Ambiguity Resolution in Conversational Question Answering through Minimal Question Identification

Stanford CS224N Custom Project

Sahil Kulkarni
Department of Computer Science
Stanford University
sahil1@stanford.edu

Abstract

In this project, we propose a method for resolving ambiguity in conversational question answering by encouraging language models to identify and ask clarifying questions. We present the Minimal Question Identification (MQI) method, an iterative approach that encourages language models to generate clarifying questions through assessment and fine-tuning phases. Our work highlights the potential of the MQI method for improving language models' abilities to handle ambiguous settings and answer questions under incomplete information. We show that our method achieves higher accuracy and question quality than two baseline models. We also set the stage for next steps to assess the validity of our method in more difficult settings.

1 Key Information to include

- Mentor: Siyan Li (siyanli@stanford.edu)
- External Collaborators (if you have any): Satchel Grant (grantsrb@stanford.edu)
- Sharing project: No

2 Introduction

With the proliferation of large language models (LLMs) being deployed in user-facing systems, it is crucial we ensure they respond reliably and accurately to general purpose queries. One solution to this problem may be to encourage LLMs to identify when there is insufficient information to answer a question, and subsequently ask clarifying questions to resolve the ambiguity – a task known as Clarification-based Question Answering (CbQA) (Guo et al., 2021).

Despite the importance of ambiguity resolution, clarification question generation appears to be an under explored area within the literature (Guo et al., 2021). Furthermore, existing approaches to CbQA suffer from limitations such as insufficient adaptability to diverse queries (Yuan et al., 2019), high computational cost, and over dependence on human supervision (Nakano et al., 2022).

In pursuit of this goal, we propose a method, inspired by the success of recent "Iterated Learning" methods, particularly STaR (Zelikman et al., 2022) and Selection-Inference (Creswell et al., 2022), which we call Minimal Question Identification (MQI). Our method leverages the strengths of in-context learning and fine-tuning to encourage language models to ask clarifying question in settings with incomplete information. We prompt a pre-trained LLM, S, which we call the student model, to perform the CbQA task and fine-tune it on its own examples which result in successful ambiguity resolution. To minimize the need of human intervention we utilize another pre-trained LLM T, which we denote the teacher model, to guide S in this task. We show that using this method results in considerably higher accuracy for CbQA on the bAbI 15 task (Jason et al., 2015) than baseline models, however, find key limitations in its ability to generalize to unseen domains.

3 Related Work

Previous work have proposed wide ranging approaches to ambiguity resolution. The CbQA task is closely related to Interactive Question Answering (IQA) (Yuan et al., 2019), where a model engages in a dialogue with a user to refine the user's query and ultimately provide the desired information. For example, Yuan et al. (2019) propose using a reinforcement learning agent to interact with a text-based environment to answer ambiguous questions about objects. Such methods, however have been shown to fail to generalize to more general queries.

Recently, the "Iterated Learning" framework has gained attention as a means to effectively train language models through clever prompting strategies which encourage the generation of explicit thought traces. In particular, STaR (Zelikman et al., 2022) and Selection-Inference (Creswell et al., 2022) have demonstrated success in leveraging this framework for various reasoning tasks. Our MQI method draws inspiration from these works but specifically targets the CbQA task.

4 Approach

4.1 Clarification-based Question Answering

In this section, we describe the clarification-based question answering (CbQA) task, which aims to identify the minimal number of clarifying questions needed to answer a question with incomplete information. We are given an initial dataset of D context-question-answer triples, denoted as $\{(C^i, q^i, a^i)\}_{i=1}^D$. Each context C^i consists of n_i logical statements, $C^i = \{s_1^i, \ldots, s_{n_i}^i\}$, that provide complete information to answer the corresponding question q^i. We assume that the answer a^i is a single word and under these constraints, $p_S(a^i \mid s_1^i, \ldots, s_{n_i}^i, q^i) = 1$.

To create a scenario with incomplete information, we preprocess our data by removing k_i essential statements from C^i to form the context $C_0^i = \{s_1^i, \ldots, s_{n_i-k_i}^i\}$. As a result, C_0^i no longer contains the complete information required to answer q_i.

Then, CbQA is the task for a student model S to ask k_i clarifying questions to a respondent in order to elicit the withheld information, and thereby answer the initial question q_i correctly. More precisely, if we convert the withheld statements into question form, $Q^i = \{q_{c_j}^i\}_{j=1}^{k_i}$, the task is for the student model S receiving (C_0^i, q^i) as input, to produce Q^i and a_i through feedback from a respondent T.

We use a fine-tuned language model for T, which we denote the teacher model. To promote the desired behavior, we manually create four examples and include them in an initial prompt, denoted as $\mathcal{P}$. Therefore, an oracle would generate the following dialogue: $(\mathcal{P}, C_0^i, q^i, q_{c_1}^i, s_{n_i-k_i+1}^i, q_{c_2}^i, s_{n_i-k_i+2}^i, \ldots, q_{c_{k_i}}^i, s_{n_i}^i, a^i)$.

4.2 Teacher Model

To automate the responses to the student, we fine-tune a teacher model T on the entire dataset where each example takes the form $((C^i, q_{c_j}^i), s_j^i)$. Then, at inference time T receives $(C^i, S(\cdot))$ as input, so it has access to the complete information. Note that its fine-tuning examples do not account for the potentially extraneous questions/responses that S may generate. For now, we assume that the zero-shot abilities of T are sufficient to handle such cases, but seek to address this in future work. We also note that T need not be a language model, rather in more general settings one may be able to use human respondents to provide feedback. We use the pre-trained GPT-2 medium model (Radford et al., 2019) for the teacher model, as the expected responses follow a simple pattern for the datasets we use. For future work, we would like to investigate the use of a larger model with stronger few-shot capabilities to handle more complex settings.

4.3 Minimal Question Identification (MQI) Method

We now describe the MQI method, a procedure akin to (Zelikman et al., 2022) intended to encourage S to ask clarifying questions without an explicit in-context prompt. First, given a training set $\mathcal{D} = \{((\mathcal{P}, C_0^i, q^i), a^i)\}_{i=1}^D$, we perform the CbQA task for each example in $\mathcal{D}$. Particularly, for $(x_i, y_i) \in \mathcal{D}$ we evaluate the student and teacher model in succession, until either S provides the correct answer (determining this is trivial as a^i is a single word), or a maximum number of

prespecified turns $t > \max_i k_i + 1$ is reached. If the evaluation results in a correct answer, we remove (x_i, y_i) from $\mathcal{D}$ and we store the entire generated dialogue in a set $\mathcal{X}$ to use for fine-tuning. We then repeat this process until the set of correctly answered questions does not change or we reach a prespecified maximum number of trial rounds r. We call this procedure the assessment phase.

Next, we fine-tune S on the correctly answered questions in $\mathcal{X}$. Note that the generated intermediate dialogues may not correspond to the oracle's. We then perform assessment again on the samples the student model failed to answer in the previous assessment phase. We repeat this assessment-fine-tuning process until the set of correctly answered questions does not change or we reach a prespecified maximum number of assessment rounds A, similar to Rationale Generation Bootstrapping in (Zelikman et al., 2022).

We are then left with a subset of examples in $\mathcal{D}$ for which the student model is unable to successfully elicit the required information and answer the question. To improve performance on these examples we provide the model with a hint, similar to Rationalization in (Zelikman et al., 2022), by appending a ground truth clarifying question-answer pair to a subset of the remaining inputs, i.e. $x'_i = (\mathcal{P}, C^i_0, q^i, q^i_{c_j}, s^i_j)$, and repeat the former procedure until S successfully answers the question or we reach the maximum rounds. Note that the choice of hint may depend on the dataset.

The general framework of Minimal Question Identification is largely equivalent to the STaR framework as mentioned above. However, we note that MQI is oriented toward information elicitation rather than explicit reasoning based on complete information. As a result, it is designed to handle multi-turn dependencies. To our knowledge, this component is a novel contribution, and is inspired by the success of the iterative rationale generation of (Creswell et al., 2022).

Algorithm 1 AssessFineTune$(\mathcal{D}, \mathcal{C}, A, r, t)$

$\mathcal{X} \leftarrow \emptyset$
for assessment $= 1, \ldots, A$ **do**
  if $|\mathcal{X}|$ unchanged **then**
    break
  end if
  for trial $= 1, \ldots, r$ **do**
    for $(x_i, y_i) \in \mathcal{D}$ **do**
      $v \leftarrow x_i$
      for turn $= 1, \ldots, t$ **do**
        $v \leftarrow (v, S(v))$
        if $S(v) == y_i$ **then**
          $\mathcal{D} \leftarrow \mathcal{D} \setminus \{(x_i, y_i)\}$
          $\mathcal{X} \leftarrow \mathcal{X} \cup \{v\}$
          break
        end if
        $\# \mathcal{C} = \{C^i\}_{i=1}^D$
        $v \leftarrow (v, T((C^i, S(v))))$
      end for
    end for
    if $|\mathcal{X}|$ unchanged **then**
      break
    end if
  end for
  $S \leftarrow \text{FineTune}(S, \mathcal{X})$
end for

Algorithm 2 Hint$(\mathcal{H}, \mathcal{D})$

$\# \mathcal{H} = \{(\mathcal{P}, C^i_0, q^i, q^i_{c_j}, s^i_j)\}_{i=1}^D$
$d \subseteq \mathcal{D}$
for $(x_i, y_i) \in \mathcal{D}$ **do**
  if $(x_i, y_i) \in d$ **then**
    $x_i \leftarrow x'_i \in \mathcal{H}$
  end if
end for

Figure 1: **MQI Method**: Algorithm 1 outlines the assessment-fine-tuning process. Here $S(v)$ denotes the completion given the updated context. Algorithm 2 outlines the hint procedure. Here we update a subset of the remaining inputs to contain a hint. Then, more generally, the MQI method is the procedure of repeatedly performing the two algorithms in succession. For the purpose of our initial explorations we performed only one iteration of each, however, we imagine that for more complex tasks one may require multiple.

mice are afraid of cats.
sheep are afraid of cats.
Emily is a mouse.
wolves are afraid of mice.
Gertrude is a cat.
cats are afraid of sheep.
Jessica is a sheep.
Winona is a wolf.
Q: What is Jessica afraid of?
A: cats

mice are afraid of cats.
Emily is a mouse.
wolves are afraid of mice.
Gertrude is a cat.
cats are afraid of sheep.
Jessica is a sheep.
Winona is a wolf.
QC1: What are sheep afraid of?
S1: sheep are afraid of cats.
A: cats

mice are afraid of cats.
sheep are afraid of cats.
Emily is a mouse.
wolves are afraid of mice.
Gertrude is a cat.
cats are afraid of sheep.
Winona is a wolf.
Q: What is Jessica afraid of?
QC1: What kind of animal is Jessica?
S1: Jessica is a sheep.
A: cats

mice are afraid of cats.
Emily is a mouse.
wolves are afraid of mice.
Gertrude is a cat.
cats are afraid of sheep.
Winona is a wolf.
Q: What is Jessica afraid of?
QC1: What kind of animal is Jessica?
S1: Jessica is a sheep.
QC2: What are sheep afraid of?
S2: sheep are afraid of cats.
A: cats

Figure 2: Oracle dialogues created from task 15 of the bAbI dataset, for various examples of k_i. From left to right: $k_i = 0$, $k_i = 1$, $k_i = 1$, $k_i = 2$. The ground-truth clarifying question format using the rule-based method is shown. Withheld statements are directly used to create the ground-truth teacher response.

4.4 Baselines

To assess the overall improvement in performance the MQI method provides on our task, we take GPT-J (Wang, 2021) without MQI as one of our baseline models. This provides a reference performance measure of GPT-J's few-shot abilites on our tasks. We also use GPT-J using MQI without hints as a baseline model. This model is the checkpoint we receive after running the initial assessement-fine-tuning but before the hint procedure. Therefore, this allows us to assess the improvement in performance we receive through providing stronger supervision on a subset of the data.

5 Experiments

5.1 Data

5.1.1 Basic Deduction

To test the validity of our approach, we use task 15 of the bAbI dataset (Jason et al., 2015), which are basic deduction questions. We modify the dataset to align with the CbQA task format by performing the steps described in the previous section. In doing so, for each sample we randomly set $k_i \in \{0, 1, 2\}$. The subsequent conversion from statement to question is done using a simple rule-based method. The resulting dataset contains 1000 training samples and 500 test samples. An example is shown in Figure 2.

5.1.2 Out-of-Domain Evaluation

It is also a key interest to assess the efficacy of the MQI method in facilitating model generalization to out-of-domain questions under incomplete information. To evaluate this, we perform a qualitative analysis on part 1 of the PuzzTE dataset (Szomiu and Groza, 2021). This dataset contains 250 ambiguous height-related logical puzzles with varying levels of ambiguity.

5.2 Evaluation method

5.2.1 Model Evaluations

We evaluate each of three student models, specifically the base GPT-J, GPT-J with Minimal Question Identification without hints (GPT-J-MQI-NH), and GPT-J-MQI with hints, on the basic deduction task. To assess the capacity of the models to perform the CbQA tasks using their internal knowledge, rather than relying on the pattern in the prompt, we evaluate each model both with and without the initial prompt $\mathcal{P}$.

For the out-of-distribution assessment on PuzzTE, we analyze the qualitative differences in responses between GPT-J-MQI and the standard GPT-J. For the purpose of this analysis, we do not use an initial

prompt that encourages question generation for either model. However, we would like to explore the effect this has in future work.

5.2.2 Evaluation Metrics

To ensure that T provides valid responses to clarifying questions, we evaluate it on the entire dataset by computing the average Jaccard Similarity between the generated response $\hat{s}^i_j$, given $(C^i, q^i_{c_j})$, and the ground truth response s^i_j – which we denote AvgJS(T).

We would like to evaluate the ability of S to produce the correct answer to the initial question. For the purpose of our initial exploration we enforce the answer to each question to be a single word. Therefore, we simply use test set accuracy as our evaluation metric.

To evaluate the quality of the questions generated by S, we propose the following metric. For each test example, we compute the maximum Jaccard Similarity between each clarifying question generated by S and the set of ground truth clarifying questions. We then compute the mean of these scores over the entire test set – which we denote AvgMaxJS(S). This provides us with a sense for how aligned the questions generated by S are with the expected clarifying questions, on average.

Finally, to evaluate the ability of the model to generate the minimal number of clarifying questions we compute the average absolute deviation between the number of questions posed by S, and the ground truth number of clarifying questions k_i – which we denote AvgAD(S).

5.3 Experimental details

We use the provided script in (Jason et al., 2015) to generate the bAbI 15 dataset. We then use our own script to remove and generate valid context-question-statement triples which are used to fine-tune T and generate the train/test set. We use scripts provided in (Wang, 2021) to fine-tune the models. Our implementation of MQI is largely done independently, but adapts from (Zelikman et al., 2022) for evaluation. Due to memory constraints we run batch inference over the entire datasets for the student and teacher model separately, rather than in sequence as outlined in the algorithm. For MQI we use 2 maximum assessment rounds, 5 maximum trial rounds, and 7 maximum turns. Following (Zelikman et al., 2022) we use 10^{-6} as our learning rate, a batch size of 8, and 1 epoch for fine-tuning.

5.4 Results

5.4.1 Basic Deduction

Model	Accuracy	AvgMaxJS(S)	AvgAD(S)
GPT-J	17%	-	-
GPT-J-MQI-NH	83%	88%	0.14
GPT-J-MQI	92%	97%	0.05
GPT-J (given $\mathcal{P}$)	56%	67%	0.83
GPT-J-MQI-NH (given $\mathcal{P}$)	88%	91%	0.08
GPT-J-MQI (given $\mathcal{P}$)	**94%**	**98%**	**0.02**

Table 1: Evaluation metrics for CbQA on bAbI 15. AvgMaxJS(S) and AvdAD(S) are not reported for the standard GPT-J model, as it is not able to generate the response structure we use to evaluate the metrics.

We find that AvgJS(T) is $\sim 99\%$ which is expected as we fine-tune T on these examples and the expected string is present in the context. Next, from the results in Table 1 we find that using the in-context prompt $\mathcal{P}$ significantly improves performance on the task. Particularly, without the prompt the standard GPT-J model is only able to correctly answer in some cases for when $k_i = 0$, as there is complete information. With the prompt, however, we find that it is able to produce the desired structure of responses in some cases. We also note that the prompt improves the performance of the models using MQI. This is somewhat expected as the prompt encourages the desired response structure, which leads to correct answers more often.

We find that the model using the full MQI procedure results in the highest performance on the CbQA task across all metrics. The full MQI model evaluated with prompts achieves 38% higher accuracy than the corresponding the baseline GPT-J model, and 6% higher accuracy than the baseline MQI model without hints. We also find that without the prompt, the full MQI model achieves 9% higher accuracy than the MQI model without hints. These results are somewhat expected as 1. this model receives the most amount of supervision, 2. the number of withheld statements is relatively low (≤ 2), and 3. the bAbI data are simple toy questions that largely follow the same pattern, which makes the test/training sets similar.

6 Analysis

6.1 Basic Deduction

The simple structure of the bAbI 15 task makes the prompt a strong signal boosting inital performance. As a result, we suspect that the relatively high performance of the standard GPT-J model may be attributed to its ability to simply use the relevant line in the prompt corresponding to the correct completion to generate clarifying questions, and subsequently guessing an entity (e.g. mice, wolves, Emily) for the final answer – rather than reasoning about the given problem. On the other hand, models using the MQI method appear to consistently be able to generate the correct clarifying questions and entity, with significantly fewer instances of names being produced in the final answer, as demonstrated through the evaluation metrics. This leads us to believe that the fine-tuning process encourages the model to understand the task, resulting in better performance during evaluation.

To understand the qualitative differences between the models using MQI with and without hints, we examine the examples that the model without hints fails to solve. In these cases, we often find that while the model is able to generate the correct clarifying questions, it predicts the answer to be the entity that most frequently appears in the statements describing the scared relationships – as seen in Figure 3. This could suggest that the model using hints might be relying more on the understanding of the problem, rather than certain statistical patterns in the data.

Another limitation of our approach lies in the tendency for the teacher model to respond inaccurately. For example, in some cases we find that the diversity of clarifying questions generated by the student models, particularly in cases without the initial prompt, leads to the teacher model producing incorrect responses. We provide an example in Figure 4, in which the ground-truth withheld statement is "mice are afraid of cats.".

```
sheep are afraid of mice.
Gertrude is a sheep.
Winona is a mouse.
Emily is a wolf.
mice are afraid of cats.
cats are afraid of mice.
wolves are afraid of mice.
Q: What is Jessica afraid of?
QC1: What kind of animal is Jessica?
S1: Jessica is a mouse.
A: mice
```

Figure 3: An example of an incorrect answer provided by GPT-J-MQI-NH

```
wolves are afraid of cats.
Winona is a cat.
cats are afraid of wolves.
sheep are afraid of wolves.
Jessica is a wolf.
Gertrude is a mouse.
Emily is a sheep.
Q: What is Gertrude afraid of?
QC1: What are the common phobias of mice?
S1: mice are afraid of sheep.
A: sheep
```

Figure 4: An example illustrating the diversity of questions in S leading to incorrect completions by T.

6.2 Out-of-Domain Qualitative Evaluation

We would like models that have used the MQI method to ask clarifying questions in unseen settings with incomplete information. We qualitatively compared the responses between the full MQI model and the standard GPT-J model on a handful of ambiguous examples from the PuzzTE dataset. Unfortunately, we did not observe any noticeable differences in response quality or incidences of

question generation. We observed that both models appeared to confidently provide incorrect answers to ambiguous questions. We seek to conduct a more comprehensive analysis of the generalization ability of MQI models in future work by assigning confidence scores to model outputs using sentiment analysis, and comparing aggregate statistics between various models.

7 Conclusion

Our main findings demonstrate that the MQI method significantly improves the performance of models on the CbQA task. Specifically, the full MQI model achieved the highest accuracy across all evaluation metrics, outperforming the baseline GPT-J and MQI model without hints. This highlights the effectiveness of our method in encouraging large language models to ask clarifying questions and resolve ambiguity.

However, we observe some key limitations to our work. The basic deduction task we employed has a simple structure, which may have contributed to the strong performance of the models. Furthermore, the teacher model's fine-tuning examples do not account for potentially extraneous questions/responses that the student model may generate, which in some cases leads to it producing inaccurate responses – resulting in the student model in failing to correctly answer the initial question.

For future work, we propose exploring more complex datasets and using a larger teacher model with stronger few-shot capabilities to handle more challenging settings. Additionally, investigating the effects of using human respondents to provide feedback to the student model and incorporating more sophisticated evaluation metrics could be beneficial. Overall, our work provides a solid foundation for further research on using minimal question identification to improve clarification-based question answering.

References

Antonia Creswell, Irina Higgins, and Murray Shanahan. 2022. Selection-inference: Exploiting large language models for interpretable logical reasoning.

Meiqi Guo, Mingda Zhang, Siva Reddy, and Malihe Alikhani. 2021. Abg-coqa: Clarifying ambiguity in conversational question answering. In *Automated Knowledge Base Construction (AKBC)*.

Weston Jason, Antoine Bordes, Sumit Chopra, Rush. Alexander M., Joulin Armand Merriënboer, Bart van and, and Tomas Mikolov. 2015. Towards ai-complete question answering: A set of prerequisite toy tasks.

Yuya Nakano, Seiya Kawano, Koichiro Yoshino, Katsuhito Sudoh, and Satoshi Nakamura. 2022. Pseudo ambiguous and clarifying questions based on sentence structures toward clarifying question answering system. In *Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering*, pages 31–40, Dublin, Ireland. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Roxana Szomiu and Adrian Groza. 2021. A puzzle-based dataset for natural language inference.

Ben Wang. 2021. Mesh-transformer-jax: Model-parallel implementation of transformer language model with jax.

Xingdi Yuan, Marc-Alexandre Cote, Jie Fu, Zhouhan Lin, Christopher Pal, Yoshua Bengio, and Adam Trischler. 2019. Interactive language learning by question answering.

Eric Zelikman, Yuhai Wu, Jesse Mu, and Noah Goodman. 2022. Star: Bootstrapping reasoning with reasoning. In *NeurIPS*.