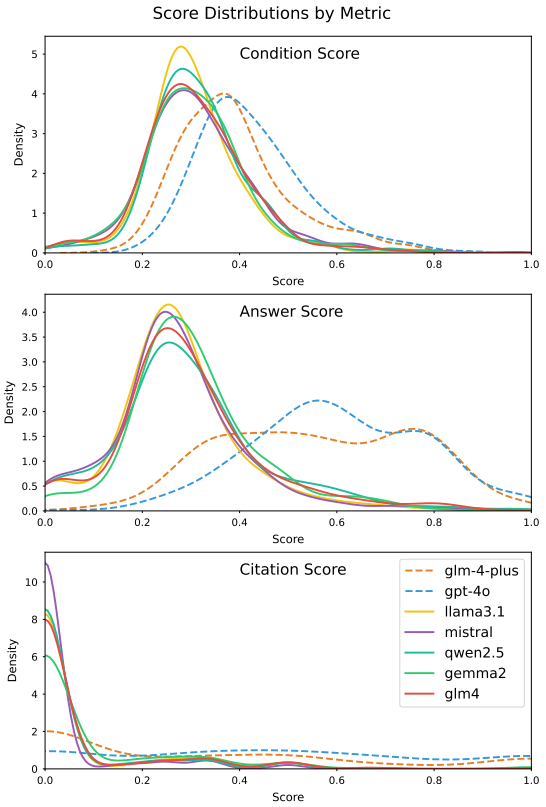

Figure 3: Comparison of score distributions across metrics for models of different scales.

5.2 Answer Generation Performance

The results for answer generation show more significant variability across models. Gemma2 achieved the highest Answer Score of 0.306 ($\sigma = 0.135$), followed by Qwen2.5 with a score of 0.297 ($\sigma =$

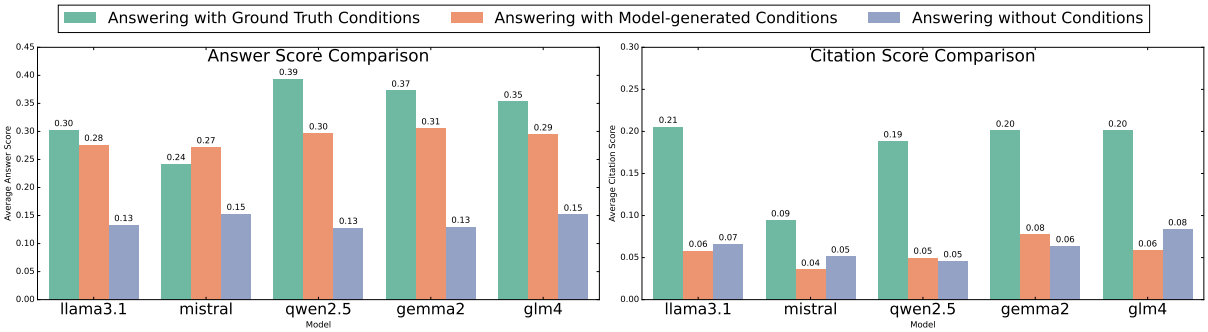

Figure 4: Model performance in Answer Score and Citation Score, comparing answering without conditions, answering based on identified conditions (Main Experiment), and answering based on ground truth conditions.

0.159) and GLM4 at 0.295 ($\sigma = 0.153$), suggesting that both tasks are of comparable difficulty for the current models.

5.3 Citation Generation Performance

The most notable performance gap across models is in citation generation, where even the best-performing model, Gemma2, achieves a relatively low Citation Score of 0.077 ($\sigma = 0.173$) due to excessively large quantity of 0 scores. This result indicates that, despite improvements in condition and answer generation, LLMs struggle with accurately attributing information to sources.

5.4 Scaling Analysis

We conducted scaling experiments with larger models, such as GPT-4o and GLM4-plus, following the same evaluation protocol. Figure 3 presents a performance comparison of models with different scales across key metrics.

Our findings reveal that larger models exhibit enhanced capabilities in handling complex queries, including those involving the identification of multiple conditions. In particular, their performance peaks at a Condition Score of 0.45–0.50, significantly surpassing that of smaller models. Moreover, the Answer Score distributions for these larger models display a distinctive bimodal pattern, with peaks between 0.5 and 0.7, whereas smaller models typically show a single peak around 0.25.

Despite these encouraging trends, improvements in Citation Scores are modest. Larger models achieve citation scores in the range of 0.08–0.09, compared to 0.05–0.07 for smaller models. This indicates that while scaling up model size leads to better handling of complex queries, accurate citation generation is still challenging. Additionally, the relatively small gap (approximately 0.15 in Condition Scores) between the largest and smallest models suggests diminishing returns, highlighting the need for novel strategies to further enhance model performance.

5.5 Extensive Study on the Significance of Conditions

To validate the importance of conditions in RAG and QA systems, we conducted comparative experiments across three approaches: RAG with self-generated conditions (the same as the main experiment), RAG with annotated ground truth conditions, and traditional RAG without considering conditions. As shown in Figure 4, both Answer Score and Citation Score demonstrate consistent hierarchical patterns across all tested models. In general, answering with ground truth conditions achieves a mean of 0.33 for the Answer Score, outperforming self-generated conditions (0.29) and the baseline without conditions (0.14) by margins of 0.04 and 0.19, respectively. The Citation Score shows a more significant improvement, where ground truth conditions (0.18) demonstrate a clear advantage over both self-generated conditions and the unconditioned approach (both at 0.06), representing a 200% improvement in citation accuracy.

These aggregated results strongly validate our hypothesis: the inclusion of condition discovery in ambiguous QA, especially with accurate ground truth conditions, substantially improves both answer quality and citation accuracy. The consistent performance gaps across both metrics underscore the fundamental importance of conditional information in enhancing RAG system performance.

5.6 Case Study Analysis

We conducted case studies to analyze how different models handle specific queries. For example, for the query "where is the TV show *The Ranch* located?" shown in Table 3, we observe that smaller

Ground Truth Conditions	
Condition	**Description**
GT1	The show *The Ranch* is primarily set in a fictional small town called Garrison in Colorado. The show's story revolves around the Bennett family and their Iron River Ranch.
GT2	While set in Colorado, the show was primarily filmed at a sound stage in Burbank, California. The town of Ouray, Colorado appears in the opening sequence.
GT3	The show features both interior shots (filmed in California) and exterior establishing shots (filmed in Colorado).

Model Evaluations			
Scale	**Model**	**Generated Condition**	**Analysis**
Small Models	LLaMA 3.1	Other types of ranches and related concepts remain undeveloped in terms of their broader societal implications.	Completely irrelevant
	LLaMA3.1	Movie ranches and TV series sets in California remain undeveloped.	Incorrect context
	Gemma2	Definition of Ranching	Generic definition
	Gemma2	Production of *The Ranch* (2018 TV Series)	Not location-focused
	GLM4	The term 'ranch' refers to land primarily used for raising grazing livestock and is a subtype of farm.	Generic definition
	GLM4	Sable Ranch in Santa Clarita was a filming location used for various film and television series before being destroyed in a wildfire.	Wrong location
	Qwen2.5	The destruction of Sable Ranch during the Sand Fire wildfire.	Wrong location
	Qwen2.5	The plot and characters of the TV series *The Ranch* (2006)	Not location-focused
Large Models	GPT-4o	Setting of the TV show *The Ranch*	Clear setting focus
	GPT-4o	Filming locations for *The Ranch*	Location specific
	GLM4-plus	Filming Location of *The Ranch*	Direct focus
	GLM4-plus	Setting of *The Ranch* in Colorado	Abstract but accurate

Table 3: Ground truth conditions and model evaluations for the query "Where is the TV show *The Ranch* located?"

models often generate irrelevant or overly generic conditions, such as definitions of "ranching" or "movie sets." In contrast, larger models tend to provide more focused and accurate location-specific conditions. Through these case studies, we identified distinct score patterns that correlate with response quality. In general, scores below 0.20 consistently indicate irrelevant responses, such as LLaMA3.1's condition about "broader societal implications" (score: 0.11). Scores between 0.20 and 0.35 represent partially relevant but imprecise responses, exemplified by Gemma2's generic "Definition of Ranching" (score: 0.24). Moreover, higher scores between 0.35 abd 0.50 indicate accurate but insufficiently detailed responses, while scores above 0.50, achieved by GLM4-plus's "Setting of *The Ranch* in Colorado" (score: 0.53), represent high-quality, focused responses. These thresholds remain consistent across different queries and models, suggesting that they can be reliably used as quality indicators.

In summary, larger models tend to generate more precise and accurate conditions compared to smaller models. However, they still face significant challenges in citation accuracy, which remains a bottleneck across all model sizes.

6 Conclusion and Future Work

This work introduces **CondAmbigQA**, a novel framework and benchmark designed to address ambiguity in QA y explicitly identifying conditions. Our experiments demonstrate that incorporating explicit condition identification enhances both answer quality and interpretability by clarifying the decision-making process. The analysis reveals that while larger models excel in condition processing, even moderate-sized models gain substantial benefits from this guidance. Additionally, our human-LLM collaborative annotation process has helped ensure a high-quality dataset with reduced subjectivity and bias. Overall, CondAmbigQA establishes a new paradigm for enhancing performance and reliability in ambiguous QA scenarios.

Our findings suggest that condition identification could serve as a foundation for enhancing the reasoning capabilities of language models. Future research could integrate condition-based frameworks into the architecture of LLMs to improve their logical reasoning abilities. This could involve the development of specialized reasoning mechanisms that explicitly model condition representations and their logical dependencies.

Limitations

Despite the promising results, several limitations remain:

- **Dataset Constraints:** The dataset currently contains only 200 annotated instances due to resource limitations. While it has been carefully curated, it may not yet fully capture the diversity of ambiguity patterns encountered in real-world scenarios. A key challenge in dataset construction is the high annotation complexity. Unlike simple classification tasks, annotators must review retrieved passages, LLM-generated responses, and corresponding citations, performing extensive cross-checking to ensure that identified conditions are both accurate and well-grounded. This labor-intensive nature of annotation limits the speed of dataset expansion but ensures higher-quality data. However, the interactive annotation process allows us to expand the dataset, with plans to include 500 instances in the coming months. Additionally, the dataset will be publicly available, inviting contributions from the research community.

- **Methodological Challenges:** While our evaluation framework is comprehensive, it may not fully capture subtle variations in model responses. Current evaluation relies on G-Eval, which, despite its effectiveness, might not always align with human judgments in nuanced cases. Further investigation is needed to refine the evaluation metrics and improve their reliability in assessing condition-based QA.

- **Scalability Considerations:** While our experiments indicate improvements in answer quality and interpretability, the scalability of the condition-guided approach in large-scale deployments remains to be thoroughly evaluated. The additional computational overhead associated with processing explicit conditions might limit the practicality of our approach in time-sensitive or resource-constrained applications.

These limitations highlight the need for future refinement of both the framework and the associated methodologies, ensuring that the benefits of condition-based disambiguation can be maintained across a broader spectrum of applications and model architectures.

References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *arXiv preprint arXiv:2402.14740*.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Sourav Banerjee, Ayushi Agarwal, and Saloni Singla. 2024. Llms will always hallucinate, and we need to live with this. *arXiv preprint arXiv:2409.05746*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. 2024. Longrope: Extending llm context window beyond 2 million tokens. *arXiv preprint arXiv:2402.13753*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. RAGAs: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158, St. Julians, Malta. Association for Computational Linguistics.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023a. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023b. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh Dhole, Wanyu Du, Esin Durmus, Ondřej Dušek, Chris Chinenye Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Mihir Kale, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Andre Niyongabo Rubungo, Salomey Osei, Ankur Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 96–120, Online. Association for Computational Linguistics.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China. Association for Computational Linguistics.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, et al. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

John Hewitt, Nelson F Liu, Percy Liang, and Christopher D Manning. 2024. Instruction following without instruction tuning. *arXiv preprint arXiv:2409.14254*.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2024. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36.

Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023. Towards mitigating LLM hallucination via self reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1827–1843, Singapore. Association for Computational Linguistics.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Hyuhng Joon Kim, Youna Kim, Cheonbok Park, Junyeob Kim, Choonghyun Park, Kang Min Yoo, Sanggoo Lee, and Taeuk Kim. 2024. Aligning language models to explicitly handle ambiguity. *arXiv preprint arXiv:2404.11972*.

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. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

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, Dublin, Ireland. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Varun Magesh, Faiz Surani, Matthew Dahl, Mirac Suzgun, Christopher D Manning, and Daniel E Ho.

2024. Hallucination-free? assessing the reliability of leading ai legal research tools. *arXiv preprint arXiv:2405.20362*.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. 2024. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures. *arXiv preprint arXiv:2406.06565*.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22.

Fabio Petroni, Federico Siciliano, Fabrizio Silvestri, and Giovanni Trappolini. 2024. Ir-rag@ sigir24: Information retrieval's role in rag systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3036–3039.

Cheng Qian, Bingxiang He, Zhong Zhuang, Jia Deng, Yujia Qin, Xin Cong, Yankai Lin, Zhong Zhang, Zhiyuan Liu, and Maosong Sun. 2024. Tell me more! towards implicit user intention understanding of language model driven agents. *arXiv preprint arXiv:2402.09205*.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hongda Sun, Weikai Xu, Wei Liu, Jian Luan, Bin Wang, Shuo Shang, Ji-Rong Wen, and Rui Yan. 2024. Determlr: Augmenting llm-based logical reasoning from indeterminacy to determinacy. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9828–9862.

Yixuan Tang and Yi Yang. 2024. Multihop-RAG: Benchmarking retrieval-augmented generation for multi-hop queries. In *First Conference on Language Modeling*.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Siyuan Wang, Zhuohan Long, Zhihao Fan, Zhongyu Wei, and Xuanjing Huang. 2024a. Benchmark self-evolving: A multi-agent framework for dynamic llm evaluation. *arXiv preprint arXiv:2402.11443*.

Zihao Wang, Anji Liu, Haowei Lin, Jiaqi Li, Xiaojian Ma, and Yitao Liang. 2024b. Rat: Retrieval augmented thoughts elicit context-aware reasoning in long-horizon generation. *arXiv preprint arXiv:2403.05313*.

Thomas Wasow, Amy Perfors, and David Beaver. 2005. The puzzle of ambiguity. *Morphology and the web of grammar: Essays in memory of Steven G. Lapointe*, pages 265–282.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Ben Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Siddartha Naidu, et al. 2024. Livebench: A challenging, contamination-free llm benchmark. arxiv. *arXiv preprint arXiv:2406.19314*.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation. *arXiv preprint arXiv:2401.15884*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Jing Yao, Xiaoyuan Yi, and Xing Xie. 2024. Clave: An adaptive framework for evaluating values of llm generated responses. *arXiv preprint arXiv:2407.10725*.

Hao Yu, Aoran Gan, Kai Zhang, Shiwei Tong, Qi Liu, and Zhaofeng Liu. 2024. Evaluation of retrieval-augmented generation: A survey. *arXiv preprint arXiv:2405.07437*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Xinyu Zhu, Junjie Wang, Lin Zhang, Yuxiang Zhang, Yongfeng Huang, Ruyi Gan, Jiaxing Zhang, and Yujiu Yang. 2023. Solving math word problems via cooperative reasoning induced language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4471–4485, Toronto, Canada. Association for Computational Linguistics.

Appendix

A Dataset Examples

Question: When did the show Last Man Standing start?

Condition: "Last Man Standing" is an American sitcom that aired on ABC and Fox. The show originally premiered on ABC in 2011 and was later picked up by Fox in 2018.

Ground Truth: The show first premiered on ABC on October 11, 2011, marking its initial broadcast with a special one-hour episode.

Citations:
Fragment 1: "The show premiered on ABC on October 11, 2011, with a one-hour special episode."

Fragment 2: "The show originally aired on ABC, then switched to Fox, where it continued in 2018."

Fragment 3: "Last Man Standing debuted on ABC on October 11, 2011, airing two episodes in the first hour."

Retrieval Fragments:
Fragment 1: "Last Man Standing debuted on ABC on October 11, 2011, marking its official start."

Fragment 2: "The show's premiere on ABC occurred on October 11, 2011, as a one-hour special."

Fragment 3: "The show, starring Tim Allen, first aired on ABC in 2011 before transitioning to Fox in 2018."

Condition: "Last Man Standing" was canceled by ABC and later re-aired by Fox. The show continued to air after transitioning from ABC to Fox.

Ground Truth: On Fox, the show "started" again on September 28, 2018, marking its re-premiere.

Citations:
Fragment 1: "The show's re-premiere occurred on Fox on September 28, 2018."

Fragment 2: "After being canceled by ABC, Fox picked up the show, with the first new episode airing on September 28, 2018."
Fragment 3: "Fox aired the first season on September 28, 2018, marking the show's new chapter."

Retrieval Fragments:
Fragment 1: "Fox began airing the seventh season on September 28, 2018, after the show's cancellation on ABC."
Fragment 2: "The show's first season on Fox premiered on September 28, 2018, following its ABC cancellation."
Fragment 3: "Last Man Standing, which had been canceled by ABC, returned for its seventh season on Fox on September 28, 2018."

B Query Prompts Template

Query Analysis Instructions Template

You are a professional question analysis assistant. Your task is to analyze questions and their previous incomplete annotations, determining whether these questions contain ambiguities or have multiple possible answers. Please carefully read the following instructions and complete the analysis as required. First, you will receive two inputs: `<questions> {{QUESTIONS}} </questions>`
`<previous_annotations> {{PREVIOUS_ANNOTATIONS}} </previous_annotations>`
Please follow these steps:
1) Read each question and annotation carefully.
2) Analyze each question for:
a) ambiguity - explain different interpretations
b) multiple possible answers - provide examples
3) Consider: question clarity, vague terms, context sufficiency, subjective elements
4) Use format:

```
<analysis>
<question_number>Number</question_number>
<question_text>Text</question_text>
<ambiguity_analysis>Results</ambiguity_analysis>
<multiple_answers>Results</multiple_answers>
</analysis>
```

5) Complete in English
6) Compare with previous annotations

C Dataset Prompts

Dataset Prompts (Part 1)

Question Answering:
You are tasked with providing a structured answer to a question based on the given text fragments. Your goal is to present possible interpretations supported by the fragments, clearly distinguishing between preconditions and detailed answers.
Question: <question> [INSERT QUESTION HERE] </question>
Text fragments:
<fragments>
[INSERT FRAGMENTS HERE]
</fragments>
Answer format:
<answer>
Interpretation [X]:
Preconditions:
* [Necessary background information or assumptions, not directly answering the question] [Fragment X]
* [Necessary background information or assumptions, not directly answering the question] [Fragment Y]
Detailed answer:
* [Specific information directly answering the question] [Fragment Z]
* [Specific information directly answering the question] [Fragment A, Fragment B]
[Repeat the Interpretation structure for as many interpretations as necessary]
</answer>
Ensure all interpretations are distinct, citing relevant fragments for support. If conflicting information is found, present all viewpoints with sources.

Ambiguity Analysis:
Analyze potential ambiguities in the question "[INSERT QUESTION HERE]" based on the provided interpretations. Consider different contexts and how they influence interpretations.
<analysis>
Ambiguity point [X]: [Describe ambiguity that could lead to different interpretations]
Impact:
1. [Impact on Interpretation 1] [Based on Fragment X, Y]
2. [Impact on Interpretation 2] [Based on Fragment Z, A]
Contextual considerations: [How different backgrounds might affect understanding]
[Repeat the Ambiguity point structure for as many ambiguities as necessary]
</analysis>
Explain how each ambiguity leads to different valid answers, citing relevant fragments.

Evidence Evaluation:
For each interpretation of the question "[INSERT QUESTION HERE]", evaluate the supporting evidence. Consider source reliability, consistency across fragments, and potential biases.
<evaluation>
Interpretation [X]: [Brief summary of Interpretation X]
Evidence assessment:
* Strengths: [List strong evidence supporting this interpretation] [Fragment X, Y]
* Weaknesses: [Point out potential issues or shortcomings] [Fragment Z]
* Consistency: [Evaluate the consistency of information across fragments]
Overall credibility: [Provide an overall assessment, e.g., "High", "Medium", or "Low"]
[Repeat the Interpretation structure for as many interpretations as necessary]
</evaluation>
Provide a balanced assessment, citing specific fragments to support your evaluation.

Dataset Prompts (Part 2)

Structured Answer:
Please provide your answer using the following format:

<answer>
Interpretation [X]:
Preconditions:
* [Necessary background information or assumptions, not directly answering the question] [Fragment X]
* [Necessary background information or assumptions, not directly answering the question] [Fragment Y]
Detailed answer:
* [Specific information directly answering the question] [Fragment Z]
* [Specific information directly answering the question] [Fragment A, Fragment B]
[Repeat the Interpretation structure for as many interpretations as necessary]
</answer>
Provide all possible interpretations, ensuring that preconditions and detailed answers are clearly distinct. Every statement must be supported by at least one fragment citation. If you find conflicting information, present all viewpoints and clearly indicate the source of each.

Calibration:
You are tasked with generating a response based strictly on the provided retrieved fragments. Do not introduce any external knowledge or assumptions. Your job is to fill out the following fields using only the information present in the fragments. If any information is missing, leave that field blank.
1. Condition: Summarize the context of the question strictly using the provided fragments. Do not speculate beyond the given information.
2. Ground truth: Provide the exact answer to the question based on the retrieved fragments. Use only what is explicitly stated.
3. Citations: List the relevant fragments that support your answer. Include the title and text of the fragments that were used.
4. Reason: Explain how the answer was derived solely from the fragments, and mention why any gaps in information were left unfilled.
Fragments: retrieved fragments

Output format:
"condition": "<summary based on fragments>", "ground truth": ["<answer derived from fragments>"], "citations": ["title": "<fragment title>", "text": "<fragment text>"], "reason": "<explanation>"

Merging:
You are provided with a question and several annotated dictionaries. Your task is to merge all the dictionaries without changing the structure or key names. Consolidate similar information, eliminate redundancy, and ensure that the final output accurately reflects the content of all dictionaries. Do not introduce external knowledge or assumptions.
Question: question

Dictionaries: dictionaries

Instructions:
- Merge the "condition" fields from all dictionaries into one, keeping only unique and relevant information.
- Merge the "ground truth" fields into a single list, ensuring no redundant entries.
- Combine the "citations" fields from all dictionaries, ensuring all relevant citations are included without duplication.
- Leave the "reason" field as an empty string.

Output format:
"condition": "<merged condition from all dictionaries>", "ground truth": ["<merged ground truth from all dictionaries>"], "citations": ["title": "<citation title from any dictionary>", "text": "<citation text from any dictionary>"], "reason": ""

D Evaluation Prompts

Evaluation Prompts

RAG with Conditions Prompt:
Question: {question}
Retrieved fragments:
{Fragment 1 - {title}: {text}}
...
Please complete the following tasks:
1. Identify up to THREE key conditions related to the question based solely on the provided fragments.
2. For each condition, provide a corresponding detailed answer.
3. Cite the sources (fragment numbers) that support each condition and answer.
4. Output the results in JSON format with the following structure.

Modified Condition-based Prompt:
Question: {question}
Context fragments:
{Fragment 1 - {title}: {text}}
...
Conditions to address:
Condition 1: {condition}
...
IMPORTANT: Respond with ONLY the following JSON format, no other text.

Standard RAG Prompt:
Question: {question}
Retrieved fragments:
{Fragment 1 - {title}: {text}}
...
Please complete the following tasks:
1. Answer the question based solely on the provided fragments.
2. Cite up to THREE sources (fragment numbers) that support your answer.

Evaluation Metrics - Condition Correctness:
- Name: “Condition Correctness”
- Criteria: “Determine whether the actual condition is factually correct based on the expected condition.”
- Evaluation steps:
1. Check whether the facts in ’actual condition’ contradicts any facts in ’expected condition’.
2. Heavily penalize omission of critical details in the condition.
3. Ensure that the condition is clear and unambiguous.

Evaluation Metrics - Answer Correctness:
- Name: “Answer Correctness”
- Criteria: “Determine whether the actual answer is factually correct based on the expected answers.”
- Evaluation steps:
1. Check whether the facts in ’actual answer’ contradicts any facts in ’expected answers’.
2. Heavily penalize omission of critical details in the answer.
3. Ensure that the answer directly addresses the question without irrelevant information.