A Teacher-Student Approach to Creating Verified Synthetic Clarification and Correction Dialogues for TableQA Tasks

Christian Poelitz, Nick McKenna
Microsoft Research, United Kingdom
cpoelitz@microsoft.com, nmckenna@microsoft.com*

Abstract

Real dialogues with AI assistants for solving table questions-answering tasks often follow dynamic, unpredictable paths due to imperfect information provided by the user or in the data, which must be caught and handled. Developing datasets which capture such user-AI interactions is difficult and time-consuming. In this work, we develop a novel framework for synthetically generating controlled, multi-turn conversations between a user and AI assistant for the task of table-based question answering (TableQA), which can be generated from an existing dataset with fully specified TableQA examples for any target domain. Each conversation aims to solve a table-based reasoning question through collaborative effort, modeling one of two real-world scenarios: (1) an AI-initiated clarification, or (2) a user-initiated correction. Critically, we employ a strong teacher LLM to verify our synthetic conversations by functional correctness, ensuring high quality. Finally, we demonstrate synthetic datasets generated from TableQA tasks as benchmarks of frontier LLMs. We find that even larger models struggle to effectively issue clarification questions and accurately integrate user feedback for corrections, demonstrating important areas for future research.

Keywords: Evaluation Methodologies, Question Answering, Language Modeling

1. Introduction

AI assistants are effective for solving data-centric tasks, in particular table-based question-answering (TableQA) (Li et al., 2021; Nararatwong et al., 2022), with numerical reasoning frequently addressed by for example generating code that operates on the table (Deng et al., 2022; Chen et al., 2021). Yet, solving problems in dialogue with AI assistants is often more complex than single-turn requests. These can be *multi-turn* conversations with an unpredictable trajectory, especially if the task depends on incomplete information both from a user and the data context to achieve a solution. Identifying missing information and recovering from any errors that arise is critical. To measure and improve the performance on such conversations we typically rely on available conversational datasets with labeled dialog acts of clarifications and corrections. However, creating this dialogue data is often a laborious and manual task, making it difficult to produce for a target domain.

To address these gaps, we propose a novel framework with a **teacher-student** method for synthetically generating controlled, multi-turn conversations between a user and an AI assistant for TableQA tasks, involving clarifications and corrections. Our approach leverages existing TableQA datasets to create synthetic yet realistic dialogues that model real-world interactions. Each conversation is designed to solve a table-based reasoning question through collaborative effort, modeling one of two common scenarios:

1. **AI-initiated clarification**: The assistant identifies ambiguous or incomplete information in the user's *question* or *data* and proactively asks for clarification to answer the question.
2. **User-initiated correction**: The assistant provides an initial answer that may be incorrect due to missing information and a user provides the missing information as a correction.

In our approach, a teacher model collaborates with a student model (acting as the AI assistant) to generate synthetic multi-turn conversations. These dialogues, which incorporate both clarifications and corrections, form a dataset for benchmarking (and training) the student model. Using these dialogues, we test how well the students are able to generate clarification questions without knowing whether the task can be solved with the available information, as well as how effectively they can leverage user feedback to correct a wrong answer.

Our synthetic benchmark is designed with solvability as a core requirement. In every dialogue, we ensure that (i) the student model is capable, in principle, of asking the right clarification question, meaning tasks are never impossible; (ii) when the student poses a clarification question, there is a clear pathway to correct a wrong answer; and (iii) even if the student does not ask for clarification, user intervention with necessary corrections can guide the model to the correct answer.

We make the following contributions:

1. We present a novel teacher-student framework for generating realistic conversational data from arbitrary, existing TableQA tasks which are verifiable for quality. We offer a solution to the unpredictability and manual creation

*Now at GitHub Applied Science

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 4199–4212
11–16 May 2026.

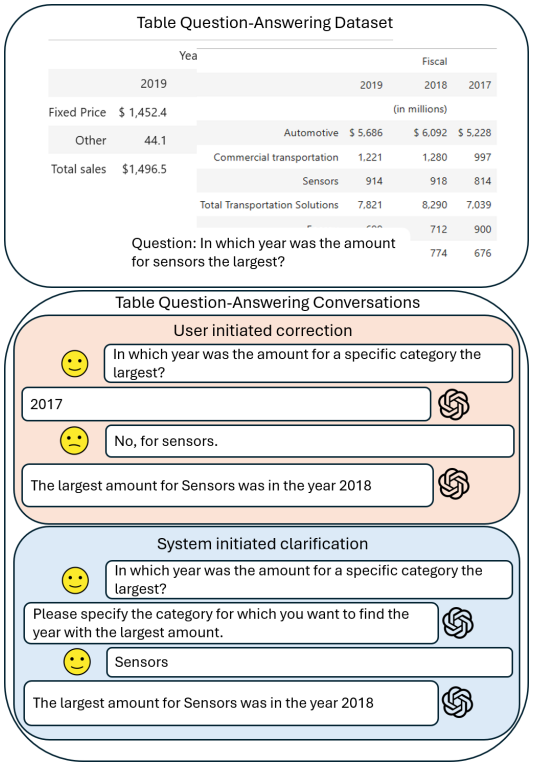

Figure 1: Illustration of synthetic corrections and clarifications based for the table question: In which year was the amount for sensors the largest?. The information about the field Sensors is ablated. Top: Illustration with both question and table. Middle: User-initiated correction. Bottom: AI-initiated clarification.

issues of current datasets, as well as for evaluating models on these tasks.

2. We establish baselines for several large language models (LLMs) on multi-turn TableQA benchmarks with missing information. In the scenario of AI-initiated clarification, we observe that even strong LLMs like GPT4-Turbo struggle. In the scenario of user-initiated correction, we find that on average the models perform better when they are directly corrected by a user rather than when they ask for clarification.

3. We demonstrate the effectiveness of using additional synthetic conversations as training data, resulting in significant improvements in model performance, increasing the accuracy of smaller LLMs like Llama3.1 8b and Qwen2.5 7b on clarifications.

2. Background and related work

Our approach for synthetically generating conversations with clarifications and corrections in TableQA intersects with several research areas, ranging from generating synthetic training data, synthetic benchmarks for LLMs and synthetic data for task specific dialogues.

Synthetic data generation Early approaches to generate synthetic data for training originate from traditional instruction tuning data (Longpre et al., 2023) and include approaches like self-instruct to generate instructions using LLMs (Wang et al., 2023) or LLM-based augmentations with rewrites (Xu et al., 2023). More recently, task-specific synthetic training data using domain ontologies have emerged (Sudalairaj et al., 2024; Li et al., 2024). For a more general overview on recent approaches for synthetic training data, see (Long et al., 2024; Liu et al., 2024).

Besides using LLMs to synthetically generate training data, recently more and more benchmarks for evaluations and automatic evaluation methods leverage LLMs to generate data and evaluate model responses. Previous benchmarks (Zheng et al., 2023; Dubois et al., 2024; Lin et al., 2024a; Gu et al., 2024) mainly leverage synthetic data and LLMs for evaluations. Such automatic evaluation methods are based on using LLMs as a judge, respectively auto-annotator. While several works on LLMs-as-a-judge showed that open-ended quality criteria show a high correlation between LLM-based judgments and human judgments (Murugadoss et al., 2024), works on correctness criteria and verification show the dependence on LLMs with high competence in the corresponding domain (Lin et al., 2024b). Model hallucination is always a risk in synthetic data, especially data for evaluation use cases, so we aim to solve this problem categorically using program-aided verification.

Dialogue systems The majority of the above-mentioned approaches are single turn approaches, synthetically generating a single user-AI interaction. Prior work on generating conversational data and dialogue acts with clarifications (Deng et al., 2023b; Rahmani et al., 2023; Aliannejadi et al., 2021b; Budzianowski et al., 2020) often leverage human annotation with crowdsourcing making data generation expensive.

More recent approaches leverage LLMs to synthesize conversations grounded in knowledge sources (Bao et al., 2023) or hand-crafted templates (Kulkarni et al., 2024) for example. Generating conversations with clarification has been explored before in conversational recommendation (Ren et al., 2021), information retrieval (Zamani et al., 2020) , question answering (Guo et al., 2021) or for open-domain questions by LLMs (Zhang et al., 2025). Our approach is a scalable and automatic way to generate controlled conversations with clarifications and corrections.

General data-centric tasks Related work on data-centric dialogues focus often on clarification of the user queries and regard the data as fixed (Chen et al., 2024; Deng et al., 2022; Aliannejadi et al., 2021a; Wu et al., 2023; Guo et al., 2021), we take a novel approach and assume the data is owned by the user, and can be collaboratively improved with the help of an agent to meet a user's information need. Further, in this work we generate entire dialogues from existing target domain datasets, alleviating the burden of manually collecting conversations as in prior work like (Deng et al., 2022).

In contrast to previous approaches, we propose an automatic evaluation framework which reformulates existing, non-conversational TableQA benchmarks, concentrating on crucial dialogues around corrections and clarifications. Our approach synthetically generates both evaluation and training data, and can be easily applied to any LLM to measure the quality on generating clarification questions as well as correcting response.

3. Framework

Our framework consists of two key components: 1) **a teacher-student method** to generate a benchmark for a student model (an LLM acting as TableQA assistant) with clarification questions and corrections verified to lead to correction of a wrong answer. We verify that the generated clarification questions by the student are useful in correcting wrong solutions and that the student can recover from its initial error to come to the final correct answer. 2) **an evaluation method** to evaluate the student models on its ability to generate clarification questions and fix incorrect solutions on the synthetically generated dialogues in the benchmarks. We filter synthetic questions based on these components to produce a quality benchmark.

3.1. Teacher-student method

For the benchmark, we synthetically generate realistic human-AI dialogues for TableQA tasks with missing information and verified correctable solutions by either, clarifications initiated by the AI assistant, or corrections initiated by the user (see Fig. 2 for an illustration). In contrast to previous approaches like (Chen et al., 2024; Deng et al., 2022), we verify that the student model can come to a correct final answer after a clarification question and a correction, making sure we do not create unsolvable tasks. Further, in contrast to (Deng et al., 2022; Chen et al., 2024) our approach creates conversations with clarifications and corrections on task description and data, ablating necessary information from both. This is our generation pipeline:

1. Seed Question Solvability Starting from an original TableQA benchmark dataset Or, for each initial TableQA task $T = (q, t)$ for a question q about a table t represented as columns $[c_1, \cdots, c_m]$ and each column c_i consisting of values $[v_{1i}, \cdots, v_{ki}]$ and a ground-truth answer a^*, the teacher model checks whether the student model can solve the task by comparing the answer to the ground-truth a^*. If the student model can solve the task we can proceed with the next steps to perform controlled ablations. Otherwise, sample error would not be attributable to our ablations, thus we discard unsolvable samples.

The teacher then applies novel **ablation strategies** removing information I from either the table t or the question q, noted as T_{-I}. We then verify that ablating this information makes the task *unsolvable* by the student model. Adding this information back in a follow-up clarification (initiated by the student model) or correction (initiated by a user), the task becomes solvable again for the student.

2. Ablation strategies To reformulate the original TableQA tasks in such a way that a student model can only solve them by asking clarification questions and incorporating user given corrections, we remove necessary information making the tasks not directly solvable.

Removing necessary information from the tables involves deleting either whole columns i.e. $I = c_i$, or values $I = v_{i,j}$ from the table which is needed to give the correct answer. Removing necessary information from the question is done by rephrasing the original question q to q' such that the question is either ambiguous or not answerable due to missing information, hence $I = \textit{diff}(q, q')$, the semantic difference between the two questions. In both cases, we use the teacher model to perform the ablation.

If the removal of the necessary information makes the student model fail to produce the correct answer, producing a wrong answer a_{wrong}, we add the new task T_{-I} to a candidate set C of potentially clarifiable (3A) and correctable (3B) questions under the student model, otherwise we discard it.

3A. Clarification After ablating the information, the teacher instructs the student model to generate clarification questions to help answer the unsolvable tasks: Given T_{-I} and a candidate answer a, generate a clarification question q_{cl} such that $I \subseteq r$ (the missing information is contained in the user's response) for r a user response to q_{cl}; and find a new answer a_{new} such that $a_{new} = a^*$.

To simulate conversations with clarifications, the teacher model guides the student model to generate a clarification question q_{cl} for each example in C, asking for more information. The teacher model will not give the ground-truth but gives hints of what

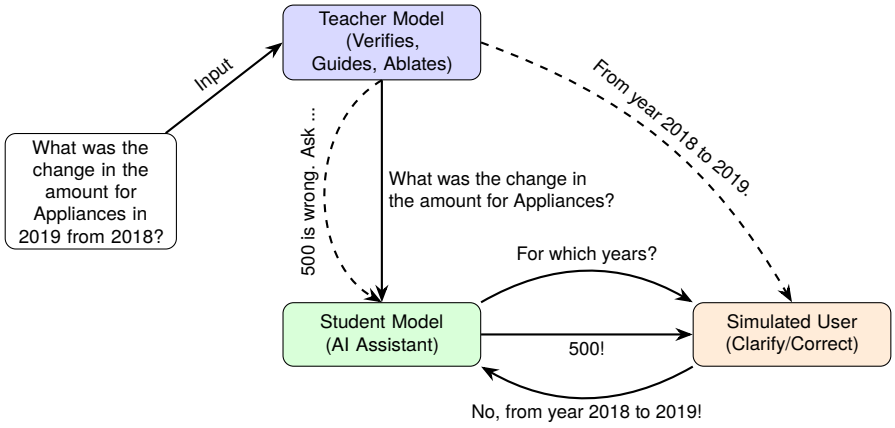

Figure 2: Teacher-Student framework illustrating synthetic dialogue generation with clarifications and corrections. Starting with an initial (single turn) TableQA task, the teacher model rephrases and removes information such that the student cannot solve it (anymore). Then, the teacher **guides** the student how to ask clarification questions and how to use the provided user corrections to solve the task.

information is needed to answer the question based on the above ablations.

Given the clarification question, the teacher model further simulates a user response r containing the missing information. We verify that the clarification question together with the user response does help solving the task by comparing the model's final answer with the ground-truth. If the answer is correct, all aspects are verified and we add the sample to the set of clarifiable questions Cl.

3B. Correction Analogous to the clarification process, the teacher instructs the student model to answer the *unsolvable* task with a given user correction without initiating a clarification question.

Given T_{-I}, a (possibly wrong) candidate answer a, and a user response r such that $I \subseteq r$, find a new answer a_{new} such that $a_{new} = a^*$. Hence, a user-initiated correction task is similar to the clarification task, but here the model does not initiate a clarification question and instead directly returns an answer, which will by definition be incorrect.

To simulate conversations with corrections, the teacher model generates for each example in C a user response r with correction instructions containing the necessary information I. The student generates a new answer a_{new}, given the simulated response, which is evaluated by the teacher model comparing the new answer to the ground-truth. If the answer is correct, all aspects are verified and we add the sample to the set of correctable questions Co.

4. Benchmark Creation

To create a benchmark with synthetic conversations containing AI-initiated clarifications and user-initiated corrections we apply the above introduced teacher-student approach on two state-of-the-art TableQA benchmark datasets, **TAT-QA** (Zhu et al., 2021) and WikiTableQuestions (**Wiki-TQ**) (Pasupat and Liang, 2015). Both datasets contain user questions for a given table and a ground-truth answer. Our framework applies to all such datasets, as long as it contains a question, a table and a ground-truth answer. We select from each dataset 1000 samples from the dev-splits as initial TableQA tasks. We use GPT4-Turbo (OpenAI et al., 2024) as teacher model with access to the ground-truth data in all our experiments and use five different LLMs of different sizes and model families as student model acting as TableQA assistant: **GPT4-Turbo** itself as student model as capable as the teacher. **Llama3.1** (Grattafiori et al., 2024) both 8b as small and 70b as general purpose language models and **Qwen2.5** (Qwen et al., 2025) both 7b as small and 72b as larger models with stronger reasoning and coding skills.

Starting from the samples, the teacher creates the conversations for a benchmark with a student model. On each sample, the teacher will ablate information from the questions and the table, but will only add the sample to the benchmark if the student can come to a final correct answer after a clarification or correction (see above). This means, the final size of the benchmark depends on the accuracy of the student model on the AI-initiated clarification and user-initiated correction tasks under the supervision of the teacher.

In Tab. 1, we report the results of the generated benchmarks for the student models: (Acc_{Or}) measures the accuracies of the different student models as TableQA assistant on the initial tasks from TAT-QA and Wiki-TQ (without the ablations); based on the correctly answered initial tasks (in set C),

	TAT-QA					Wiki-TQ				
	Original	Ablate table		Ablate question		Original	Ablate table		Ablate question	
Student	Acc_{Or}	Acc_{Cl}	Acc_{Co}	Acc_{Cl}	Acc_{Co}	Acc_{Or}	Acc_{Cl}	Acc_{Co}	Acc_{Cl}	Acc_{Co}
GPT4-Turbo	0.829	0.797	0.882	0.566	0.834	0.665	0.711	0.749	0.507	0.862
Llama3.1 70b	0.571	0.737	0.786	0.62	0.716	0.637	0.555	0.665	0.483	0.747
Llama3.1 8b	0.234	0.351	0.439	0.329	0.351	0.295	0.299	0.417	0.355	0.553
Qwen2.5 72b	0.703	0.645	0.786	0.62	0.773	0.579	0.587	0.717	0.573	0.784
Qwen2.5 7b	0.438	0.473	0.581	0.442	0.487	0.3	0.417	0.467	0.377	0.587

Table 1: Performance of student models while creating the synthetic benchmarks for TableQA **with** teacher guidance. For each benchmark TAT-QA (left) and Wiki-TQ (right) we report: (i) Acc_{Or}, the baseline accuracy on the original table questions; (ii) Acc_{Cl}, the accuracy on tasks derived from originally correct examples after ablating essential information (from the table or question) and an AI-initiated clarification; and (iii) Acc_{Co}, the accuracy when the missing information is provided via a user-initiated correction. These metrics illustrate that larger models not only achieve higher baseline accuracies but also recover more effectively when corrections are provided, compared to when they must generate clarification questions on their own.

(Acc_{Cl}) measures the accuracies of the student models on the AI-initiated clarification tasks with teacher model guidance; and (Acc_{Co}) measures the accuracies of the student models on the corrections tasks with the teacher model guidance.

We observe that the bigger models (GPT4, Llama3.1 70b, and Qwen2.5 72b) perform the best on the initial tasks, resulting in a larger pool of candidates for their benchmark (high Acc_{Or}). These models are also better in solving the AI-initiated clarifications tasks (high Acc_{Cl}) and user-initiated correction tasks (high Acc_{Co}) after the teachers ablates information. For example, Qwen2.5 72b can solve 70% of the original TAT-QA tasks. For these tasks, we perform the above-described ablations such that Qwen2.5 72b can no longer solve them. Now, generating a clarification question, the model can correct the answer in 64% of the cases; given directly the missing information as user correction, the model can correct the answer in 78% of the cases. We see that generating a clarification question results in lower performance compared to when a user gives the correction directly (without making the model generate the clarification question). This discrepancy is attributed to the model not being able to integrate the missing information from the simulated user in their new solution. The analysis in section 4.1 shows the simulated user response contains the right missing information with high accuracy.

Based on the performance of the students, we can generate a benchmark with conversations containing a set Cl of clarifications and a set Co of corrections. Starting from the initial TableQA tasks, the sizes of these sets depend on the accuracy of the corresponding student models, i.e., for Llama3.1 8b starting with 1000 of the Wiki-TQ TableQA tasks, the benchmark will contain around 299 clarifications and 417 corrections. While this means we throw away a lot of TableQA tasks due to the student not being able to finally get to a correct answer, we ensure the benchmark contains only tasks which are solvable in principle by a clarification or a correction.

Teacher / Student	GPT4 -Turbo	Llama3.1 70b	Llama3.1 8b	Qwen2.5 72b	Qwen2.5 7b
GPT4-Turbo	0.86	0.64	0.66	0.81	0.63
Llama3.1 70b	0.88	0.76	0.71	0.89	0.73
Llama3.1 8b	0.78	0.57	0.56	0.766	0.56

Table 2: Pearson correlations between human annotations and correctness judgments of different teacher models, higher values indicate higher agreement on correctness.

4.1. Teacher Model Analysis

The teacher model plays a central role at several locations in our framework including a) measuring the correctness of a student model answer, b) rephrasing the TableQA task; and c) simulating a user response providing information to correct an answer. We analyze the quality of the teacher on these tasks using both human annotations and automatic comparisons methods.

Teacher Model We hand-labelled the answers from three different student models (GPT4-Turbo, Llama3.1 70b and Llama3.1 8b) for correctness and compared the results with judgments from different teacher models (all considered models: GPT4-Turbo, Llama3.1 70b and 8b, Qwen2.5 72b and 7b).

We see a strong correlation 2 between our human judgments of correctness of the student answers with GPT4-Turbo's as well as Qwen2.5 72b's correctness judgments as teacher model. These models also perform well on the original TableQA tasks, which aligns with findings that the LLM's abilities on evaluation tasks depends on their ability to

	Sentence Sim		2-Rouge	
	Original	Pacific	Original	Pacific
Ours	0.77	0.74	0.71	0.71
Pacific	0.84	1	0.77	1

Table 3: Average sentence similarity and 2-gram overlap (Rouge-score) between the original user questions in the TAT-QA dataset and their rephrased version to remove information by both, our proposed teacher model and the human rephrases in the Pacific dataset.

solve the task themselves (Lin et al., 2024b). The major source of disagreements come from formatting issues i.e., rounding a number like 0.59 given a ground-truth answer of 0.6. Based on these finding we choose GPT4-Turbo as teacher model for all our experiments.

Task ablation To study the quality of ablating question information by the teacher model, we compare the resulting rephrased questions q' to the original question, as well as to the corresponding human questions from the Pacific (Deng et al., 2022) dataset as reference. We find that the rephrased questions in both approaches, ours from the teacher model and human ones from the Pacific dataset, are both semantically and lexically similar. In Tab.3, we report the sentence similarity (Reimers and Gurevych, 2019) and the 2-rouge scores (Lin, 2004) between original questions from TAT-QA, the rephrased questions by the teacher model and the questions from the Pacifc dataset.

We find that the teacher rephrases the questions similar to how humans perform this task in the Pacific dataset. For example one strategy to rephrase the question is to remove nominal entities mentioned at the end, which is also observed in the Pacific dataset, i.e., the original question *What was the discount rate for 2019?* from the TAT-QA dataset is rephrased both by our teacher model approach and the human annotators in the Pacific dataset as *What was the discount rate?*.

Analogue to the questions, we study the quality of ablating table information by measuring the differences to the original table. In 98% of all cases, the teacher changes the table after being instructed to remove relevant information. 36% of the time the teacher removes columns, 67% of the time the teacher removes values (not a whole column). We find that on average 5% of the original table is removed (the main table stays intact) and in 85% of the cases the table ablation successfully results in a deficiency to solve the task, where at least one of the student models can clarify or correct it.

User responses We analyze how well the teacher model simulates a user response by measuring the amount of the ablated information contained to correct an answer. We perform a semi-automatic annotation where we first check how much of the ablated information (either from the question or the table) is contained in the user response. To account for rephrasing and simple misses due to formatting, we add an additional human annotation to perform the same task.

For ablating question information, we find that in more than 99% of all simulated user responses, the ablated information is contained, either directly or rephrased. This is also independent on whether the student can finally answer the question. For ablating table information, we find that in 95% of all simulate user responses, the response contains a subset of the ablated information from the table. In average 83% of the ablated information is returned to the student by the simulated user. We observe that the teacher model does not always return all the information it has removed from the table. Non-relevant, but ablated information, are often omitted.

4.2. Human study

We additionally conducted a crowd-sourced annotation study on Prolific to assess the quality of the teacher-generated questions, the simulated user responses, and the overall conversational quality. Three independent annotators evaluated 120 examples from the GPT4-generated benchmark, assigning scores from 1 (low) to 3 (excellent).

On average, both the conversational quality and the simulated interactions were rated highly. For cases involving missing information in the question, the teacher-generated clarification questions and simulated user responses received mean scores of 2.7 and 2.8, respectively. For cases involving missing information in the table, the scores were slightly lower, at 2.5 for the questions and 2.6 for the responses. The overall conversational quality was rated at 2.6 for both question- and table-ablation scenarios.

Inter-annotator agreement was 72% for the quality of question ablations and 60% for table ablations, indicating moderate to substantial consensus among raters.

5. Evaluation

Based on the benchmarks generated for each student model from the dev-splits from TAT-QA and Wiki-TQ as introduced above, we evaluate the student models on solving the tasks in Cl and Co **without** the teacher guiding the clarification or correction process. We still leverage the teacher model to simulate user responses, but the teacher no longer helps the student directly.

We evaluate the student using two prompting

	TAT-QA					Wiki-TQ				
Student	P	R	$F1$	Acc_{Cl}	Acc_{Co}	P	R	$F1$	Acc_{Cl}	Acc_{Co}
GPT4-Turbo	0.82	0.36	0.5	0.81	0.91	0.81	0.38	0.51	0.67	0.75
Llama3.1 70b	0.55	0.94	0.69	0.86	0.89	0.56	0.91	0.69	0.63	0.71
Llama3.1 8b	0.29	1.0	0.45	0.63	*	0.59	1.0	0.74	0.57	*
Qwen2.5 72b	0.93	0.16	0.27	0.86	0.91	0.74	0.16	0.27	0.55	0.69
Qwen2.5 7b	0.44	1.0	0.61	0.78	*	0.45	0.98	0.62	0.71	0.6

Table 4: Performance of the student models on their created synthetic benchmarks for TableQA **without** teacher guidance using a **follow-up instruction prompt**. For each benchmark TAT-QA (left) and Wiki-TQ (right) we report: (i) precision, recall and F1-score on AI-initiated clarification questions; (ii) Acc_{Cl}, the accuracy on the examples with the ablated essential information after the AI-initiated clarification; and (iii) Acc_{Co}, the accuracy on the examples with the ablated essential information after a user-initiated correction. *No user-initiated corrections since the model asked always a clarification question. These metrics illustrate that smaller models almost always ask for clarification if they are prompted for such a decision.

strategies, as well as by supervised finetuning: 1) Similar to the teacher-student interactions to generate a benchmark, we first prompt the model to answer the table question and then prompt the model as follow-up to decide whether to ask a clarification question or not. After the response from the student model, we simulate a user response either providing the information asked for in the clarification question (if the student initiates this) or provide a direct user correction with the missing information if the model does not initiate clarification question. 2) We prompt the student model to answer the table questions and to decide whether to return either the answer or a clarification question in a single prompt (hence, first solve the task, next decide whether to ask for clarification), with zeroshot and fewshot prompting (Brown et al., 2020). 3) Finally, we create a new training dataset based on the train-split of the TableQA benchmarks TAT-QA and Wiki-TQ and finetune the student models on these tasks. Then, we measure whether the models can improve their performance on the clarification and corrections tasks through finetuning by evaluating them on the dev-split based benchmarks using zero-shot prompting.

Given the benchmark for each student we measure precision, recall and F1-score on generating clarification questions based on the successful clarifications (positive class) and successful original tasks which did not need a clarification to answer (negative class). Further we measure the accuracy of the final answer after either the student initiated a clarification (Cl) or the user directly corrected the model with the missing information if the model did not initiate a clarification questions (Co). This measure whether the student can recover from a wrong answer.

1) Follow-up instruction prompt First we evaluate the student models to solve the clarification and corrections tasks when prompted as a follow-up after receiving an initial answer (Tab. 4).

We observe that the stronger models are more reluctant in generating clarification questions, resulting in a lower recall, but when they ask they are accurate, resulting in high precision i.e., Qwen2.5 72b has a low recall of 0.16 but a high precision of 0.93 on benchmark from TAT-QA. For the weaker models, we observe the opposite, a high recall and a low precision i.e., Llama3.1 8b has a recall of 1 but a low precision of 0.29. These models are less confident in their answers, asking almost always clarification questions, even if not needed. This behavior is consistent across the datasets.

On both datasets, we see that recovery from a wrong answer without the teacher guidance, with or without asking a clarification questions, is challenging even for highly capable student models. GPT4-Turbo for example achieves 81% accuracy on tasks when it initiated a clarification, and 91% when a user directly provided the correction. For the smaller models, like Llama3.1 8b, we observe lower accuracies, ranging between 57% and 67%.

As observed before, we see higher accuracies on the corrections tasks (Co) when the student models decide not to ask a clarification question, compared to the AI-initiated clarification tasks (Cl) when the models ask a clarification question. For example, Llama3.1 70b is only able to give a correct final answer after clarification in 63% of the tasks in the benchmark from Wiki-TQ, but in 71% of the tasks without asking a clarification question.

2) Few-shot prompts In contrast to testing the student models with a follow-up prompt to decide whether to ask a clarification question, next we test them in a single step, prompting them only once to generate an answer and clarification question if needed. This tests the student models on their self-criticizing abilities. We are testing the students using zero- and fewshot prompting (Tab. 5 - top and middle section). For the fewshot prompts we

	TAT-QA					Wiki-TQ				
Student	P	R	$F1$	Acc_{Cl}	Acc_{Co}	P	R	$F1$	Acc_{Cl}	Acc_{Co}
Zeroshot										
GPT4-Turbo	0.98	0.31	0.47	0.9	0.86	0.98	0.44	0.61	0.65	0.79
Llama3.1 70b	0.96	0.09	0.16	0.8	0.75	0.93	0.22	0.35	0.53	0.76
Llama3.1 8b	0.5	0.01	0.03	0.67	0.46	0.62	0.03	0.05	0.55	0.61
Qwen2.5 72b	1.0	0.07	0.13	0.88	0.79	1.0	0.09	0.17	0.79	0.74
Qwen2.5 7b	0.92	0.22	0.36	0.55	0.66	0.84	0.34	0.49	0.57	0.72
Fewshot										
GPT4-Turbo	0.98	0.28	0.43	0.89	0.83	0.97	0.36	0.52	0.59	0.6
Llama3.1 70b	0.84	0.12	0.2	0.7	0.64	0.96	0.19	0.32	0.62	0.66
Llama3.1 8b	0.29	0.08	0.11	0.33	0.33	0.86	0.07	0.12	0.43	0.42
Qwen2.5 72b	1.0	0.1	0.19	0.85	0.74	0.91	0.08	0.14	0.71	0.71
Qwen2.5 7b	0.88	0.46	0.59	0.46	0.44	0.85	0.48	0.61	0.57	0.66

Table 5: Performance of the student models on their created synthetic benchmark for TableQA **without** teacher guidance using **zero and fewshot prompts**. For each benchmark TAT-QA (left) and Wiki-TQ (right) we report: (i) precision, recall and F1-score on AI-initiated clarification questions; (ii) Acc_{Cl}, the accuracy on the examples with the ablated essential information after the AI-initiated clarification; and (iii) Acc_{Co}, the accuracy on the examples with the ablated essential information after a user-initiated correction. These metrics illustrate with zero- and fewshot prompts all models are reluctant in asking clarifications before returning an answer.

	TAT-QA				
Student	P	R	$F1$	Acc_{Cl}	Acc_{Co}
Llama3.1 8b	**0.62**	**0.56**	**0.59**	**0.71**	**0.78**
Qwen2.5 7b	0.45	**0.71**	**0.55**	**0.72**	**0.69**

Table 6: Performance of the student models on the synthetic benchmark for TableQA after finetuning. Finetuning increases the recall and also improve the accuracies on the final answers. Bold numbers indicate improvement over the non-finetuned model using zero-shot prompting.

extract 4 examples from the train benchmarks and present them in two different orders to avoid positional biases.

We observe that, in contrast to testing the students with a follow-up prompt, all models are now more reluctant to ask clarification questions. With zeroshot prompting for example, the smaller student models Llama3.1 8b, and Qwen2.5 7b drop in recall from 1 to 0.22, respectively 0.01 when not prompted as a follow-up to decide to ask a clarification question and instead prompted to make this decision before returning an answer.

3) Finetuning We additionally generate a training dataset using the teacher-student method using the train-splits of the TableQA datasets from TAT-QA (Tab. 6). We finetune Llama3.1 8b and Qwen2.5 7b for four epochs on this train dataset and test both finetuned models with zeroshot prompting on the synthetically generated benchmarks from above, analogue to the previous evaluation. We observe that after finetuning, the performance of both models improve in generating clarification questions and fixing the initial incorrect answer after a clarification or a correction from a user. Llama3.1 8b benefits the most from finetuning, significantly increasing in recall compared to zeroshot and fewshot prompting the non-finetuned model.

6. Conclusion

In this work, we introduced a novel teacher–student framework to generate a benchmark of synthetic conversations with AI-initiated clarifications and user-initiated corrections for TableQA tasks. Our approach leverages a strong teacher model to verify that every synthesized dialogue constitutes a solvable sub-task, ensuring that the generated data provides a reliable benchmark for evaluating large language models. Further, in contrast to previous approaches, we leverage ablations of information from both the question and the table to synthetically generate dialogues with the need of clarifications and corrections.

We conducted extensive experiments using two well-known TableQA datasets (TAT-QA and Wiki-TQ) and evaluated a range of student models, from highly capable systems such as GPT4-Turbo to smaller models like Llama3.1 8b and Qwen2.5 7b. Our results show that stronger models not only achieve higher baseline accuracy on original tasks but are also more effective at handling information deficiencies when guided by teacher feedback. Moreover, we demonstrate that finetuning on the synthetic conversations significantly improves student models' ability to generate clarification questions and correctly incorporate user-initiated corrections.

7. Limitations

The benchmarks are dependent on the characteristics of a specific student model, meaning it may overlook failure cases that are common across other models. Additionally, the framework requires a strong teacher LLM to generate and validate the synthetic conversations—although this is still more cost-effective than human annotation, it can be expensive and the teacher model may itself exhibit issues such as hallucinations, biases, or lower performance than human evaluators. Even though the main authors performed a human evaluation on the generated questions and corrections, we cannot guarantee general correctness and naturalness. Hallucinations and data leakages are possible, although our approach makes sure that the student never has access to the ground-truth and we only keep generated dialogues that end in a correct answer.

8. Bibliographical References

Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2021a. Building and evaluating open-domain dialogue corpora with clarifying questions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4473–4484, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeffrey Dalton, and Mikhail Burtsev. 2021b. Building and evaluating open-domain dialogue corpora with clarifying questions.

Jianzhu Bao, Rui Wang, Yasheng Wang, Aixin Sun, Yitong Li, Fei Mi, and Ruifeng Xu. 2023. A synthetic data generation framework for grounded dialogues. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10866–10882, Toronto, Canada. Association for Computational Linguistics.

Shraddha Barke, Christian Poelitz, Carina Negreanu, Benjamin Zorn, José Cambronero, Andrew Gordon, Vu Le, Elnaz Nouri, Nadia Polikarpova, Advait Sarkar, Brian Slininger, Neil Toronto, and Jack Williams. 2024. Solving data-centric tasks using large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 626–638, Mexico City, Mexico. Association for Computational Linguistics.

Keping Bi, Qingyao Ai, and W. Bruce Croft. 2021. Asking clarifying questions based on negative feedback in conversational search. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '21, page 157–166, New York, NY, USA. Association for Computing Machinery.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2020. Multiwoz – a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling.

Maximillian Chen, Ruoxi Sun, Sercan Ö. Arık, and Tomas Pfister. 2024. Learning to clarify: Multi-turn conversations with action-based contrastive self-training.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan R Routledge, et al. 2021. Finqa: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3697–3711.

Yang Deng, Wenqiang Lei, Wai Lam, and Tat-Seng Chua. 2023a. A survey on proactive dialogue systems: problems, methods, and prospects. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, IJCAI '23.

Yang Deng, Wenqiang Lei, Wenxuan Zhang, Wai Lam, and Tat-Seng Chua. 2022. PACIFIC: Towards proactive conversational question answering over tabular and textual data in finance. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6970–6984, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yang Deng, Lizi Liao, Liang Chen, Hongru Wang, Wenqiang Lei, and Tat-Seng Chua. 2023b. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. In *Findings of the Association for Computational Linguistics:*

EMNLP 2023, pages 10602–10621, Singapore. Association for Computational Linguistics.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Ar-

caute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models.

Alex Gu, Baptiste Roziere, Hugh James Leather, Armando Solar-Lezama, Gabriel Synnaeve, and Sida Wang. 2024. CRUXEval: A benchmark for code reasoning, understanding and execution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 16568–16621. PMLR.

Meiqi Guo, Mingda Zhang, Siva Reddy, and Malihe Alikhani. 2021. Abg-coQA: Clarifying ambiguity in conversational question answering. In *3rd Conference on Automated Knowledge Base Construction*.

Atharva Kulkarni, Bo-Hsiang Tseng, Joel Ruben Antony Moniz, Dhivya Piraviperumal, Hong Yu, and Shruti Bhargava. 2024. SynthDST: Synthetic data is all you need for few-shot dialog state tracking. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1988–2001, St. Julian's, Malta. Association for Computational Linguistics.

Haoran Li, Qingxiu Dong, Zhengyang Tang, Chaojun Wang, Xingxing Zhang, Haoyang Huang, Shaohan Huang, Xiaolong Huang, Zeqiang Huang, Dongdong Zhang, Yuxian Gu, Xin Cheng, Xun Wang, Si-Qing Chen, Li Dong, Wei Lu, Zhifang Sui, Benyou Wang, Wai Lam, and Furu Wei. 2024. Synthetic data (almost) from scratch: Generalized instruction tuning for language models.

Xiao Li, Yawei Sun, and Gong Cheng. 2021. Tsqa: Tabular scenario based question answering. *Pro-*

ceedings of the AAAI Conference on Artificial Intelligence, 35(15):13297–13305.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2024a. Wildbench: Benchmarking llms with challenging tasks from real users in the wild.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zicheng Lin, Zhibin Gou, Tian Liang, Ruilin Luo, Haowei Liu, and Yujiu Yang. 2024b. Criticbench: Benchmarking llms for critique-correct reasoning.

Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, and Andrew M. Dai. 2024. Best practices and lessons learned on synthetic data.

Lin Long, Rui Wang, Ruixuan Xiao, Junbo Zhao, Xiao Ding, Gang Chen, and Haobo Wang. 2024. On LLMs-driven synthetic data generation, curation, and evaluation: A survey. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11065–11082, Bangkok, Thailand. Association for Computational Linguistics.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: Designing data and methods for effective instruction tuning.

Fangwen Mu, Lin Shi, Song Wang, Zhuohao Yu, Binquan Zhang, ChenXue Wang, Shichao Liu, and Qing Wang. 2024. Clarifygpt: A framework for enhancing llm-based code generation via requirements clarification. *Proc. ACM Softw. Eng.*, 1(FSE).

Bhuvanashree Murugadoss, Christian Poelitz, Ian Drosos, Vu Le, Nick McKenna, Carina Suzana Negreanu, Chris Parnin, and Advait Sarkar. 2024. Evaluating the evaluator: Measuring llms' adherence to task evaluation instructions.

Rungsiman Nararatwong, Natthawut Kertkeidkachorn, and Ryutaro Ichise. 2022. Enhancing financial table and text question answering with tabular graph and numerical reasoning. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 991–1000, Online only. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Ra-

jeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report.

Paul Owoicho, Ivan Sekulic, Mohammad Aliannejadi, Jeffrey Dalton, and Fabio Crestani. 2023. Exploiting simulated user feedback for conversational search: Ranking, rewriting, and beyond. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 632–642, New York, NY, USA. Association for Computing Machinery.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480, Beijing, China. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report.

Hossein A. Rahmani, Xi Wang, Yue Feng, Qiang Zhang, Emine Yilmaz, and Aldo Lipani. 2023. A survey on asking clarification questions datasets in conversational systems. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2698–2716, Toronto, Canada. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Xuhui Ren, Hongzhi Yin, Tong Chen, Hao Wang, Zi Huang, and Kai Zheng. 2021. Learning to ask appropriate questions in conversational recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 808–817, New York, NY, USA. Association for Computing Machinery.

Shivchander Sudalairaj, Abhishek Bhandwaldar, Aldo Pareja, Kai Xu, David D. Cox, and Akash Srivastava. 2024. Lab: Large-scale alignment for chatbots.

Leila Tavakoli, Johanne R. Trippas, Hamed Zamani, Falk Scholer, and Mark Sanderson. 2024. Online and offline evaluation in search clarification. *ACM Trans. Inf. Syst.*, 43(1).

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Zeqiu Wu, Ryu Parish, Hao Cheng, Sewon Min, Prithviraj Ammanabrolu, Mari Ostendorf, and

Hannaneh Hajishirzi. 2023. InSCIt: Information-seeking conversations with mixed-initiative interactions. *Transactions of the Association for Computational Linguistics*, 11:453–468.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions.

Jingjing Xu, Yuechen Wang, Duyu Tang, Nan Duan, Pengcheng Yang, Qi Zeng, Ming Zhou, and Xu Sun. 2019. Asking clarification questions in knowledge-based question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1618–1629, Hong Kong, China. Association for Computational Linguistics.

Hamed Zamani, Susan Dumais, Nick Craswell, Paul Bennett, and Gord Lueck. 2020. Generating clarifying questions for information retrieval. In *Proceedings of The Web Conference 2020*, WWW '20, page 418–428, New York, NY, USA. Association for Computing Machinery.

Yueheng Zhang, Xiaoyuan Liu, Yiyou Sun, Atheer Alharbi, Hend Alzahrani, Basel Alomair, and Dawn Song. 2025. Can llms design good questions based on context?

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. 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 3277–3287, Online. Association for Computational Linguistics.

Jie Zou, Aixin Sun, Cheng Long, Mohammad Aliannejadi, and Evangelos Kanoulas. 2023. Asking clarifying questions: To benefit or to disturb users in web search? *Information Processing Management*, 60(2):103176.