I Could've Asked That: Reformulating Unanswerable Questions

Anonymous ACL submission

Abstract

When seeking information from unfamiliar documents, users frequently pose questions that cannot be answered by the documents. While existing large language models (LLMs) identify these unanswerable questions, they do not assist users in reformulating their questions, thereby reducing their overall utility. We curate COULDASK, a benchmark composed of existing and new datasets for document-grounded question answering, specifically designed to study reformulating unanswerable questions. We evaluate state-of-the-art open-source and proprietary LLMs on COULDASK. The results demonstrate the limited capabilities of these models in reformulating questions. Specifically, GPT-4 and Llama2-7B successfully reformulate questions only 38% and 19% of the time, respectively. Error analysis shows that 62% of the unsuccessful reformulations stem from the models merely rephrasing the questions or even generating identical questions.

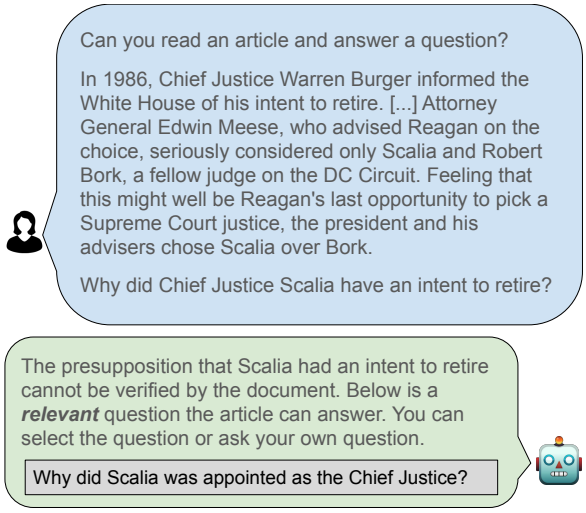

Figure 1: An example of an LLM suggesting an alternative relevant question the user could have asked whose answers can be found in the document, besides only informing users with the presupposition errors.

1 Introduction

Applying large language models (LLMs) to perform question answering (QA) over documents, such as legal and medical texts, has become increasingly popular (Agrawal et al., 2022; Guha et al., 2023). However, users' limited knowledge of these documents often results in the formulation of unanswerable questions, whose assumptions either conflict with or cannot be verified with the information available in the documents. We will refer to these assumptions as *presupposition errors*.[1] Gao et al. (2023) and Yu et al. (2023) found that around 30% of information-seeking questions written by users include presupposition errors. Research in the field has primarily focused on the detection of unanswerable questions (Rajpurkar et al., 2018; Tran et al., 2023; Hu et al., 2023) and providing explanations for why such questions cannot be answered (Yu et al., 2023; Kim et al., 2023). However, this goal is insufficient for fostering an effective interaction between users and LLMs. Identifying unanswerable questions only serves as a starting point in question reformulation; without additional guidance or feedback on how to rephrase the question, users, especially those unfamiliar with the content, might find themselves caught in a repetitive cycle of formulating questions. This work aims to improve the utility of QA systems by introducing a new task that requires both detecting unanswerable questions and generating questions closely related to the initial query and based on information in the document. This approach helps users gain a more targeted and holistic understanding of the content they are interested in. We provide an example of how to suggest such questions in Figure 1. Although generating relevant and grounded questions conditioned on initial queries offers greater utility to users, it remains a difficult task even for the best models in a few-shot setting.

[1] Kim et al. (2023) refer to these assumptions as questionable assumptions.

Issue	Strategy	%	Original question	Revised question
Contradictory	Correct	17	What other foods are adulterated with turmeric?	What other substances are commonly adulterated in turmeric?
			Which major cities are located at great lakes?	Which U.S. states have jurisdictions that extend into the Great Lakes?
Unverifiable	Generalize	20	When was the Twenty-third Amendment to the United States Constitution passed into law?	Has the Twenty-third Amendment to the United States Constitution been proposed?
			How many children does Jennifer Lawrence have?	Is Jennifer Lawrence expecting her first child?
Unverifiable	Nearest Match	44	How many arms does Krishna have?	How is Krishna typically depicted in terms of arms?
			How many schools are in CPS?	What kind of schools are included in CPS?
Unverifiable	Specify	19	While working at Edison, what inventions did Nikola Tesla make?	What specific aspects of electrical engineering did Nikola Tesla work on and further develop while working at Edison?
			How many children did Toni Morrison have?	How many children did Toni Morrison have with Harold Morrison before their divorce?

Table 1: Strategies from a human user reformulating questions on 100 examples.

We first characterize human-reformulated questions and describe several different strategies for updating questions to remove presupposition errors. Motivated by these strategies, we curate COULDASK, a benchmark for document-grounded QA that consists of a combination of existing and new datasets to study question reformulation in the presence of presupposition errors. We evaluate several prompting methods such as few-shot prompting and chain-of-thought prompting, employing state-of-the-art open-source and proprietary models. The results illustrate the limitations of existing models and prompting techniques in accurately detecting unanswerable questions and reformulating questions: the F1 scores for identifying unanswerable questions range from 43.93% to 68.35%, and success rates for reformulating questions range from 11.46% to 25.38%, depending on base models. Analysis shows that most of the unsuccessful reformulation come from rephrasing or repeating the original questions and that LLMs are worse at reformulating questions necessitating global edits compared to those solely requiring local edits.

2 Related Work

Several datasets have been proposed to study unanswerable questions. Rajpurkar et al. (2018) curate the first document-grounded QA dataset that features unanswerable questions. More recently, Yu et al. (2023) and Kim et al. (2023) have collected questions with presupposition errors from Google user queries and Reddit posts, respectively.

How to identify unanswerable questions, especially with off-the-shelf LLMs, has remained understudied. Kim et al. (2021) proposed to first extract presuppositions from a question and then perform natural language inference (NLI) to check for presupposition violations. However, this pipeline requires supervision. In practice, supervision is often not available; Kim et al. (2023) thus explore prompting large language models in the chain-of-thought style to identify unanswerable questions. However, the results remain unsatisfying; using GPT-3 only yields detection accuracy that is only slightly better than random guesses.

Unanswerable questions closely related to ambiguous questions (Min et al., 2020). While there has been extensive research into reformulating ambiguous questions (Rao and Daumé III, 2018; White et al., 2021; Pyatkin et al., 2023), the problem of reformulating unanswerable questions receives little attention. Strategies for rephrasing ambiguous questions often involve making questions more specific by mentioning precise entities or events (Min et al., 2020). In contrast, as we will show in Section 3, reformulating unanswerable questions necessitates a wide range of strategies.

Finally, we discuss the connection between document-grounded QA and open-domain QA (Kim et al., 2023). Document-grounded QA is essentially open-domain QA with the correct document retrieved. Reformulating questions based on the identified document is a separate skill from retrieving the document. Therefore, question reformulation is an interesting task by itself.

	Total	Unans%	Document Length	Question Length	# Question Entities	Source	Domain
SQuADv2	1000	50.70	143.83 ± 59.97	11.21 ± 3.54	2.29 ± 0.94	Human	Wikipedia
QA^2	506	48.81	818.28 ± 684.61	7.74 ± 1.39	2.05 ± 0.57	Human	Mostly Wikipedia
BanditQA	2070	35.56	261.09 ± 145.19	8.37 ± 2.43	1.70 ± 0.72	Human	Wikipedia
BBC	278	21.22	477.75 ± 289.41	16.37 ± 3.49	3.31 ± 1.04	GPT-4	News
Reddit	313	36.10	477.43 ± 307.18	14.27 ± 2.75	2.93 ± 0.78	GPT-4	Social Media
Yelp	165	30.91	387.70 ± 147.83	15.27 ± 3.04	3.04 ± 0.76	GPT-4	Review

Table 2: An overview of COULDASK. Unans% is the percentage of unanswerable questions.

3 Task: Question Reformulation

To assist users with question reformulation when reading unfamiliar documents, we define the following task. Given a document and a user question, the system must determine if the question is unanswerable. Upon identifying the unanswerable question, it must reformulate the question such that the new question is answerable by the document while remaining relevant to the original question.

As this task is challenging to formally define, we begin with a qualitative study over a set of example reformulations by a human user, shown in Table 1. Different strategies are applied for different presupposition errors in the reformulation process. For handling presuppositions that are contradictory to the documents, the human user corrects the presuppositions. When it comes to presuppositions that are unverifiable given the documents, we observe three strategies. The first strategy takes a step back by asking about a less specific event then event in the original question. The second strategy seeks a "nearest match" question that the document can answer due to a flaw in the original. The third strategy refines the original question by asking about something more specific that can be verified by the document. While these strategies are not exhaustive, they demonstrate the challenging nature of the problem, and the necessity for establishing sources of ground truth in the document.

4 The COULDASK Benchmark

Motivated by the need to reformulate questions both to rely on verified information and to avoid contradictions, we develop a benchmark dataset. We consider two important challenges in constructing benchmarks for this task. **(1)** The benchmark should cover a wide range of domains to cover different types of presuppositions. (Existing QA datasets that study unanswerable questions mostly rely on Wikipedia articles (Rajpurkar et al., 2018; Gao et al., 2023; Kim et al., 2023).) **(2)** The evaluation method should be capable of fairly assessing equally good reformulated questions, considering the subjective nature of question reformulation.

4.1 Datasets

Following the desiderata, we select three existing datasets – SQuADv2, BanditQA (Gao et al., 2023), and QA^2 (Kim et al., 2023) – and to cover a broader range of domains, we create three new datasets in the domains of news, review, and Reddit, where the questions are generated by models and verified by crowdworkers. We summarize statistics for all datasets in Table 2.

BBC, Reddit, and Yelp. We synthetically generate questions with a question generation model. We use the documents from BBC news articles[2], Reddit pages[3], and Yelp reviews[4], respectively. To not artificially craft unanswerable questions, we instruct the question generation model to produce questions normally and later identify the unanswerable ones. To produce a dataset that is challenging for LLMs, we additionally leverage a question checking model[5]. We search for questions that confuse the question checking model. Specifically, we sample the question checking model five times to produce an answer for each of the questions. We gather questions where the question checking model flags the questions are unanswerable any of those five times. We then ask three crowdworkers from Amazon Mechanical Turk (MTurk) to independently verify whether the question is answerable or not.[6] The question generation model pro-

[2]https://huggingface.co/datasets/SetFit/bbc-news

[3]https://huggingface.co/datasets/reddit_tifu

[4]https://huggingface.co/datasets/yelp_review_full

[5]GPT-4 is used for both question generation model and question checking model.

[6]In particular, to annotate whether a question is answerable, we instruct the crowdworkers to select a span from the document to answer the question. If they cannot identify a span, the question is deemed unanswerable. To reduce noise, we take the majority vote from three crowdworkers. To further ensure a low noise level in annotations, we remove the examples where answer spans annotated by different crowdworkers are disjoint from each other.

Example	Revised question	Ans	Rel
Question: When did Chick-fil-A open their first restaurant in Pennsylvania?	• Has Chick-fil-A opened their first restaurant in Pennsylvania?	✓	✓
Document: [...] he registered the name Chick-fil-A, Inc. From 1964 to 1967, the sandwich was licensed to over fifty eateries, including Waffle House and the concession stands of the new Houston Astrodome. The Chick-Fil-A sandwich was withdrawn from sale at other restaurants when the first standalone location opened in 1967, in the food court of the Greenbriar Mall, in a suburb of Atlanta. Since 1997, the Atlanta-based company has been the title sponsor of the Peach Bowl, an annual college football bowl game played in Atlanta on New Year's Eve.	• Which Chick-fil-A mentioned in the article is the closest to Pennsylvania?	✓	✓
	•When did Chick-fil-A open their first free-standing location?	✓	✗
	• How many eateries of Chick-fil-A are licensed from 1964 to 1967?	✓	✗
	• Where is the first Chick-fil-A restaurant in Pennsylvania?	✗	✓
	• When did Chick-fil-A open its first restaurant in a northern state, specifically in Pennsylvania?	✗	✓
	• When were chicken sandwiches invented at Chick-fil-A?	✗	✗

Table 3: Different aspects of question reformulation. Ans indicates whether the reformulated question can be answered by the document, and Rel indicates whether the reformulated question is relevant to the original question.

duces 9500, 9964, and 10000 QA pairs for the BBC, Reddit, and Yelp datasets, respectively. We keep 278, 313, and 165 examples that have confused the question checking model. Finally, the crowdworkers identify 21.22%, 36.10%, and 30.91% of the questions are truly unanswerable. We thus construct examples that are both challenging to LLMs and high-quality.

SQuADv2, QA^2, and BanditQA. We adapt these datasets to be used for document-grounded QA. Questions in SQuADv2 are mechanistically formulated with the intention of being unanswerable, whereas the other two datasets feature naturally occurring unanswerable questions. QA^2 is composed of natural Google search queries, and BanditQA includes questions formulated by users during interactions with LLMs. More information on the modifications we make to these datasets can be found in Appendix A.

4.2 Evaluating Reformulation

To improve the utility of QA systems in responding to unanswerable questions, the reformulated questions must be (1) answerable by the documents and (2) relevant to the original questions posed by the users. As illustrated by the examples in Table 3, a reformulation could be answerable but not relevant, or relevant but not answerable. A successful reformulation should satisfy both conditions. There are multiple equally good reformulations for each question. For instance, when given the original question, it is hard to determine which of the first two reformulations in Table 3 would be closer to the user's intent. As a result, we opt for a reference-free evaluation approach.

Considering the above factors, we propose two metrics: (1) whether reformulation is answerable by the document, and (2) the entity overlap ratio[7] between the original question and the reformulation to indicate the level of relevance. For the unanswerable questions that are not successfully detected by the system, we consider the reformulation to be unanswerable and have zero entity overlap. To automatically evaluate (1), we train a Llama2-7B model on COULDASK to classify whether the reformulation is answerable or not. The classifier achieved 95% accuracy on a held-out validation set. We release the classifier on Hugging Face. For (2), we use GPT-4 to tag entities. Finally, a reformulation that is answerable but irrelevant, or vice versa, is not yet helpful. To consolidate the evaluation into a single unified score, we assign a score of 1 to a reformulation if it is both answerable and has an entity overlap ratio of more than 50%. We refer to this score as *success rate*.

5 Experimental Setup

Models. We test a range of instruction-finetuned LLMs. For proprietary models, we consider GPT-3.5 (gpt-3.5-turbo-16k-0613) and GPT-4 (gpt-4-0613). For open-source models, we consider a list of 7-billion-parameter models: Llama2[8] (Touvron et al., 2023), Mistral[9] (Jiang et al., 2023), and Zephyr[10] (Tunstall et al., 2023).

[7] We justify our choice of entity overlap ratio as a relevance metric with human evaluation in Section 7.

[8] huggingface.co/meta-llama/Llama-2-7b-chat-hf

[9] huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

[10] huggingface.co/HuggingFaceH4/zephyr-7b-beta

		SQuADv2	QA^2	BanditQA	BBC	Reddit	Yelp	Average
GPT-3.5	ZS	8.88	12.15	9.38	33.90	8.85	23.53	16.11
	ZS CoT	8.09	8.91	8.97	33.90	5.31	19.61	14.13
	FS	8.09	10.93	8.02	35.59	7.96	15.69	14.38
	FS CoT	10.26	16.60	9.10	40.68	6.19	29.41	18.71
GPT-4	ZS	**18.74**	**21.86**	10.60	47.46	7.96	33.33	23.33
	ZS CoT	17.55	**21.86**	**10.46**	**52.54**	**10.62**	**39.22**	**25.38**
	FS	10.45	14.17	10.60	54.24	**10.62**	31.37	21.91
	FS CoT	12.03	16.19	10.19	45.76	**10.62**	35.29	21.68
Llama2	ZS	2.37	6.07	6.52	16.95	2.65	3.92	6.41
	ZS CoT	4.73	4.86	6.25	10.17	2.65	15.69	7.39
	FS	7.50	13.36	8.70	27.12	7.08	17.65	13.57
	FS CoT	3.75	4.86	5.30	22.03	6.19	13.73	9.31
Mistral	ZS	6.11	4.05	6.25	8.47	2.65	5.88	5.57
	ZS CoT	3.16	5.26	5.16	15.25	0.88	7.84	6.26
	FS	5.13	3.24	5.57	22.03	8.85	13.73	9.76
	FS CoT	4.93	8.50	5.03	22.03	10.62	17.65	11.46
Zephyr	ZS	7.89	8.91	8.97	28.81	5.31	13.73	12.27
	ZS CoT	10.65	11.74	9.38	35.59	3.54	19.61	15.08
	FS	5.92	11.74	7.34	16.95	3.54	11.76	9.54
	FS CoT	4.73	12.15	7.88	22.03	7.08	11.76	10.94

Table 4: Question Reformulation. Success rates using different prompting methods with both proprietary and open-source models on COULDASK. The best method for each base model is highlighted with an underscore, and the best method across all base models is bolded.

Comparisons. We consider several prompting approaches: zero-shot (ZS) and few-shot (FS) prompting and ZS and FS chain-of-thought (CoT) prompting. We first prompt LLMs to determine whether the input question cannot be answered by the provided document. For ZS and FS prompting, we use the prompt provided by SurgeAI[11], which explicitly instructs the model to not produce an answer if the answer cannot be determined from the document alone. For ZS and FS CoT prompting, we expand the aforementioned prompt by asking the model to think step by step to come up with a reason to explain and support its decision. For question reformulation, all methods are provided with their previous turns where the models determine the questions are unanswerable. ZS and FS prompting instruct the model to make minimum edits to the original question to make it answerable. ZS and FS CoT prompting instruct the model to think step by step to reason about the minimum edits they can make.

[11]SurgeAI prompt: "Answer the following from the above passage alone, and if you can't determine the answer based on the passage, say that you don't know the answer."

6 Results

Table 4 presents the success rate for question reformulation. Success rates vary dramatically from domain to domain. News (BBC) appears to be the least challenging domain, with success rates ranging from 22.03 to 52.54 depending on base models. Reddit is a challenging domain, with success rates ranging from 7.08 to 10.62. The results on Wikipedia are mixed. While SQuADv2, QA^2, and BanditQA are in the Wikipedia domain, LLMs achieve the lowest success rates on BanditQA. We hypothesize that user queries written during interaction require deeper revision.

Among all base models, GPT-4 achieves the highest average success rate (25.38), while Mistral has the lowest average success rate (11.46). Among all open-source models, Zephyr has the best performance. When it comes to prompting methods, there is not a clear winner. Different LLMs can be improved with different prompting approaches. GPT-3.5 and Zephyr benefit the most from ZS CoT

Error Category	%	Original question	Revised question
Simply rephrasing / producing the same questions	62	When was he born?	When was Jay Chou born and raised in Taipei, Taiwan?
		Which team won the most NFL Europe titles?	Who was the team that won the most NFL Europe titles?
Producing questions by copying documents	14	When did the Cubs play their first home game at Wrigley Field?	When did the Cubs move to Wrigley Field?
		what are the characteristics of the Classical Roman architecture?	What are some of the characteristics of Classical Roman architecture that were adopted by Renaissance architects?
Producing irrelevant questions	17	What is Snow Patrol's most famous song?	Who are some famous musicians that have expressed admiration for Snow Patrol?
		which campus is the biggest in University of California?	Which college in the University of California system is not controlled by the Regents or President?
Other	11	How many arms does Krishna have?	I apologize for the confusion. The passage does not mention how many arms Krishna has.

Table 5: Error analysis on 100 revised questions generated by different models.

prompting, GPT-4 from ZS CoT, Llama2 from FS, and Mistral from FS CoT. We include individual metrics for question reformulation in Appendix B.

7 Analysis

Qualitative analysis We randomly sample and analyze 20 reformulated questions from each base model (a total of 100 questions) that cannot be answered by the corresponding documents. We summarize the results in Table 5. We identify three major types of errors. The most frequent type is that the models simply rephrase or generate the same questions. Most of the errors in this type are contributed by open-source models such as Llama2.

Another type of error that occurs 14% of the time is that the models generate a question by copying a document span that looks similar to the original question. For example, given the original question about the characteristics of the Classical Roman architecture, the model identifies a span that is similar to the original question and generates a question based on this span. However, the model does not plan thoroughly about whether the new question can be answered or not.

For 17% of the time, the models propose questions that are irrelevant to the original questions. For example, in the Snow Patrol question, the subject of the reformulation has entirely changed from asking about the most famous song to famous musicians. Finally, we group the remaining eleven errors in the Other type, where there usually is not even a question produced.

Detecting unanswerable questions. Being able to detect unanswerable questions is a necessary precondition for successful question reformulation. Table 6 presents the F1 scores for unanswerable question detection using different prompting approaches with GPT-3.5, GPT-4, and Zephyr as base models. Performance is better across all LLMs for existing datasets compared to the new datasets, which indicates the effectiveness of our approach in generating more challenging questions. Among all models, GPT-4 performs best at identifying unanswerable questions. Proprietary and open-source models of similar sizes achieve comparable F1 scores. GPT-3.5 outperforms Zephyr on BanditQA, BBC, and Reddit, whereas Zephyr exceeds GPT-3.5's performance on SQuADv2, QA^2, and Yelp. Finally, among all prompting techniques, FS CoT consistently improves upon ZS, with a larger degree of improvement observed in smaller models compared to larger ones.

Limited benefit from FS CoT on Question Reformulation. We explore why FS CoT improves the detection of unanswerable questions but not question reformulation by conducting a qualitative analysis of GPT-4 outputs on a BanditQA example, as illustrated in Figure 2. It is relatively straightforward to determine why a question is unanswerable — either the document does not provide the necessary information or there is a presupposition that conflicts with the document. However, question reformulation demands compositional reasoning. The model needs to first decide on a reformulation strategy and then plan the specific steps to achieve

		SQuADv2	QA^2	BanditQA	BBC	Reddit	Yelp	Average
GPT-3.5	ZS	56.57	52.58	68.50	44.26	50.39	41.30	52.27
	ZS CoT	51.32	46.36	64.32	47.27	50.00	36.96	49.37
	FS	58.36	56.25	73.87	41.79	51.66	37.50	53.24
	FS CoT	64.73	62.41	77.55	48.48	49.60	47.06	58.31
GPT-4	ZS	**85.80**	**69.00**	69.13	52.81	60.59	56.74	65.68
	ZS CoT	84.20	67.71	68.26	52.58	57.56	58.57	64.81
	FS	77.39	63.06	79.89	**56.63**	65.45	55.07	66.25
	FS CoT	76.64	66.82	**80.40**	55.87	**67.58**	**62.77**	**68.35**
Zephyr	ZS	64.65	51.89	68.76	43.48	37.70	35.14	50.27
	ZS CoT	64.26	53.85	69.48	43.28	41.24	36.78	51.48
	FS	66.36	58.96	66.58	38.33	49.12	44.68	54.01
	FS CoT	67.14	63.55	72.04	41.38	46.77	48.00	56.48

Table 6: Comparing F1 scores for unanswerable question detection with different prompting methods using both proprietary and open-source models on COULDASK. The best method for each base model is highlighted with an underscore, and the best method across all base models is bolded.

Document:
In December 2007, Sport England published a survey which revealed that residents of St Albans were the 10th most active in England in sports and other fitness activities. 30.8% of the population participate at least 3 times a week for 90 minutes.
Question:
What is St Albans Sport?

ZS CoT for unanswerable question detection:
The article did not provide information on what St Albans Sport is, so I don't know.
ZS CoT for question reformulation:
The article discusses the activity level of the residents of St Albans in sports and fitness but does not detail a specific sport. Instead of asking "What is St Albans Sport?" we could ask, "How often do residents of St Albans participate in sports and fitness activities?" to get an answer from the given passage.

ZS for question reformulation:
What is the level of participation in sports and other fitness activities in St Albans?

Figure 2: Example outputs produced by GPT-4. Via prompting, the model first detects unanswerable questions then reformulates the questions with a second prompt.

the reformulation. The reformulated question generated by GPT-4 via FS CoT prompting is closer to being answerable. However, the model misses a subtle detail — the document only mentions the exercise habits of 30.8% of residents, not the general population. Therefore, FS CoT alone, without further methodological innovation, does not fully address the challenge of question reformulation.

Compositional Modifications vs. Answerability. We hypothesize that it is more difficult to reformulate an unanswerable question to be answerable when it requires compositional modifications, which means making global edits to the question instead of making local edits.

q span	
Original Q	**What medicine is made from Coca?**
Reformulated Q	What substance is derived from the Coca leaf?
short span	
Original Q	When was **USB-C** developed?
Reformulated Q	When was USB developed?

Table 7: Examples of a question with the minimum span being the entire question (q span) and a question where the minimum span is a noun (short span).

To quantify edits required, we follow Lee et al. (2020) and annotate the minimum span of a question to explain why the question cannot be answered by the document. We use GPT-4 to annotate unanswerable questions in BanditQA, QA^2, and Yelp. We divide the questions into two categories. The first category comprises questions whose minimum span is the entire question. We call these *q span*. The second category covers questions that have a shorter minimum span, typically a noun or a noun phrase. We call these *short span*. We calculate how many reformulations are answerable, broken down by the type of minimum span. We consider the reformulations generated by zero-shot prompting with GPT-4, GPT-3.5, and Zephyr.

The results of this analysis are summarized in Figure 3. Our findings show a consistent pattern that fewer reformulations are answerable when the minimum span constitutes the entire question compared to when the minimum span is shorter, with only one exception in zero-shot prompting with Zephyr on Yelp.

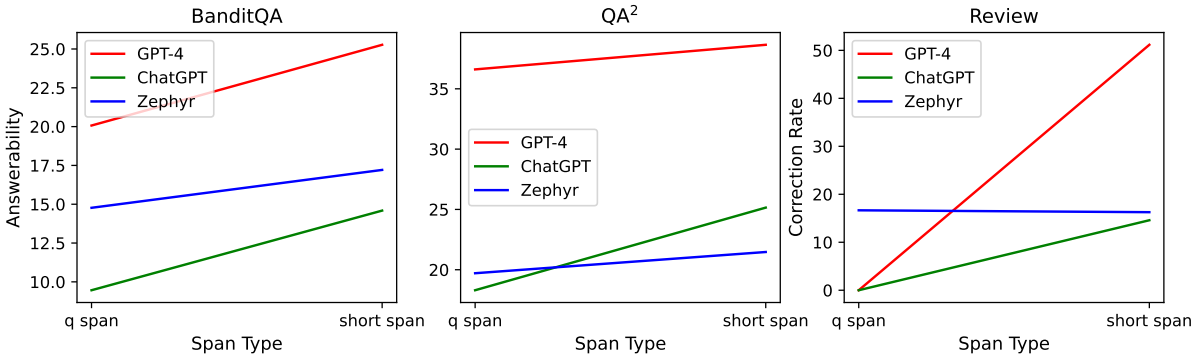

Figure 3: The relation between the percentage of reformulations answerable by the documents and the type of minimum spans in the original questions. q span are the examples where the minimum span is the full question, while short span includes all other instances.

	Edit Distance	Entity Overlap
Fleiss' κ	40.96	93.99

Table 8: The correlation between the tested metrics and human-judged relevance between two questions, evaluated using Fleiss' κ.

We present examples for each span type in Table 7. When the minimum span is the entire question, it implies that we cannot attribute the presupposition error to a smaller segment of the question. As a result, the modification has to be applied to the full question. For example, for the q-span example in Table 7, modifying individual parts of the question cannot correct the presupposition error, and therefore the model needs to make global changes. On the contrary, when the minimum span is short, it only requires local edits. For the short-span question in Table 7, replacing USB-C with USB is enough to correct the question. Future efforts should be more devoted to the more challenging cases to make progress on question reformulation.

Sufficiency of entity overlap ratios. Our metric is for relevance is based on a minimum entity overlaps. To judge the sufficiency of this metric, we compare it to human evaluation. As a baseline we consider Levenshtein edit distance, a method to measure the similarity between two questions, as a baseline metric. We have a human annotator evaluate 200 pairs of reformulated questions produced by zero-shot prompting with GPT-4 and GPT-3.5 on BanditQA. The question pairs are randomly shuffled to ensure the annotator remains unaware of the model source of each reformulation. The annotator then selects the more relevant question from each pair, based on alignment with the original question's intent. Subsequently, we identify the question with a higher entity overlap ratio and the question with a lower edit distance as the more relevant ones, respectively.

We calculate the Fleiss' κ score between human evaluations and each of the metrics, with the results summarized in Table 8. Compared to edit distance, the entity overlap ratio more accurately represents the relevance between the original and reformulated questions (93.99 vs. 40.96). A Fleiss' κ score of 93.99 also suggests a near-perfect agreement between the entity overlap ratio and human judgements. We hypothesize that the specific semantic properties of the questions are more central than the surface form represented by edit distance.

8 Conclusion

Users often ask unanswerable questions when they seek information from unfamiliar documents. Existing LLMs identify these questions but do not aid users in reformulating new questions, resulting in ineffective user-LLM interactions. We introduce COULDASK, a compiled set of grounded-document QA datasets designed to study both unanswerable question detection and question reformulation. Our results demonstrate the limited capabilities of state-of-the-art LLMs in reformulating questions. Using ZS CoT prompting, GPT-4 only manages to achieve a success rate of 25%. This suggests a significant potential for improvement in these models, which will in turn improve user-LLM interaction.

Limitations

While our benchmark offers advantages over existing sources, we acknowledge the following limitations. Questions in BBC, Reddit, and Yelp are generated by GPT-4, and they may not accurately represent questions posed by humans. Despite best efforts to ensure high-quality annotations, occasional human errors are possible. Additionally, our benchmark only collects English questions and thus lacks language diversity. Finally, regarding evaluation, the way we currently measure success rates only focuses on mistakes made on unanswerable questions. If an answerable question is detected to be unanswerable, we do not evaluate question reformulation in such cases.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1998–2022, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ge Gao, Hung-Ting Chen, Yoav Artzi, and Eunsol Choi. 2023. Continually improving extractive QA via human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 406–423, Singapore. Association for Computational Linguistics.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Re, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, Jessica Wu, Joe Nudell, Joel Niklaus, John J Nay, Jonathan H. Choi, Kevin Tobia, Margaret Hagan, Megan Ma, Michael Livermore, Nikon Rasumov-Rahe, Nils Holzenberger, Noam Kolt, Peter Henderson, Sean Rehaag, Sharad Goel, Shang Gao, Spencer Williams, Sunny Gandhi, Tom Zur, Varun Iyer, and Zehua Li. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Shengding Hu, Yifan Luo, Huadong Wang, Xingyi Cheng, Zhiyuan Liu, and Maosong Sun. 2023. Won't get fooled again: Answering questions with false premises. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5626–5643, Toronto, Canada. 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*.

Najoung Kim, Phu Mon Htut, Samuel R. Bowman, and Jackson Petty. 2023. $(QA)^2$: Question answering with questionable assumptions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8466–8487, Toronto, Canada. Association for Computational Linguistics.

Najoung Kim, Ellie Pavlick, Burcu Karagol Ayan, and Deepak Ramachandran. 2021. Which linguist invented the lightbulb? presupposition verification for question-answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3932–3945, Online. Association for Computational Linguistics.

Gyeongbok Lee, Seung-won Hwang, and Hyunsouk Cho. 2020. SQuAD2-CR: Semi-supervised annotation for cause and rationales for unanswerability in SQuAD 2.0. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5425–5432, Marseille, France. European Language Resources Association.

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.

Valentina Pyatkin, Jena D. Hwang, Vivek Srikumar, Ximing Lu, Liwei Jiang, Yejin Choi, and Chandra Bhagavatula. 2023. ClarifyDelphi: Reinforced clarification questions with defeasibility rewards for social and moral situations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11253–11271, Toronto, Canada. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.

Sudha Rao and Hal Daumé III. 2018. Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2737–2746, Melbourne, Australia. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay

Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Son Quoc Tran, Gia-Huy Do, Phong Nguyen-Thuan Do, Matt Kretchmar, and Xinya Du. 2023. Agent: A novel pipeline for automatically creating unanswerable questions. *arXiv preprint arXiv:2309.05103*.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Julia White, Gabriel Poesia, Robert Hawkins, Dorsa Sadigh, and Noah Goodman. 2021. Open-domain clarification question generation without question examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 563–570, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xinyan Yu, Sewon Min, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. CREPE: Open-domain question answering with false presuppositions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10457–10480, Toronto, Canada. Association for Computational Linguistics.

A Dataset Details

We provide more details about the existing datasets we include in COULDASK.

SQuADv2 is a collection of QA pairs from Wikipedia articles, including questions with presupposition errors. Such questions are written by crowdworkers who are instructed to craft questions that are plausible but have presuppositions that cannot be verified by the associated articles. These annotators either modify valid questions or create new questions based on entities or topics related to the text, ensuring the questions appear relevant and deceivingly valid.

QA^2 contains QA pairs from general web articles. QA^2 sources questions from Google's autocompletion API. It calls the API on prefix strings – when, where, which, how, what, why, who, whose – to complete the queries. Crowdworkers are recruited to (1) search for relevant documents and (2) check whether these questions contain presupposition errors. QA^2 features naturally occurring presupposition errors and diverse document sources. The original QA^2 dataset is open-domain QA and only provides annotations for relevant URLs. To obtain the context, we scrape the text from their annotated URL and remove the examples where the contexts are not clear.

BanditQA investigate whether continual human feedback can improve extractive QA systems; we repurpose the dataset to study naturally occurring presupposition errors. During data collection, users are presented with Wikipedia passages, and they are required to write questions about things they are curious to know. In a later verification step, over 30% of these questions are identified to have presupposition errors. BanditQA is closest to the setting of this study – when users are unfamiliar with the documents, they come up with presuppositions that cannot be grounded in the documents.

B More Results on Question Reformulation

We present individual metrics for question reformulation. Table 9 presents the answerability (top) and entity overlap ratios (bottom) achieved by each method using each base model. GPT-4 can most often reformulate answerable questions (38.73) and propose the most relevant questions (41.91). Zephyr excels in suggesting questions with fewer

		SQuADv2	QA^2	BanditQA	BBC	Reddit	Yelp	Average
				Answerability				
GPT-3.5	ZS	13.81	23.89	12.50	38.98	9.73	31.38	25.52
	ZS CoT	11.05	18.22	11.69	40.68	6.19	21.57	18.63
	FS	13.02	27.53	16.03	40.68	9.73	21.57	21.34
	FS CoT	16.17	32.79	18.48	50.85	7.08	33.33	26.45
GPT-4	ZS	**28.01**	**36.84**	23.10	71.19	13.27	47.06	36.58
	ZS CoT	26.04	34.82	21.47	71.19	14.16	49.02	36.12
	FS	21.50	30.36	**24.32**	74.58	15.04	47.06	35.48
	FS CoT	20.51	35.63	**24.32**	**76.27**	**16.81**	**58.82**	**38.73**
Llama2	ZS	3.94	10.53	10.6	18.64	2.65	5.88	8.71
	ZS CoT	6.11	10.53	9.51	13.56	3.54	19.61	10.48
	FS	12.23	24.70	13.32	33.90	7.96	27.45	19.93
	FS CoT	6.31	10.12	7.88	28.81	9.73	15.69	13.09
Mistral	ZS	9.47	7.29	9.92	11.86	4.42	9.80	8.79
	ZS CoT	7.10	10.93	9.10	16.95	4.42	9.80	9.72
	FS	9.27	12.15	10.60	30.51	11.50	17.65	15.28
	FS CoT	9.47	17.00	11.55	32.20	13.27	25.49	18.16
Zephyr	ZS	16.77	21.46	16.17	44.07	8.85	15.69	20.50
	ZS CoT	17.55	23.08	16.03	44.07	7.08	23.53	21.89
	FS	19.92	27.94	18.75	35.59	7.08	21.57	21.81
	FS CoT	20.91	34.82	19.70	44.07	13.27	31.37	27.36
				Entity Overlap Ratios				
GPT-3.5	ZS	22.78	16.16	34.32	31.75	44.54	25.98	29.26
	ZS CoT	21.15	12.21	31.91	32.85	38.72	27.78	27.44
	FS	20.11	16.13	32.59	36.30	46.31	28.95	30.06
	FS CoT	24.73	20.45	37.49	37.60	36.73	33.69	31.78
GPT-4	ZS	**38.64**	**24.87**	29.13	45.00	44.73	48.69	38.51
	ZS CoT	37.81	23.82	28.54	**63.04**	42.37	**55.88**	**41.91**
	FS	25.33	17.38	31.60	47.20	46.17	40.78	34.74
	FS CoT	27.19	18.79	32.80	53.83	**50.66**	48.79	38.68
Llama2	ZS	5.10	8.10	18.61	15.42	7.96	9.15	10.72
	ZS CoT	8.66	8.10	19.09	8.47	10.55	19.28	12.36
	FS	22.28	17.61	**34.76**	20.56	27.30	23.86	24.40
	FS CoT	12.44	8.47	27.38	20.65	18.60	15.52	17.18
Mistral	ZS	11.14	6.07	16.32	8.45	6.93	8.82	9.62
	ZS CoT	10.68	7.76	15.95	14.44	10.18	7.52	11.09
	FS	11.17	7.09	18.01	21.50	27.80	21.73	17.88
	FS CoT	11.40	11.13	17.80	22.43	21.98	21.27	17.67
Zephyr	ZS	23.63	13.83	33.54	26.44	18.73	16.99	22.20
	ZS CoT	22.34	15.18	35.50	32.12	21.39	18.79	24.22
	FS	13.17	13.66	28.79	13.33	19.32	19.44	17.95
	FS CoT	10.53	14.44	29.37	23.02	19.69	16.99	19.01

Table 9: Question reformulation quality broken down by answerability (top) and entity overlap ratios (bottom) on COULDASK. The best method for each base model is highlighted with an underscore, and the best method across all base models is bolded.

presupposition errors, becoming the second best at reformulating questions. Overall, FS CoT prompting generates the fewest presupposition errors with four out of five base models. For proposing the most relevant reformulation, there is not a clear winner. GPT-3.5 achieves the highest entity using ZS CoT prompting. Meanwhile, GPT-4, Mistral, and Zephyr use FS CoT, and Llama2 utilizes FS.

C More Results on Unanswerable Question Detection

We include results on unanswerable question detection using Llama2 and Mistral in Table 10. Both of these models perform worse than Zephyr.

		SQuADv2	QA^2	BanditQA	BBC	Reddit	Yelp	Average
Llama2	ZS	16.76	32.34	47.08	27.37	15.71	18.91	26.36
	ZS CoT	25.63	33.33	46.88	17.20	21.94	32.10	29.51
	FS	55.00	53.04	67.37	30.66	40.17	40.00	47.71
	FS CoT	36.08	36.69	58.38	31.48	35.35	28.58	37.76
Mistral	ZS	36.65	26.50	47.27	22.78	19.31	20.90	28.90
	ZS CoT	37.34	31.93	44.47	26.19	24.16	21.92	31.00
	FS	40.06	37.28	52.44	36.04	42.02	32.70	40.09
	FS CoT	42.41	46.90	57.96	41.07	38.10	37.11	43.93

Table 10: Comparing F1 scores for unanswerable question detection with different prompting methods using Llama2 and Mistral on COULDASK. The best method for each base model is highlighted with an underscore, and the best method across all base models is bolded.