

Better Results Through Ambiguity Resolution: Large Language Models that Ask Clarifying Questions

Bernadette Tix¹[0009-0005-3460-7657] and Kim Binsted¹[0000-0003-4580-6166]

¹ University of Hawaii at Manoa, Honolulu HI 96822, USA

Abstract. Here we present a pilot study on the Clarifying Questions Document Generator (CQDG), an AI document-generation application designed to ask follow-up questions to the user after receiving their initial prompt. Study participants wrote a prompt requesting the AI to generate a short document such as an email, letter, or other short document. The AI then generated follow-up questions and engaged in a short question-and-answer dialog before creating the requested document. This study examines users' willingness to engage in a question-and-answer exchange with an AI, as well as their satisfaction with the output of this exchange compared to a baseline output generated using only the users' original prompts. It was predicted that users would prefer the output that included the solicited information over the baseline result. However, the initial results suggest that there was little to no overall improvement in the final output, with about half of users preferring the baseline output to the result of the question-and-answer exchange. This paper will discuss possible reasons for this result as well as suggestions for how future systems could be improved, which will be incorporated into a larger study later this year.

Keywords: Large Language Models, Clarifying Questions, Generative AI.

1 Introduction

Ambiguity has historically been a challenge in Natural Language Processing (NLP) and continues to present obstacles for modern systems. Large Language Models (LLMs) generate output by calculating a "most likely" response to any given input, with inputs usually given in the form of a prompt from a human user. However, prompts are often ambiguous, and even the best possible prediction cannot fully resolve underspecified prompts [1]. Even in a conversation between two humans, both speaking the same language and communicating clearly, misunderstandings are common. While there are many ways to resolve ambiguity in a human conversation, perhaps the most obvious way is to ask for clarification. However, commonly used LLM systems such as ChatGPT, Bard, and Bing do not ask clarifying questions in response to ambiguous prompts.

This is a problem in fields where precision is important. Detailed discussions, including follow-up questions, are a necessary part of human communication in such

fields. Even for simple requests, a lack of follow-up questions can lead to suboptimal answers or cause the LLM to misunderstand the user’s needs.

LLMs are capable of identifying ambiguity in user prompts and forming questions in response to ambiguity when prompted to do so. This has already been demonstrated with AIs answering simple ambiguous questions [2, 3], but has not yet been demonstrated with AIs intended to generate longer-form responses such as letters or documents. We propose that an LLM-based system that asks clarifying questions when needed will produce content that is more closely aligned to the desires of the human user than a comparable system which asks no clarifying questions.

2 Background

Ambiguity has historically been a challenge for NLP, including parsing [4–6], Named Entity Recognition (NER) [7–9], story understanding [10–13], and numerous other NLP tasks. In the past decade, neural models have been employed to create software that can exhibit reading comprehension-like behaviors on ambiguous, natural language text. Recurrent Neural Networks (RNN) [14, 15] and Long Short-Term Memory Networks (LSTM) [10, 16–18] demonstrated successes with specialist systems dedicated to specific NLP tasks.

Large Language Models (LLM) offer a more general solution than specialized LSTM systems. Over the past several years LLMs have improved dramatically, demonstrating state-of-the-art performance in multiple areas of NLP, matching or outperforming specialized LSTM-based systems on several NLP benchmarks [19–21] including answering questions about children’s stories [22], common-sense reasoning [23], reading comprehension [24], translation, and summarization [21], among others.

Output from LLMs can be further improved by the introduction of “chain-of-thought reasoning” [25], in which the LLM is prompted to write out a full logical argument for its conclusion in small steps, rather than skipping straight ahead to the final conclusion. Chain-of-thought reasoning in LLMs leads to fewer hallucinations, more factually correct responses, more advanced reasoning, and improved ability to solve puzzles or trick questions. LLM systems operating on a chain-of-thought model also have the potential to explain the reasoning that led them to a given conclusion, which is considered to be a desirable trait in both logical and ethical reasoning, and a necessary step for humans to trust the results of an analysis [26, 27].

2.1 Context, Ambiguity, and User Needs

Understanding context is necessary for accurate reasoning and communication. When precision is needed in human communication, a wide variety of methods are used to clarify what would otherwise be inherently ambiguous language. For example, when gathering requirements for new software, a high degree of precision is needed, usually far more than is initially provided, which is the motive for the phase of ‘requirements gathering’ within software engineering. Requirements gathering has been researched at length, and often employs a wide variety of techniques including questionnaires, face-

to-face dialogs between customer and developer, and various exercises designed to improve user engagement with the requirements gathering process [28, 29]. None of this would be necessary if software engineers could reliably get good results by simply asking users to “please state your requirements clearly!” One of the key goals of requirements gathering is to understand the context of the desired software, for example, what problem the software is needed to solve, and what specific change or improvement it is hoped the software will achieve.

Prior research in conversational interfaces has shown that better results can be achieved when the full context of a conversation is considered, not just the immediate prompt [30]. Systems such as ChatGPT are in a disadvantaged position with regards to context, as users can enter any prompt on any topic, and the LLM must provide a response with no knowledge of who the user is or why they are asking. It is unreasonable to assume that an LLM could interpret the user’s intended meaning in the absence of context when this is not possible even in communication between humans, using inherently ambiguous and context-dependent language. We believe that asking clarifying questions is a skill which LLMs must master if they are to communicate clearly and precisely with humans.

2.2 Prior Work

Several recent works have addressed the concept of LLMs using clarifying questions. The CLAM architecture [2] presents a method for using an LLM to assess ambiguity, generate a clarifying question if needed, and then generate an answer based on the user’s response to the question. CLARA [3] showed that a similar framework could be used to interpret user commands given to a robotic arm. ClarifyDelphi [31] uses clarifying questions to assist in context-sensitive ethical reasoning. Zhang et al. 2023 present a framework for asking clarifying questions before retrieving data from a database [32]. ClarifyGPT [33] demonstrates the benefits of asking clarifying questions for LLM code-creation tasks. Follow-up questions have been shown to be effective at steering the conversation in automated surveys conducted by LLM chatbots [34].

Our research differs from these prior works in several ways:

1. We examine the overall quality of generated text documents. The ability to create and modify original documents is a key strength of LLMs over earlier AI systems.
2. In our research, human users rate the quality of documents based on their own needs and subjective judgements. This is a realistic scenario for documents generated for human use. Prior research has relied heavily on “simulated humans” modeled by AI or automated metrics which may not reflect users’ subjective experience of document quality.
3. We also examine users’ willingness to engage in question-and-answer dialog with the AI. It is hoped that the AI asking follow-up questions will encourage users to engage more deeply with their own prompts and with the document creation process. However, it is also possible that users will find the process annoying or arduous.

2.3 Existing Benchmarks and Evaluation Methods

There are many benchmarks currently in use for the evaluation of LLMs. Many of the common benchmarks, such as the BLEU benchmark [35] and BERTScore [36], measure the overall quality of the generated text. However, they do not measure how well the output corresponds to the initial prompt or to a user-desired outcome. Other benchmarks test the LLM’s ability to give the correct answer to questions with previously established correct answers, including numerous question-answer (QA) datasets [37]. Some QA datasets target specific types of questions, including CoQA for Conversational Question Answering [24], TruthfulQA for misleading questions [38], and the Children’s Book Test for reading comprehension of short stories [22]. These styles of benchmark are poorly suited to determining whether a generated content has fulfilled a user’s needs. Measuring the overall quality of the text, as BLEU and BERTScore do, does not tell us whether the high-quality text has solved the user’s problem or merely provided elegant but irrelevant prose. QA datasets are only suitable for measuring the LLM’s ability to produce short, accurate responses to questions with objectively right and wrong answers. This is not suitable for the evaluation of longer-form content. A letter, essay, or short story cannot be objectively classified as “correct” or “incorrect.” The overall quality of such a document can only be measured subjectively, by the evaluation of the reader.¹

Validation of generative models for visual art and music may offer some guidance here. As with long-form textual content, the quality of visual art and music cannot generally be objectively evaluated. Furthermore, such systems are most often employed in the task of generating content (art or music) from a short textual prompt, and quality of these systems must be evaluated on how closely the output matches the intent behind the prompt given to the model. Despite the challenges associated with subjective analysis by human evaluators, including higher costs and challenges with methodology and sample size, it is often the only way to gain reliable feedback on the quality of output from creative systems [39, 40]. For instance, the experiments which validated the quality of DALL-E had human evaluators rate images for both realism and accuracy relative to each image’s corresponding prompt [41].

3 System Architecture

For this study we created a web-based application called the Clarifying Questions Document Generator (CQDG). The key components of CQDG are:

- A user-facing front-end.
- A back-end powered by OpenAI’s GPT3.5 API.
- A database for logging results from the use of the system.

¹ In some cases, an objective measure may be possible for documents with a purpose, such as whether a generated resume resulted in an interview in a job application. However, in general, user assessment of document quality is subjective.

CQDG was designed as an interface between the user and the OpenAI API. CQDG applies specific prompt-engineering templates to the user’s questions, prompts, and responses to induce GPT3.5 to identify ambiguity, generate follow-up questions, and ultimately produce a final output that considers both the original user prompt as well as the additional information from the ensuing conversation. In some cases, the API is prompted multiple times to produce multi-step results for ambiguity analysis before the user is shown only the final response of a small sequence of API interactions. In other cases, the API is given a modified version of the user’s original prompt decorated with specific prompt engineering to steer the response. To the user it appears as if each of their inputs is given just one output in direct response to what they wrote, just as when chatting with ChatGPT directly, although several interactions between the web page and the GPT API are actually taking place during each step of the process without being shown to the user. The process of generating follow-up questions is similar to that shown by the CLAM model (Kuhn, Gal, and Farquhar 2023) [2].

The baseline document is generated by providing the OpenAI API with an unmodified version of the user’s original prompt. The experimental output, hereafter referred to as the QA Document, is generated by providing GPT3.5 with the full context of the original prompt, the follow-up questions, and all user responses. An example of a complete log of prompts and responses sent and received from GPT3.5 is provided in the appendix.

4 Methodology

4.1 Experiment Design

Since CQDG relies on user interaction in the form of question asking and answering, the use of large static databases of question-answer sets is insufficient to test this design. Direct interaction between CQDG and human users is necessary. So, participants are directed to a public website hosting CQDG, which guides them through the experiment. Participants complete the study either on a Zoom call with a researcher or in person with the researcher in the room. The participant is asked to narrate out loud their thought process and any challenges or difficulties they encounter using the system, and the researcher takes notes on any feedback given by the participants. On Zoom calls, participants are asked to share their screens so the researcher can observe their interactions with CQDG.

Step 1 Explanation and consent². CQDG shows the user an explanation of the experiment, and then asks for the user’s consent to participate in the study, with an explanation of what data will be collected and how it will be used.

Step 2: Demographic Questions. The user is asked a small set of demographic questions. For the small sample size of this pilot study, we were not able to draw

² This experiment design was approved by the University of Hawaii Institutional Review Board.

conclusions about how different groups respond to the system. However, we hope that this data will be valuable in the full study. The demographic questions are:

- Age
 - [Numerical Input]
- Gender
 - “Female”
 - “Male”
 - “Other / Nonbinary”
- “What is your prior experience with generative AI such as ChatGPT, Bard, or similar programs?”
 - “I use generative AI regularly.”
 - “I have used generative AI before, but not often.”
 - “I have never used generative AI before.”
- “Is English your primary spoken language?”
 - “Yes”
 - “No”

Step 3: Instructions. The user is shown the following instructions:

“Think of a writing task you would like the AI to help you produce. This can be a document you actually need (you will have the opportunity to keep the output) or something you only think up for the sake of the experiment. Either way, please think in detail about what you want the AI to write for you before proceeding to the next step. When you have a clear idea of what you want to ask the AI to write, enter a 1-sentence or 2-sentence prompt in the textbox below, asking the AI to write your document for you. The AI will ask you a series of questions, and you will then be given two versions of the document you requested, and asked for feedback on which version you prefer.”

A text-entry area is provided for the user to enter their prompt.

Step 4: Follow-up Questions. After the user enters their initial prompt, CQDG presents the user with three clarifying follow-up questions generated by GPT 3.5 based on the user’s prompt, along with a text-entry field for the user to enter their response.

Step 5: Document Output. After all questions have been answered, CQDG uses GPT3.5 to generate two versions of the requested document. One version uses only the user’s original prompt to generate the document (baseline). The other version additionally uses the responses to the follow-up questions (QA Document). The outputs are presented to the user in random order, one at a time. When the user is shown each document, they are asked to rate the document according to three metrics, each evaluated on a scale of 1-5:

- How close is this document to what you hoped for when you made your initial request?
 - (5) Very close to what I was hoping for.
 - (4) Somewhat close to what I was hoping for.
 - (3) A little bit like what I was hoping for.
 - (2) Not very close to what I was hoping for.
 - (1) Not at all what I wanted.

- How useful would this document be to you?
 - (5) I could use this document as-is.
 - (4) I could use this document with minimal modification.
 - (3) I could use this document with substantial modification.
 - (2) This document could be used as a general starting point but requires major revisions to be usable.
 - (1) This document is not usable at all.
- How would you rate the overall quality of this document?
 - (5) Excellent quality.
 - (4) Above average quality.
 - (3) Average quality.
 - (2) Below average quality.
 - (1) Poor quality.

Step 6: Optional Continuation and Exit Questionnaire. After ranking each output with the three questions listed above, the user is shown an exit questionnaire with the following questions:

- *Please rate the following statements on a scale of “Strongly Agree” to “Strongly Disagree” (Each of the following statements is shown with 5 options and analyzed as a scale score of 1-5: 5-Strongly Agree, 4-Slightly Agree, 3-Neutral, 2-Slightly Disagree, 1-Strongly Disagree)*
 - *It was annoying to have to answer questions even though I had already explained what I wanted the AI to do.*
 - *I felt like the AI was more engaged with my problem because it asked follow-up questions.*
 - *I would be willing to answer follow-up questions from an AI if answering questions led to better results.*
 - *I liked that the AI showed me two options to pick between, instead of only picking the option it thought was best.*
- *Do you have any additional feedback or comments (optional)?*
 - A free-text entry is provided.

5 Results

A total of eight participants completed the pilot study. Although participants were not prompted to complete the study multiple times, several participants specifically requested to run the study again with different prompts immediately after completing the study for the first time. This was allowed, and the eight participants completed the study a total of fourteen times. This is not a sufficient sample size to draw statistically significant conclusions about the overall effectiveness of CQDG. However, as a pilot study, the primary goal was to inform the design of a follow-up study with a much larger sample of participants completing the study without direct supervision from the researchers.

5.1 Participant Responses

Document Ratings. As shown in Fig. 1, participant ratings for the document resulting from the question-and-answer process were similar to the ratings given for the baseline output which was generated using only the original prompts.

Exit Survey. As shown in Fig. 2, participants responded positively to the question-and-answer process overall. Participants did not express annoyance at being asked additional questions before receiving their output, and overall felt positively about the question-answering experience.

Completion Time. The average time to complete the study, measured from the acceptance of the consent to the completion of the exit survey, was 16 minutes 46 seconds. However, there was substantial variation in completion time, with the shortest time being 6 minutes 19 seconds and the longest being 41 minutes 35 seconds. This is to be expected, since participants were free to enter their own prompts and give as brief or as detailed answers as they desired for the question-answering phase. Most of the difference in completion time is explained by the difference in time spent entering answers with varying levels of detail. The longest completion time was for a user who requested a complete resume of a long musical career and gave substantial details in their prompt and answers. The shortest was for a user who requested a haiku and gave very short and general guidance in their prompt and answers.

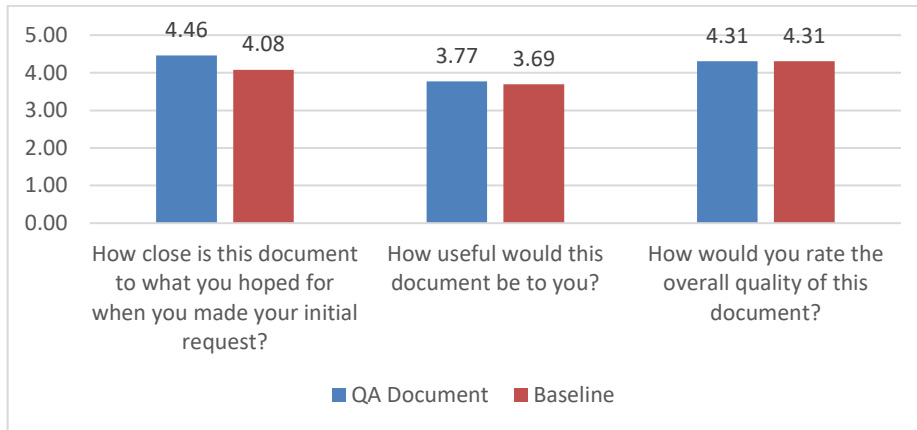


Fig. 1. Average document ratings given by participants. Note that this sample size is not large enough to be statistically significant, thus no error bars have been included.

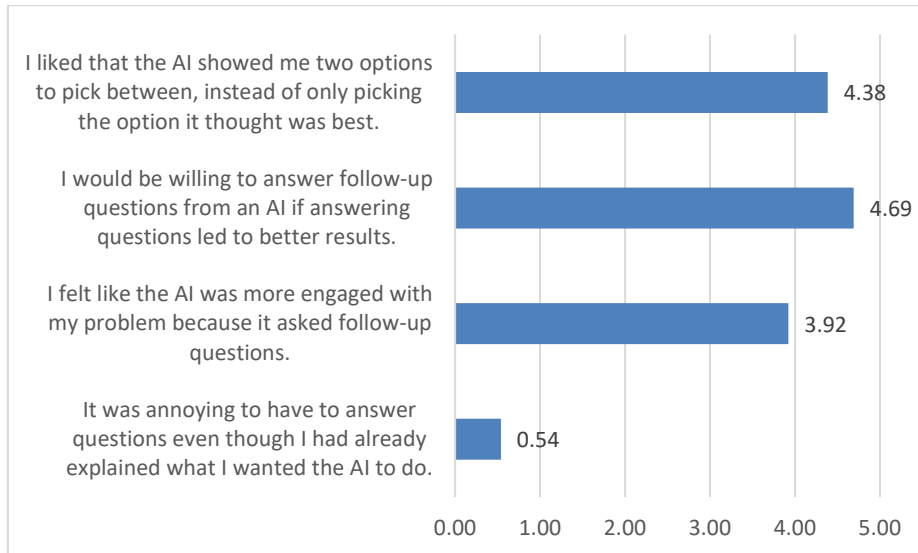


Fig. 2. Exit survey results. Overall, participants responded positively to answering questions from the AI and did not find the process annoying. Note that this sample size is not large enough to be statistically significant, thus no error bars have been included.

6 Discussion

Absolute vs Relative Measures. Participants in this study were asked to rate the quality of the produced document on an absolute scale with five options. For both the baseline and QA documents, most participants felt positively overall, but were not completely satisfied with either document, which led to most responses being in the upper half of the scale (3-5) leaving little room to differentiate the documents. Even in cases where participants expressed verbally or in written feedback that they liked/disliked some aspect of a document, this was often not reflected in the scores.

Engagement and Insight from Follow-Up Questions. Several participants expressed that the follow-up questions themselves introduced new ideas or caused them to think about aspects of their request that they had not previously considered. Participants indicated that this was a benefit of the question-answering process. Conversely, questions which asked for simple information such as the user’s name or organization were not considered helpful by participants. Baseline documents often included tags such as [your name] and [name of organization] and participants did not see a benefit to giving this information in the interactive phase rather than entering the information later.

Novel Ideas in Baseline Documents. The baseline documents included a greater variety of content that did not come directly from the participants’ prompts or responses. Given limited information to work with, GPT often produces plausible outputs that surprises participants or takes a direction they had not previously thought of. While this

was undesired in some cases, in other cases the participants expressed finding the originality to be useful and insightful. This is in line with previous findings that LLMs often perform surprisingly well at underspecified tasks [42].

Rigid Outputs from QA Documents. Conversely, the outputs that were generated using both the original prompts as well as the questions and answers typically included far less original content and often copied pieces of the participants' answers verbatim, resulting in a document that closely adhered to the participants stated needs but offered little originality. Participants expressed valuing the insight from the questions themselves, which often contained ideas they had not thought of, but this insight and originality did not carry forward into the final output.

7 Future Work

This study was designed as a pilot for a study which will include a larger sample size and allow participants to complete the survey without the direct supervision of a researcher. Based on the results of the pilot, the larger study will:

1. Allow users to read both documents and then indicate preference for one document or the other, rather than asking users to rate the documents one at a time.
2. Use higher resolution on rating scales. The 1-5 scale proved to be insufficiently sensitive.
3. Refine the prompt engineering of the sequences input to GPT. Ideally, the final output should take participants' responses into account while retaining a degree of originality, without copying participant answers verbatim.
4. Gear questions towards encouraging users to think about their needs in ways they had not previously considered or proposing expansions or alternatives, rather than gathering information that the user could easily enter into a template form (e.g. the name of their organization).
5. Provide a way to continue refining the documents after their initial creation. Several participants, especially those with prior experience with generative AI, specifically requested the ability to continue refining the outputs they were given with new instructions.
6. Compare GPT 3.5, GPT 4, and other LLMs. GPT 3.5 was only used in this case for simplicity due to the small number of participants.

8 Conclusion

We have proposed that using LLMs to generate follow-up questions can lead to superior output for text documents generated by the LLM. However, initial results do not show an obvious advantage of the QA documents over the baseline. The primary disadvantage faced by CQDG was that the QA documents focused heavily on the users' answers and did not generate as much original content as documents generated from the prompt alone. This issue could be solved by modifying the prompt engineering in

the templates that present the users' prompts and responses to the LLM. The intent behind this pilot study was to investigate users' response to CQDG and these insights will inform the design of a larger study to be conducted later this year.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

Appendix: Example Prompt and Response Log

An example of the QA Document generation process is shown below. This example was invented for the purposes of this paper and is not an actual entry by one of the study participants. Participants' answers are kept confidential by our privacy agreement. **Bold** text is for clarification only and is not part of the actual prompts or responses.

User Enters: Write a lesson plan on the Apollo 11 mission for a middle school science class.

Sent to GPT: You are a helpful AI assistant used to generate short documents. A user is requesting the creation of a new document. This is their request: user: "Write a lesson plan on the Apollo 11 mission for a middle school science class." Identify any areas of significant ambiguity or necessary information that has not been included, and write these out in a short list. Include exactly 3 items in the list.

GPT Response: 1. Duration and schedule: It is important to specify the duration of the lesson plan, such as whether it is designed for a single class period or a series of classes. Additionally, it would be helpful to know the desired schedule for the lesson plan, including any specific timing for different activities within the plan. 2. Required resources: Clarify whether there are any specific resources available for the lesson plan, such as textbooks, videos, or online materials. Knowing what resources are accessible will help in developing appropriate activities and assignments. 3. Prior knowledge and grade level: Identify the prior knowledge expected from the middle school students and their grade level. This will enable customizing the content and complexity of the lesson plan to suit their understanding and abilities.

Sent to GPT: Consider the following request: user: "Write a lesson plan on the Apollo 11 mission for a middle school science class." and the following identified points of ambiguity: "1. Duration and schedule: It is important to specify the duration of the lesson plan, such as whether it is designed for a single class period or a series of classes. Additionally, it would be helpful to know the desired schedule for the lesson plan, including any specific timing for different activities within the plan. 2. Required resources: Clarify whether there are any specific resources available for the lesson plan, such as textbooks, videos, or online materials. Knowing what resources are accessible will help in developing appropriate activities and assignments. 3. Prior knowledge and grade level: Identify the prior knowledge expected from the middle school students and their grade level. This will enable customizing the content and complexity of the lesson plan to suit their understanding and abilities." Respond as though this request was just made by the user. Start with

a friendly greeting, then ask a respectful and concise question to clarify the user's needs, keeping in mind the ambiguity you just identified. Format your response as a numbered list of exactly 3 questions.

GPT Response: Hello! I'd be happy to help you create a lesson plan on the Apollo 11 mission for a middle school science class. To ensure I fulfill your needs, I have a few questions: 1. Duration and schedule: Could you please let me know the duration of the lesson plan? Is it designed for a single class period or a series of classes? Additionally, do you have any preferences or specific timing for different activities within the plan? 2. Required resources: Are there any specific resources available for the lesson plan, such as textbooks, videos, or online materials? Knowing what resources you have access to will allow me to develop appropriate activities and assignments. 3. Prior knowledge and grade level: Could you please clarify the prior knowledge expected from the middle school students and specify their grade level? This information will enable me to tailor the content and complexity of the lesson plan to suit their understanding and abilities. Once I have these details, I'll be able to create a comprehensive and suitable lesson plan on the Apollo 11 mission for your middle school science class.

[This response is parsed into separate questions and shown to the user]

Question 1 (Shown to User): Duration and schedule: Could you please let me know the duration of the lesson plan? Is it designed for a single class period or a series of classes? Additionally, do you have any preferences or specific timing for different activities within the plan?

User Answer: This is for a single one-hour class. Please include at least one short activity in addition to lecture and discussion.

Question 2 (Shown to User): Required resources: Are there any specific resources available for the lesson plan, such as textbooks, videos, or online materials? Knowing what resources you have access to will allow me to develop appropriate activities and assignments.

User Answer: We are going to watch a 10-minute Youtube video about the Apollo program and the moon landing. Their textbook also includes a chapter on space flight.

Question 3 (Shown to User): Prior knowledge and grade level: Could you please clarify the prior knowledge expected from the middle school students and specify their grade level? This information will enable me to tailor the content and complexity of the lesson plan to suit their understanding and abilities.

User Answer: This class has students from multiple grade levels, grades 6-8. We have been covering the history of flight this quarter, and the students have made paper airplanes and launched model rockets in previous classes.

Sent to GPT: Consider the following exchange. Attempt to create the document requested by the user, considering the answers they gave when asked for details.

user: "Write a lesson plan on the Apollo 11 mission for a middle school science class." assistant: "Hello! I'd be happy to help you create a lesson plan on the Apollo 11 mission for a middle school science class. To ensure I fulfill your needs, I have a few questions:

1. Duration and schedule: Could you please let me know the duration of the lesson plan? Is it designed for a single class period or a series of classes? Additionally, do you have any preferences or specific timing for different activities within the plan?"

user: "This is for a single one-hour class. Please include at least one short activity in addition to lecture and discussion." assistant: "

2. Required resources: Are there any specific resources available for the lesson plan, such as textbooks, videos, or online materials? Knowing what resources you have access to will allow me to develop appropriate activities and assignments. "

user: "We are going to watch a 10-minute Youtube video about the Apollo program and the moon landing. Their textbook also includes a chapter on space flight." assistant: "

3. Prior knowledge and grade level: Could you please clarify the prior knowledge expected from the middle school students and specify their grade level? This information will enable me to tailor the content and complexity of the lesson plan to suit their understanding and abilities."

user: "This class has students from multiple grade levels, grades 6-8. We have been covering the history of flight this quarter, and the students have made paper airplanes and launched model rockets in previous classes."

GPT Response (Shown to User): Lesson Plan: Apollo 11 Mission

Grade Level: 6-8 Duration: 1 hour (single class period) Resources:

- 10-minute YouTube video on the Apollo program and the moon landing
- Textbook with a chapter on space flight

Prior Knowledge: Students have been studying the history of flight and have engaged in hands-on activities like making paper airplanes and launching model rockets.

Objectives: By the end of this lesson, students will be able to:

1. Explain the significance of the Apollo 11 mission in the context of space exploration.
2. Describe the steps involved in the Apollo 11 mission.
3. Analyze the impact of the Apollo 11 mission on technology and society.

Procedure:

1. Warm-up activity (10 minutes): - Begin the class by asking students what they know about the Apollo 11 mission. Encourage them to share any facts or prior knowledge they have. This will help activate their prior knowledge and set the stage for the lesson.

2. Introduction to the Apollo 11 mission (15 minutes):

- Show the 10-minute YouTube video on the Apollo program and the moon landing. This video will provide an overview of the mission and the significance of the moon landing. Pause at key points to allow for class discussion and clarification of any questions or misconceptions.

3. Lecture and discussion (20 minutes):

- Using the textbook chapter on space flight, provide a brief lecture on the Apollo 11 mission. Highlight key points such as the crew members, the spacecraft, the

launch, the lunar landing, and the return to Earth. Encourage students to ask questions and engage in discussion throughout the lecture.

4. Hands-on activity: Lunar module design challenge (15 minutes):

- Divide the class into small groups and provide each group with materials such as paper, tape, scissors, and straws.
- Explain that they will be designing and building their own lunar modules that can safely land on the moon. Encourage creativity and problem-solving skills.
- After the groups have completed their designs, invite them to present their lunar modules and explain their design choices to the class.

5. Summary and reflection (5 minutes):

- Ask students to reflect on what they have learned about the Apollo 11 mission and its significance. Allow them to share their thoughts and any new insights they gained during the lesson.
- Provide a brief summary of the key points covered in the lesson, emphasizing the impact of the Apollo 11 mission on technology and society.

Assessment:

- Assessment can be conducted through class participation and engagement during the discussion and hands-on activity.
- Additionally, students can be given a short written assignment or quiz to test their understanding of the key concepts covered in the lesson.

Extension activities:

- If time permits, students can research and present on other significant space missions, exploring the scientific and technological advancements made in those missions.
- Students can also create posters or multimedia presentations highlighting the key events and achievements of the Apollo 11 mission.

Note: If additional time is available, the lesson plan can be extended or revised accordingly.

References

1. D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M.D., Hormozdiari, F., Houlsby, N., Hou, S., Jerfel, G., Karthikesalingam, A., Lucic, M., Ma, Y., McLean, C., Mincu, D., Mitani, A., Montanari, A., Nado, Z., Natarajan, V., Nielson, C., Osborne, T.F., Raman, R., Ramasamy, K., Sayres, R., Schrouff, J., Seneviratne, M., Sequeira, S., Suresh, H., Veitch, V., Vladymyrov, M., Wang, X., Webster, K., Yadlowsky, S., Yun, T., Zhai, X., Sculley, D.: Underspecification presents challenges for credibility in modern machine learning. *J. Mach. Learn. Res.* 23, 226:10237-226:10297 (2022).
2. Kuhn, L., Gal, Y., Farquhar, S.: CLAM: Selective Clarification for Ambiguous Questions with Generative Language Models. *ICML 2023 Workshop on Deployment Challenges for Generative AI.* (2023).
3. Park, J., Lim, S., Lee, J., Park, S., Chang, M., Yu, Y., Choi, S.: CLARA: Classifying and Disambiguating User Commands for Reliable Interactive Robotic Agents,

- <http://arxiv.org/abs/2306.10376>, (2023).
<https://doi.org/10.48550/arXiv.2306.10376>.
4. Fillmore, C.J.: Some problems for case grammar. In: Shuy, R.W. and Fasold, R.W. (eds.) Report Of The Twenty-Second Annual Round Table Meeting on Linguistics and Language Studies. Georgetown Univ. Press, Washington, DC (1973).
 5. Pollard, C., Sag, I.A.: Head-Driven Phrase Structure Grammar. University of Chicago Press (1994).
 6. Valin, R.D.V.: ROLE AND REFERENCE GRAMMAR. Work Papers of the Summer Institute of Linguistics. 37, 12 (1993).
 7. Cui, L., Wu, Y., Liu, J., Yang, S., Zhang, Y.: Template-Based Named Entity Recognition Using BART, <http://arxiv.org/abs/2106.01760>, (2021). <https://doi.org/10.48550/arXiv.2106.01760>.
 8. Krishnan, V., Manning, C.D.: An Effective Two-Stage Model for Exploiting Non-Local Dependencies in Named Entity Recognition. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics. pp. 1121–1128. Association for Computational Linguistics, Sydney, Australia (2006). <https://doi.org/10.3115/1220175.1220316>.
 9. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural Architectures for Named Entity Recognition. Proceedings of NAACL 2016. (2016).
 10. Choi, S., On, K.-W., Heo, Y.-J., Seo, A., Jang, Y., Lee, M., Zhang, B.-T.: DramaQA: Character-Centered Video Story Understanding with Hierarchical QA. Proceedings of the AAAI Conference on Artificial Intelligence. 35, 1166–1174 (2021).
 11. Farrell, R., Robertson, S., Ware, S.G.: Asking Hypothetical Questions About Stories Using QUEST. In: Nack, F. and Gordon, A.S. (eds.) Interactive Storytelling. pp. 136–146. Springer International Publishing, Cham (2016). https://doi.org/10.1007/978-3-319-48279-8_12.
 12. Mueller, E.T.: Story understanding through multi-representation model construction. In: Proceedings of the HLT-NAACL 2003 workshop on Text meaning - Volume 9. pp. 46–53. Association for Computational Linguistics, USA (2003). <https://doi.org/10.3115/1119239.1119246>.
 13. Schwartz, R., Sap, M., Konstas, I., Zilles, L., Choi, Y., Smith, N.A.: Story Cloze Task: UW NLP System. In: Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics. pp. 52–55. Association for Computational Linguistics, Valencia, Spain (2017). <https://doi.org/10.18653/v1/W17-0907>.
 14. Das, S., Giles, C., Sun, G.: Learning Context-free Grammars: Capabilities and Limitations of a Recurrent Neural Network with an External Stack Memory. (1992).
 15. Mikolov, T., Zweig, G.: Context dependent recurrent neural network language model. In: 2012 IEEE Spoken Language Technology Workshop (SLT). pp. 234–239. IEEE, Miami, FL, USA (2012). <https://doi.org/10.1109/SLT.2012.6424228>.
 16. Boughoula, A., San, A., Zhai, C.: Leveraging Book Indexes for Automatic Extraction of Concepts in MOOCs. In: Proceedings of the Seventh ACM Conference on Learning @ Scale. pp. 381–384. Association for Computing Machinery, New York, NY, USA (2020). <https://doi.org/10.1145/3386527.3406749>.

17. Cheng, J., Dong, L., Lapata, M.: Long Short-Term Memory-Networks for Machine Reading, <http://arxiv.org/abs/1601.06733>, (2016). <https://doi.org/10.48550/arXiv.1601.06733>.
18. Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., Wu, Y.: Exploring the Limits of Language Modeling, <http://arxiv.org/abs/1602.02410>, (2016).
19. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language Models are Few-Shot Learners. In: *Advances in Neural Information Processing Systems*. pp. 1877–1901. Curran Associates, Inc. (2020).
20. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving Language Understanding by Generative Pre-Training. (2018).
21. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language Models are Unsupervised Multitask Learners. *OpenAI blog*. 1.8, (2019).
22. Hill, F., Bordes, A., Chopra, S., Weston, J.: The Goldilocks Principle: Reading Children’s Books with Explicit Memory Representations. *arXiv:1511.02301 [cs]*. (2016).
23. Levesque, H., Davis, E., Morgenstern, L.: The Winograd Schema Challenge.
24. Reddy, S., Chen, D., Manning, C.D.: CoQA: A Conversational Question Answering Challenge. *Transactions of the Association for Computational Linguistics*. 7, 249–266 (2019). https://doi.org/10.1162/tacl_a_00266.
25. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., Zhou, D.: Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, <http://arxiv.org/abs/2201.11903>, (2023). <https://doi.org/10.48550/arXiv.2201.11903>.
26. Lenat, D.: Not Good As Gold: Today’s AI’s Are Dangerously Lacking In AU (Artificial Understanding), <https://www.forbes.com/sites/cognitive-world/2019/02/18/not-good-as-gold-todays-ais-are-dangerously-lacking-in-au-artificial-understanding/>, last accessed 2022/12/05.
27. Lenat, D.: Getting from Generative AI to Trustworthy AI: What LLMs might learn from Cyc. (2023).
28. Moore, J.M., Shipman, F.M.: A comparison of questionnaire-based and GUI-based requirements gathering. In: *Proceedings ASE 2000. Fifteenth IEEE International Conference on Automated Software Engineering*. pp. 35–43 (2000). <https://doi.org/10.1109/ASE.2000.873648>.
29. Pandey, D., Suman, U., Ramani, A.K.: An Effective Requirement Engineering Process Model for Software Development and Requirements Management. In: *2010 International Conference on Advances in Recent Technologies in Communication and Computing*. pp. 287–291. IEEE, Kottayam, India (2010). <https://doi.org/10.1109/ARTCom.2010.24>.
30. Tabalba, R., Kirshenbaum, N., Leigh, J., Bhattacharya, A., Johnson, A., Grosso, V., Di Eugenio, B., Zellner, M.: Articulate+ : An Always-Listening Natural Language Interface for Creating Data Visualizations. In: *Proceedings of the 4th Conference on*

- Conversational User Interfaces. pp. 1–6. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3543829.3544534>.
31. Pyatkin, V., Hwang, J.D., Srikumar, V., Lu, X., Jiang, L., Choi, Y., Bhagavatula, C.: ClarifyDelphi: Reinforced Clarification Questions with Defeasibility Rewards for Social and Moral Situations, <http://arxiv.org/abs/2212.10409>, (2023). <https://doi.org/10.48550/arXiv.2212.10409>.
 32. Zhang, S., Pan, L., Zhao, J., Wang, W.Y.: Mitigating Language Model Hallucination with Interactive Question-Knowledge Alignment, <http://arxiv.org/abs/2305.13669>, (2023). <https://doi.org/10.48550/arXiv.2305.13669>.
 33. Mu, F., Shi, L., Wang, S., Yu, Z., Zhang, B., Wang, C., Liu, S., Wang, Q.: ClarifyGPT: Empowering LLM-based Code Generation with Intention Clarification, <http://arxiv.org/abs/2310.10996>, (2023).
 34. Ge, Y., Xiao, Z., Diesner, J., Ji, H., Karahalios, K., Sundaram, H.: What should I Ask: A Knowledge-driven Approach for Follow-up Questions Generation in Conversational Surveys, <http://arxiv.org/abs/2205.10977>, (2023). <https://doi.org/10.48550/arXiv.2205.10977>.
 35. Papineni, K., Roukos, S., Ward, T., Zhu, W.-J.: Bleu: a Method for Automatic Evaluation of Machine Translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. pp. 311–318. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA (2002). <https://doi.org/10.3115/1073083.1073135>.
 36. Zhang*, T., Kishore*, V., Wu*, F., Weinberger, K.Q., Artzi, Y.: BERTScore: Evaluating Text Generation with BERT. Presented at the International Conference on Learning Representations September 25 (2019).
 37. Rogers, A., Gardner, M., Augenstein, I.: QA Dataset Explosion: A Taxonomy of NLP Resources for Question Answering and Reading Comprehension. *ACM Comput. Surv.* 55, 1–45 (2023). <https://doi.org/10.1145/3560260>.
 38. Lin, S., Hilton, J., Evans, O.: TruthfulQA: Measuring How Models Mimic Human Falsehoods, <http://arxiv.org/abs/2109.07958>, (2022). <https://doi.org/10.48550/arXiv.2109.07958>.
 39. Wang, B., Zhu, Y., Chen, L., Liu, J., Sun, L., Childs, P.: A study of the evaluation metrics for generative images containing combinational creativity. *AIEDAM.* 37, e11 (2023). <https://doi.org/10.1017/S0890060423000069>.
 40. Yang, L.-C., Lerch, A.: On the evaluation of generative models in music. *Neural Comput & Applic.* 32, 4773–4784 (2020). <https://doi.org/10.1007/s00521-018-3849-7>.
 41. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-Shot Text-to-Image Generation. In: Proceedings of the 38th International Conference on Machine Learning. pp. 8821–8831. PMLR (2021).
 42. Webson, A., Loo, A.M., Yu, Q., Pavlick, E.: Are Language Models Worse than Humans at Following Prompts? It’s Complicated, <http://arxiv.org/abs/2301.07085>, (2023). <https://doi.org/10.48550/arXiv.2301.07085>.