MiMoTable: A Multi-scale Spreadsheet Benchmark with Meta Operations for Table Reasoning

Zheng Li*, **Yang Du***, **Mao Zheng***, **Mingyang Song***
Tencent Hunyuan
jasonzli@tencent.com

Abstract

Extensive research has been conducted to explore the capability of Large Language Models (LLMs) for table reasoning and has significantly improved the performance on existing benchmarks. However, tables and user questions in real-world applications are more complex and diverse, presenting an unignorable gap compared to the existing benchmarks. To fill the gap, we propose a **M**ulti-scale spreadsheet benchmark with **M**eta **o**perations for **Table** reasoning, named as MiMoTable. Specifically, MiMoTable incorporates two key features. First, the tables in MiMoTable are all spreadsheets used in real-world scenarios, which cover seven domains and contain different types. Second, we define a new criterion with six categories of meta operations for measuring the difficulty of each question in MiMoTable, simultaneously as a new perspective for measuring the difficulty of the existing benchmarks. Experimental results show that Claude-3.5-Sonnet achieves the best performance with 77.4% accuracy, indicating that there is still significant room to improve for LLMs on MiMoTable. Furthermore, we grade the difficulty of existing benchmarks according to our new criteria. Experiments have shown that the performance of LLMs decreases as the difficulty of benchmarks increases, thereby proving the effectiveness of our proposed new criterion. All data and code are open-sourced at https://github.com/jasonNLP/MiMoTable.

1 Introduction

Tabular data plays a crucial role across diverse domains, including education, finance, and others. Table reasoning involves deriving meaningful insights and answers from structured tabular data to address specific user queries (Zhang et al., 2024). This process significantly improves the efficiency of information retrieval and interpretation for users.

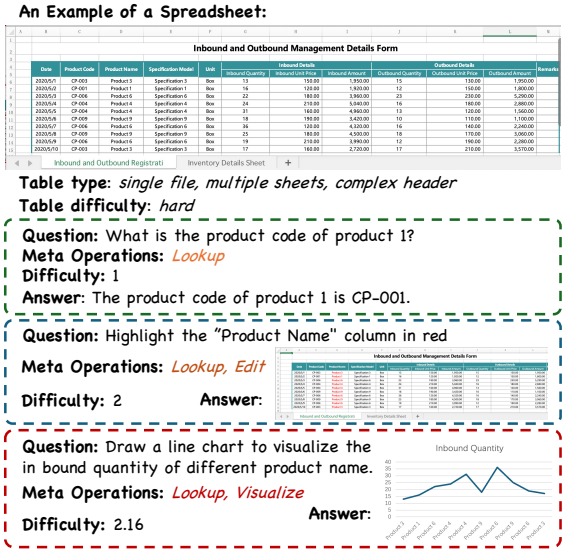

Figure 1: Examples of MiMoTable benchmark.

To foster a comprehensive understanding of this field, researchers have proposed and developed numerous table reasoning tasks, such as TableQA, Table2Text, Table Manipulation, and Advanced Data Analysis (Lu et al., 2024). Various methods have been proposed to tackle these tasks, and large language models (LLMs) have achieved promising results (Liu et al., 2022; Cheng et al., 2023). To evaluate performance, several table reasoning benchmarks have been introduced, including WikiTableQuestions (Pasupat and Liang, 2015), ToTTo (Parikh et al., 2020), SheetCopilot (Li et al., 2023a), Text2Analysis (He et al., 2024) and so on.

In the realm of table reasoning, benchmark development has not kept pace with the rapid advancements in methodological approaches. While LLMs have exhibited remarkable performance on existing benchmarks, recent studies have brought to light persistent limitations in their capacity for nuanced table comprehension (Sui et al., 2024). Upon critical analysis, we have identified shortcomings in existing benchmarks across two key aspects.

*Equal contribution.

Proceedings of the 31st International Conference on Computational Linguistics, pages 2548–2560
January 19–24, 2025.

First, current benchmarks exhibit significant limitations in their representation of real-world tabular data complexity. Most tables in these benchmarks have simple headers (single-row/column) and fail to cover all four task types comprehensively. However, real-world tables are diverse and can be divided into three parts: 1) Headers ranging from single-row/column to complex hierarchical forms. 2) Variable number of sheets in Excel files. 3) Multiple tables within a single sheet. These complexities are often overlooked in existing benchmarks, limiting their ability to accurately assess table reasoning capabilities in practical applications.

Second, although current existing benchmarks are divided according to task granularity, the difficulty of different benchmark datasets within the same task can vary. For example, the WikiSQL (Zhong et al., 2017) dataset is simpler than the unrestricted WikiTableQuestions dataset because it limits questions to those that can be answered using a subset of SQL queries. The current task divisions cannot reflect this difference in difficulty.

To address the above issues, we propose MiMoTable, a table reasoning benchmark with diverse spreadsheets and meta operations. Our dataset comprises 428 spreadsheets from real-world scenarios, spanning seven domains: architecture, finance, office, education, accounting, e-commerce, and manufacturing. Our table data is comprehensive, featuring both simple and complex headers, and varying in the number of sheets from single to multiple. Some spreadsheets even contain multiple tables within a single sheet. We have constructed 1,719 question-answer pairs based on these spreadsheets, forming (spreadsheet, question, answer) triplets. Examples of MiMoTable are shown in Figure 1.

Simultaneously, to more deeply reflect the differences in dataset problems, we propose a new criterion for categorizing problems based on meta operations. There are six types of meta operations: Lookup, Edit, Compare, Calculate, Visualize, and Reasoning. Each type of meta operation corresponds to a difficulty score. With the new criterion, we can associate each problem with one or more meta operations, thereby assigning a difficulty score to each problem. In this way, different benchmarks can be graded into different difficulty scores, facilitating better analysis and comparison.

Our main contributions are as follows:

- We propose a new benchmark comprising 428 multi-scale spreadsheets in both Chinese and English, featuring simple and complex headers, single and multiple sheets, single and multiple files, and multiple tables within a single sheet. Based on these characteristics, we classify the tables into three difficulty levels: simple, medium, and hard. We construct 1,719 (spreadsheet, question, answer) triplets covering a wide range of tasks.

- We introduce a novel criterion for categorizing table reasoning problems using meta operations, each assigned a difficulty score. These non-overlapping meta operations can be combined to represent existing tasks, allowing for a more precise evaluation of a model's capabilities in table-related tasks.

- We conducted extensive experiments, demonstrating that the proposed benchmark is challenging for existing LLMs and proving the effectiveness of the proposed meta operations.

	Simple Header	Complex Header
Single Sheet	*simple table*	*medium table*
Multiple Sheets	*medium table*	*hard table*
Multiple Files	*medium table*	*hard table*
Multiple Tables	*hard table*	*hard table*

Table 1: Categories of table difficulty.

2 MiMoTable Benchmark

In this section, we introduce how to prepare our new MiMoTable benchmark and ensure its quality.

2.1 Types and Difficulty of Tables

Most existing table reasoning benchmarks utilize single tables with simple headers, contrasting with the diverse tables encountered in real-world scenarios, particularly in spreadsheets like Excel files. After analyzing real-world spreadsheets, we categorize them along four dimensions: header types, the number of sheets per file, the number of tables per sheet, and the file count.

As shown in Figure 2, header types can be divided into simple headers and complex headers. A simple header refers to a single-row or single-column header, while all others are considered complex headers. For example, hierarchical headers are classified as complex headers. Understanding complex headers is more challenging than understanding simple ones, and multiple sheets generally contain more information than a single sheet. There-

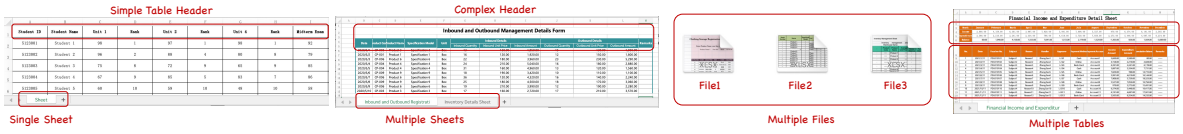

Figure 2: Illustrations of different table types, including simple header, complex header, single sheet, multiple sheets, multiple files, and multiple tables in one sheet.

Meta Operations	Description	Grade	Examples
Lookup	Locate the position of specific target	1	What is the product code of product 1?
Edit	Modify, delete or add in a table	1	Highlight the “Product Name" column in red
Calculate	The numerical computation, sum, avg, max, etc	2	How many students are in the table?
Compare	Compare two or more targets in a table	2	Who has the highest score?
Visualize	Show in chart	2	Draw a chart to show the distribution of scores.
Reasoning	Inferring information from the table content that is not explicitly included	3	Analyze the relationship between the loan term, monthly interest, and interest.

Table 2: The description and grade of meta operations.

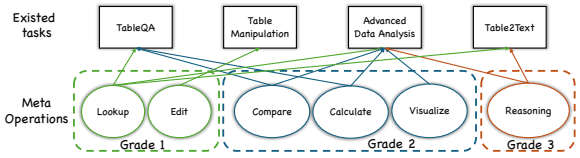

Figure 3: The relationships between tasks in the existing benchmarks and our proposed meta operations.

fore, based on the aforementioned dimensions, we classify spreadsheets into three difficulty levels: simple, medium, and hard. The specific classification rules are illustrated in Table 1.

2.2 Meta Operations

Current table reasoning benchmarks are classified by tasks, mainly including TableQA, Table2Text, Table Manipulation, and Advanced Data Analysis. This task-based categorization evaluates model performance across different tasks but fails to measure differences between benchmarks within the same task or compare benchmarks from different tasks along the same dimension.

To enhance the analysis of table reasoning benchmarks, we propose a novel criterion categorizing questions by meta operations: Lookup, Edit, Calculate, Compare, Visualize, and Reasoning. Table 2 defines each operation. These operations reflect specific LLM capabilities in handling table-related problems, with questions potentially involving multiple operations. For instance, "How many students are in the table" requires both Lookup (locating student names) and Calculate (counting them) operations.

The combination of six meta operations can encompass tasks in current table benchmarks. Figure 3 shows the mapping relationship between existing tasks and meta operations. For instance, TableQA questions may involve combinations of Lookup, Compare, and Calculate operations.

Additionally, to assess problem complexity, we categorize the six meta operations into three difficulty grades (1, 2, 3) based on common criteria, as shown in Table 2. Lookup and Edit, involving simple content location or modification, are grade 1. Compare, Calculate, and Visualize, which require logical operations, are grade 2. Reasoning, necessitating inference beyond explicit table content, is the most complex at grade 3.

With the difficulty score of meta operations, we can calculate the difficulty score of each question and the entire dataset. First, let’s assume there are N questions in the dataset. The i-th problem q_i can be associated with K_i meta operations. Suppose the k-th meta operation is denoted as op_k, then the sequence of meta operations for the i-th questions can be represented as:

$$OP_{q_i} = [op_1, op_2, ..., op_{K_i}] \quad (1)$$

The difficulty sequence corresponding to the meta operations can be represented as:

$$S_{q_i} = [s_1, s_2, ..., s_{K_i}], s_k \in 1, 2, 3 \quad (2)$$

where s_{K_i} indicates the difficulty score corresponding to meta operations op_k.

To ensure that questions involving more difficult meta operations are assigned a higher difficulty

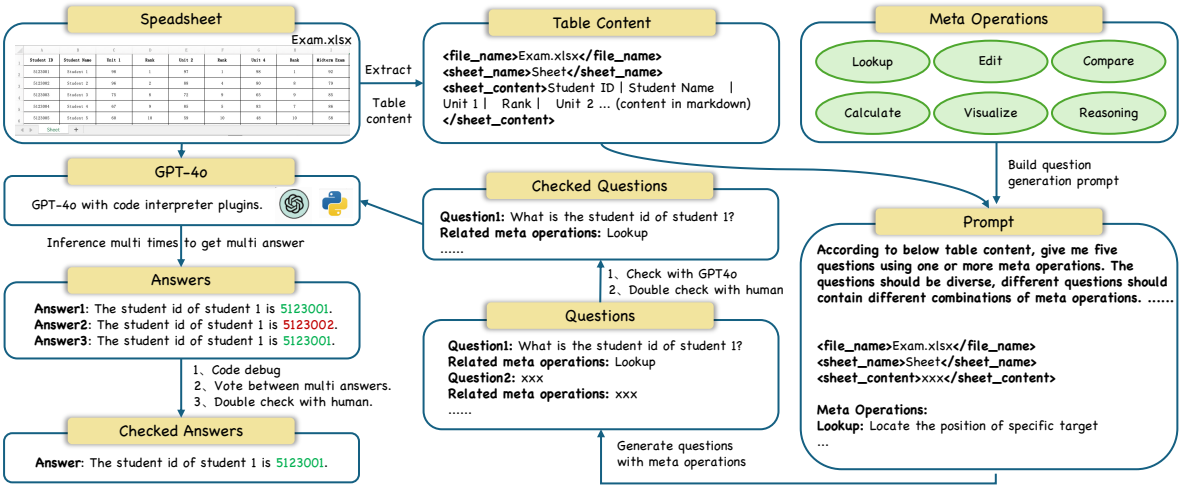

Figure 4: The data construction pipeline of MiMoTable benchmark.

score, we define the difficulty score of a question q_i as follows:

$$qs_i = ms_{q_i} + (\sum_{1}^{K_i} s_i - ms_{q_i})/M_{ms_{q_i}} \quad (3)$$

$$ms_{q_i} = \max(S_{q_i}) \quad (4)$$

The term $M_{ms_{q_i}}$ refers to the maximum value that $\sum_{1}^{K_i} s_i - ms_{q_i}$ can be achieved under the condition where the meta operations of the question have the highest level of difficulty ms_{q_i}. Every meta operation can only appear once in each question. So when the $ms_{q_i} = 3$, which means the corresponding meta operation is Reasoning, the most complex combination of the rest of the meta operations is Compare, Calculate, Visualize, Lookup, and Edit. The difficulty score sum of those meta operations is 2 + 2 + 2 + 1 + 1 = 8. So M_3=8. And in the same manner, we can get all the values of $M_{ms_{q_i}}$ as follows,

$$M_{ms_{q_i}} = \begin{cases} 1 & ms_{q_i} = 1 \\ 6 & ms_{q_i} = 2 \\ 8 & ms_{q_i} = 3 \end{cases} \quad (5)$$

So the range of qs_i is [1, 4]. With the difficulty score qs_i of a single question q_i, we define the difficulty score of the entire dataset, ds, as the average of the difficulty scores of all questions:

$$ds = \frac{\sum_1^N qs_i}{N} \quad (6)$$

2.3 Dataset Construction

We introduce how the dataset is constructed from three aspects: table collection, question generation, and answer generation. Figure 4 illustrates the whole construction process.

Table Collection. Since Excel files are the most popular spreadsheet in real-world scenarios, we choose .xlsx as the file format to be collected. The spreadsheets of our dataset are collected from publicly available sources on the internet. The Chinese tables primarily come from Baidu Wenku, while the English tables are mainly sourced from Google searches. These spreadsheets cover seven common domains: architecture, finance, office, education, accounting, e-commerce, and manufacturing. To ensure that the types of spreadsheets encompass as many real-world scenarios as possible, according to the classifications of table type mentioned before, we collected spreadsheets with both simple and complex headers, as well as those with single and multiple sheets. Even within a single sheet, our data may contain multiple tables. Additionally, we randomly sampled some individual spreadsheet files and combined them into groups of 2-5 files, and the subsequent questions and answers are generated with those multiple files as input. To maintain the quality of the collected spreadsheets, we manually checked the content, removing files with significant noise, garbled text, or non-tabular formats. Furthermore, we reviewed each spreadsheet to anonymize any potential private information. Specifically: (1) personal names, contact information, addresses, etc., are masked

Benchmarks	Table Types				Tasks			
	Header Type	Sheet Num	File Num	Table Num	TableQA	Table2Text	Table Manipulation	Advanced Data Analysis
WikiTableQuestion	simple	single	single	single	✓			
WikiSQL	simple	single	single	single	✓			
FetaQA	simple	single	single	single	✓			
HiTAB	complex	single	single	single	✓	✓		
ToTTo	simple	single	single	single		✓		
DAEval	simple	single	single	single				✓
WikiTableEdit	simple	single	single	single			✓	
Text2Analysis	simple	single	single	single	✓	✓		✓
MiMoTable(ours)	simple & complex	single & multiple	single & multiple	single & multiple	✓	✓	✓	✓

Table 3: Comparison in table types and tasks between existing benchmarks and MiMoTable

and randomly regenerated by GPT, while headers are retained due to their importance and generality; (2) we double-checked the final spreadsheets with legal professionals.

Ultimately, we obtained 428 high-quality spreadsheets containing both Chinese and English languages and various types. As shown in Table 3, compared to the current benchmarks, our collected spreadsheets far exceed in diversity of types, better reflecting various real-world scenarios.

Question Generation. As shown in Figure 4, we use GPT-4o to generate relevant questions and double-check with models and humans. First, we extract the table content from a spreadsheet in markdown format. Then, according to the extracted table content and our meta operations, GPT-4o is prompted to generate related questions. We instructed the model to generate multiple questions at once for each spreadsheet. To ensure the diversity of questions, the multiple questions should contain different combinations of meta operations. For multi-sheet or multi-file spreadsheets, we prompt the model to generate questions requiring cross-sheet or cross-file analysis. To ensure prompt effectiveness, we initially generate 50 samples, conduct a human evaluation to identify issues, and iteratively refine the prompt until most generated questions meet our criteria.

After generating initial questions with GPT-4o, we prompt it to verify if they meet requirements: relevance to table content and correct meta operations. We then manually review and filter out unsuitable questions, yielding 1,719 high-quality, comprehensive questions.

We classify the questions by existing tasks, covering TableQA, Table2Text, Table Manipulation, and Advanced Data Analysis. As Table 3 shows, our dataset exceeds all current table benchmarks in task comprehensiveness.

Answer Generation. After we collected the tables and generated the related questions based on meta operations, the final step is to obtain the corresponding answers. Since some questions are related to editing on the origin spreadsheet or drawing charts, we leverage GPT-4o with the code interpreter plugin to get initial answers. The spreadsheet files can be directly used as inputs, and the model can generate Python code to run in a code interpreter to generate the modified files or visual charts.

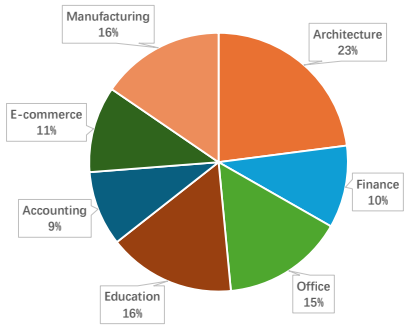

Figure 5: Domain distribution of all spreadsheets.

As shown in Figure 4, we ensure GPT-4o answer quality by first debugging its code. If the code cannot be executed without errors, the answer is considered incorrect. Second, We perform multiple inferences on each question-table pair, selecting the most frequent answer as a candidate. If all answers are different, the sample is viewed as invalid due to the inconsistency. Last, we have table analysis experts manually annotate the candidate answers, retaining the correct ones and correcting the wrong ones. We invited 10 experts with data analysis experience to annotate the dataset. Among them, native Chinese and English speakers each accounted for half of the group. Each answer is annotated twice, and Cohen's Kappa is 0.83, which indicates a high inter-annotator agreement. The answers of our dataset not only contain text but also contain Excel files and charts.

2.4 Dataset Statistic

Our MiMoTable benchmark consists of 1,719 (spreadsheet, question, answer) triplets originating from 428 different spreadsheets. In this subsection, we provide statistics from different dimensions to provide a more comprehensive understanding of our dataset.

Domains of Spreadsheets. As illustrated in Figure 5, our spreadsheets encompass seven domains in real-world applications.

Type and Difficulty of Tables. From Table 4, we can see that the table type of the collected spreadsheet is diverse, covering both simple, medium, and hard difficulty.

Difficulty	Ratio	Table Type	Num
Simple	33.6%	single file + single sheet + simple header	144
Medium	32.5%	single file + multiple sheets + simple header	30
		multiple files + simple header	37
		single file + single sheet + complicate header	72
Hard	33.9%	single file + multiple sheets + complicate header	63
		multiple files + complicate header	32
		multiple tables	50

Table 4: Distribution of table difficulty.

Meta Operations of Questions. Figure 6 shows the number of six meta operations in our benchmark questions. As the most basic operation of a table, Lookup is the most frequently occurring meta operation. More difficult meta operations such as Calculate and Reasoning also account for a relatively large proportion of our dataset, indicating that the questions of MiMoTable are diverse and comprehensive.

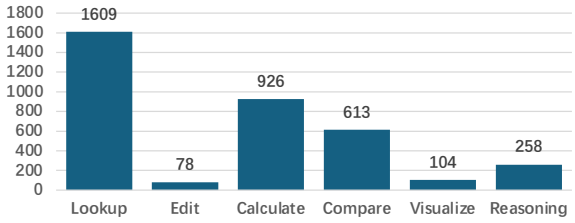

Figure 6: Distribution of meta operations.

Difficulty of Questions. To investigate the difficulty of questions, we calculate the difficulty score of each question according to the Equation 3. The score is in the range of [1, 4], so we divided the distribution of scores into three intervals, [1, 2), [2, 3) and [3, 4]. The specific values of question number and ratio are in Table 5.

Question Difficulty	Num	Ratio
[1, 2)	311	18.1%
[2, 3)	1150	66.9%
[3, 4]	258	15.0%

Table 5: Distribution of question difficulty.

3 Experiments and Results

Our experiments have two main goals: (1) to evaluate representative LLMs' performance on our dataset; and (2) to prove the effectiveness of proposed meta operations. This section presents the relevant experiments and findings.

3.1 Experimental Setup

Models. We conducted experiments on 16 selected LLMs, comprising open-source LLMs, closed-source LLMs, and tabular LLMs. The open-source LLMs we evaluated include Llama3.1[1], Llama3 (Dubey et al., 2024), Qwen2 (Yang et al., 2024), Qwen1.5 (Bai et al., 2023), Mistral (Jiang et al., 2023), DeepseekCoder (Guo et al., 2024), and Gemma (Mesnard et al., 2024). The closed-source LLMs are GPT-4o (OpenAI, 2023), Claude-3.5-Sonnet[2], and Gemini-1.5-Pro (Reid et al., 2024). We also evaluated Tablellama (Zhang et al., 2023), a tabular model fine-tuned specifically for various table tasks. However, most tabular models, such as Binder (Cheng et al., 2023), require inputs to be tables with known headers, which is not suitable for our benchmark.

Datasets. To demonstrate the generality and effectiveness of our meta operation, we conducted experiments on the newly proposed benchmark as well as two existing open-source benchmarks: WikiTableQuestion and WikiSQL. These widely-used TableQA benchmarks feature tables sourced from Wikipedia with simple headers.

Metrics. We used accuracy as the evaluation metric. Except for Tablellama, the predicted answers of other models in our experiments are all free-formed. We prompted GPT-4o to judge the correctness of the predicted answer based on the question and human-verified reference answer. Because a small portion of questions in MiMoTable are open-ended, we also instructed GPT-4o to give a score between 0-1 when it judges the question is open-ended.

Implementation Details. For all LLMs except Tablellama, we input table contents in markdown format. For GPT-4o, we also tested another popular

[1]https://ai.meta.com/blog/meta-llama-3-1/

[2]https://www.anthropic.com/news/claude-3-5-sonnet

Model	Overall	Language		Table Difficulty			Question Difficulty			Meta Operations			
		English	Chinese	Simple	Medium	Hard	[1, 2)	[2, 3)	[3, 4]	Lookup	Compare	Calculate	Reasoning
Claude-3.5-Sonnet	**77.4%**	**79.0%**	**76.2%**	81.3%	**75.5%**	**72.1%**	**89.0%**	**77.1%**	**63.3%**	**89.0%**	**79.7%**	**76.1%**	**63.3%**
GPT-4o-CI	69.2%	70.8%	68.1%	**81.7%**	67.1%	50.8%	81.0%	71.1%	45.8%	81.0%	73.1%	70.6%	45.8%
GPT-4o-TXT	69.0%	69.3%	68.8%	73.8%	66.2%	62.1%	85.1%	67.6%	53.9%	85.1%	73.1%	64.5%	53.9%
Gemini-1.5-Pro	60.2%	61.6%	59.1%	64.9%	57.4%	55.3%	86.1%	55.0%	47.6%	86.1%	60.3%	50.2%	47.6%
Llama-3.1-70B-Instruct	57.0%	56.6%	57.3%	64.0%	51.6%	51.3%	82.0%	52.1%	45.1%	82.0%	57.7%	48.8%	45.1%
Qwen2-72B-Instruct	55.7%	51.5%	58.8%	61.4%	52.6%	49.2%	80.4%	50.1%	46.9%	80.4%	56.3%	45.5%	46.9%
Llama-3-70B-Instruct	53.7%	52.3%	54.8%	60.1%	48.6%	48.5%	78.9%	47.8%	46.1%	78.9%	51.5%	44.4%	46.1%
Qwen1.5-72B-Chat	47.5%	46.1%	48.5%	51.6%	45.1%	43.2%	75.2%	41.2%	37.5%	75.2%	42.7%	40.1%	37.5%
Llama-3.1-8B-Instruct	44.1%	44.0%	44.2%	49.0%	43.3%	36.4%	70.0%	38.3%	34.9%	70.0%	41.0%	35.6%	34.9%
Qwen2-7B-Instruct	41.6%	40.5%	42.4%	45.6%	40.7%	35.5%	70.9%	34.1%	35.1%	70.9%	35.8%	32.4%	35.1%
Qwen1.5-14B-Chat	40.2%	38.6%	41.3%	44.1%	38.4%	35.3%	73.8%	32.4%	28.9%	73.8%	32.2%	30.9%	28.9%
Llama-3-8B-Instruct	39.9%	39.8%	40.0%	45.0%	37.0%	34.4%	68.4%	34.0%	27.5%	68.4%	36.1%	31.6%	27.5%
Mistral-7B-Instruct-v0.3	35.2%	35.2%	35.1%	40.5%	31.6%	30.1%	71.7%	25.8%	27.2%	71.7%	25.0%	23.4%	27.2%
Qwen1.5-7B-Chat	34.4%	33.6%	35.0%	40.1%	29.9%	29.7%	69.3%	25.1%	28.3%	69.3%	24.8%	22.9%	28.3%
Deepseek-Coder-7B-Instruct-v1.5	34.1%	33.9%	34.2%	38.4%	33.0%	27.6%	68.4%	25.1%	26.5%	68.4%	22.7%	24.2%	26.5%
Gemma-7B-Instruct	23.3%	20.6%	25.3%	28.7%	18.2%	19.9%	48.2%	15.7%	22.9%	48.2%	13.5%	14.9%	22.9%
Tablellama	21.1%	23.9%	19.1%	25.4%	20.0%	14.9%	45.4%	16.3%	9.9%	45.4%	15.5%	14.4%	9.9%

Table 6: Performance of LLMs on MiMoTable. GPT-4o-CI refers to the GPT-4o model with a code interpreter plugin.

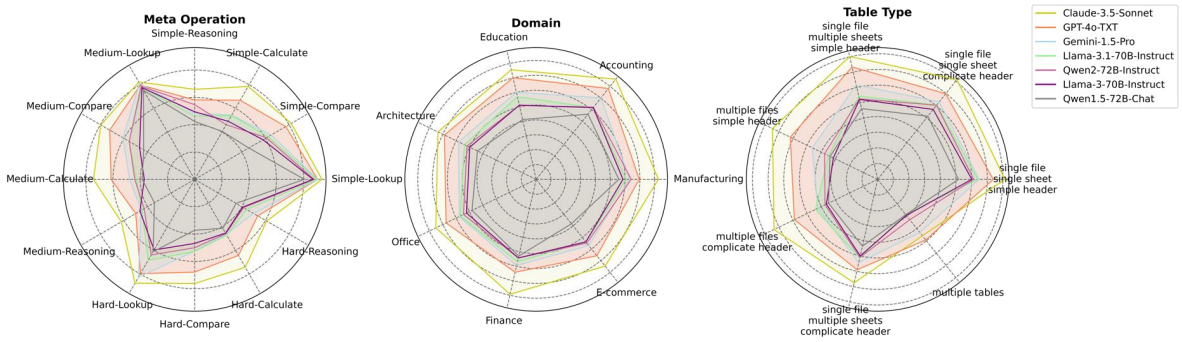

Figure 7: The performance of LLMs on MiMoTable respecting to different Meta Operations, Domains and Table Types.

approach in table reasoning, which is denoted as GPT-4o-CI in Table 6. This method uploads spreadsheets and generates Python code, executed via a code interpreter plugin, with results fed back for analysis. The format of MiMoTable and WikiTableQuestion is spreadsheet files, which can be directly as part of inputs to GPT-4o-CI. For WikiSQL, we first saved the non-file table content as Excel files and fed them to GPT-4o-CI. For Tablellama, we followed the prompt format specified in the original paper. For the existing benchmarks, we used GPT-4o to divide the questions according to the meta operations and then calculated the difficulty scores of the datasets based on Equation 6. We use the official default parameters for all models. More details can be found in the supplementary material.

3.2 Results and Analysis

Overall Performance. As shown in Table 6, we evaluate our proposed benchmark dataset using different LLMs. Because most experimental LLMs can not generate edited files and charts, we only infer the questions without meta operations Edit and Visualize for a fair comparison with GPT-4o-CI. As we can see, the best-performing model, Claude-3.5-Sonnet, achieved an overall performance of only 77.4% on our benchmark, highlighting that MiMoTable poses significant challenges for current LLMs. This underscores the need for further exploration to improve model performance on more realistic table data.

Analysis of different approaches. There are mainly two approaches to solving table reasoning

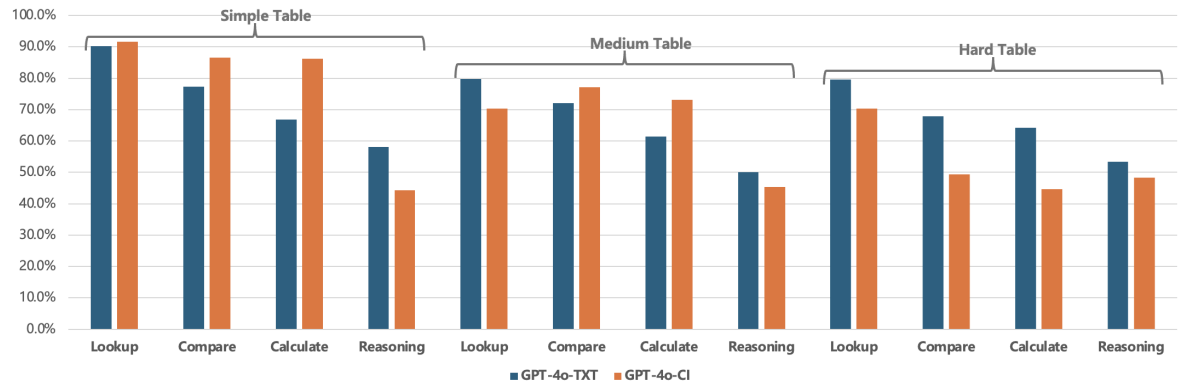

Figure 8: Performance on different data types.

problems of spreadsheets. One approach is to represent the spreadsheet content in text form and input it into the model to directly generate answers. The other is directly using the spreadsheet as input to write code, run in a sandbox, and conclude to solve the problem in a ReAct (Yao et al., 2023) way. As shown in Figure 8, we compare the performance of those two methods based on the GPT-4o model, where GPT-4o-TXT is the first text form approach and GPT-4o-CI is the second code-based approach. The GPT-4o-CI performs better than GPT-4o-TXT in Calculate and Compare when the table difficulty is simple and medium, while GPT-4o-TXT performs better in hard tables and in meta operations of Lookup and Reasoning. This reveals that the code-based approach has advantages in calculating only when the tables are not so hard, as hard tables can cause the model to be unable to write the correct code to locate the required data. The text-based approach is good at Lookup and Reasoning because the model can see the entire content of the table as long as the context window size is enough. The first radar chart in Figure 7 shows the results of more LLMs respecting to the different combinations of table difficulty and meta operations.

Capability of LLMs for table reasoning. According to the results of table difficulty in Table 6 and table types in Figure 7, most LLMs have struggled in medium and hard tables. We attribute the reasons to two factors, hierarchical header and multiple similar tables. As illustrated in Figure 9, although the hierarchical relations between table cells appear very clear in the original spreadsheets, they become much less intuitive when converted into text, which poses challenges for the LLMs to understand. Additionally, the tables in the spreadsheets with multiple sheets are usually very similar. LLMs need to comprehensively consider multiple similar tables to answer questions. In conclusion, the ability to understand complex table structures and multiple similar tables in table reasoning needs to be improved for current LLMs. We also observe differences in the performance of different models across languages. For example, Claude-3.5-Sonnet performs better in English than in Chinese, while Qwen2-72B-Instruct is the opposite. We believe this is due to the varying proportions of different languages used during the pretraining and SFT stages for each model.

Effectiveness of Meta Operations. Although WikiTableQuestion and WikiSQL are both datasets for TableQA tasks with tables that have simple headers, we found that the performance of the same LLMs on these two datasets varies significantly. For example, Llama3-70B achieves 82.0% accuracy on WikiSQL but only 66.7% accuracy on WikiTableQuestion. No objective metric exists to explain this discrepancy. By scoring the datasets according to our meta operations, we found that WikiTableQuestion is significantly more difficult than WikiSQL: the difficulty of WikiSQL is 1.5, while the difficulty of WikiTableQuestion is 2.0. The questions involving simple tables in MiMoTable, denoted as MiMoTable-Simple, have a difficulty of 2.2. We evaluated the performance of different LLMs on these three datasets—WikiSQL, WikiTableQuestion, and MiMoTable-Simple—and the results are shown in Figure 10. The x-axis represents the difficulty of the benchmarks graded by meta operations, and the y-axis shows the accuracy of the LLMs on these benchmarks. We found that as the difficulty score of the dataset increases, the model performance declines. This indicates that our proposed meta operations and difficulty scores are both generalizable and effective across different benchmarks.

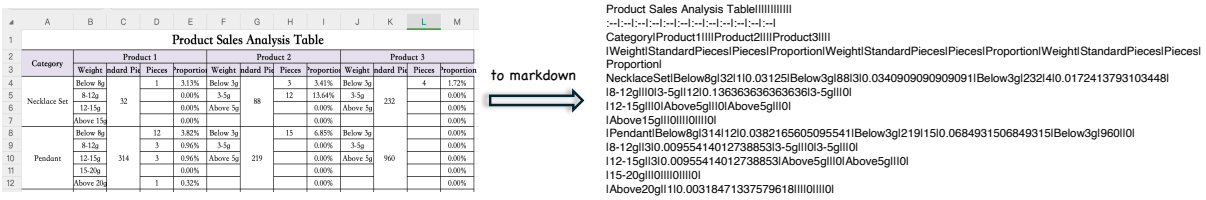

Figure 9: A spreadsheet to text with the markdown format.

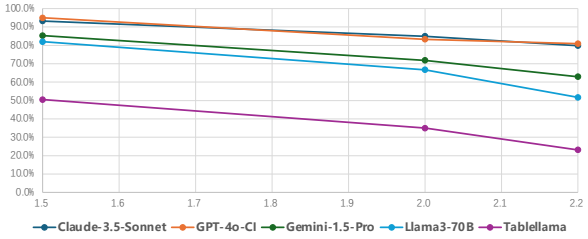

Figure 10: The relations between performance and the difficulty of benchmarks. The x-axis is the difficulty of benchmarks graded by meta operations. The y-axis is the accuracy of tested LLMs.

4 Related Work

The main tasks for table reasoning include four categories: TableQA, Table2Text, Table Manipulation, and Advanced Data Analysis (Lu et al., 2024). Researchers have proposed various table benchmarks for these tasks. TableQA is the most popular task, including benchmarks like WikiTableQuestions (Pasupat and Liang, 2015), WikiSQL (Zhong et al., 2017), FeTaQA (Nan et al., 2022), HybridQA (Chen et al., 2020), TATQA (Zhu et al., 2021), NQ-TABLES (Kwiatkowski et al., 2019), HybriDialogue (Nakamura et al., 2022), BIRD (Li et al., 2023b), Spider (Yu et al., 2018). The primary benchmark for Table2Text is ToTTo (Parikh et al., 2020). A high-quality Table Manipulation benchmark called WikiTableEdit is introduced in (Li et al., 2024). SPREADSHEETBENCH (Ma et al., 2024) is a challenging spreadsheet manipulation benchmark. For the Advanced Data Analysis task, the benchmarks DAEval (Hu et al., 2024) and DS-1000 (Lai et al., 2023) are proposed. Text2Analysis (He et al., 2024) is a recently introduced benchmark that includes both TableQA and Advanced Data Analysis tasks. Most existing table benchmarks feature simple table headers, but HiTab (Cheng et al., 2022) is a TableQA and Table2Text dataset based on hierarchical headers. AIT-QA (Katsis et al., 2022) is a dataset for TableQA with hierarchical headers specific to the airline industry.

Unlike the existing benchmarks, we propose a new benchmark, MiMoTable, the first benchmark with multi-scale spreadsheets that simultaneously covers four tasks: TableQA, Table2Text, Table Manipulation, and Advanced Data Analysis.

5 Conclusion

We propose a multi-scale spreadsheet benchmark with four tasks: TableQA, Table2Text, Table Manipulation, and Advanced Data Analysis, named MiMoTable. Experiments have shown that existing LLMs perform poorly on this benchmark, indicating that there is still significant room to improve in more realistic scenarios. For table reasoning, we also propose a new criterion for categorizing problems based on meta operations. Compared to task-based categorization, this criterion allows for a deeper and more accurate analysis of problems in table datasets. Our experiments demonstrate that the meta operations are general and effective.

6 Limitations

When validating the effectiveness of meta operations, we do not perform Supervised Fine-Tuning (SFT) on the models. Future work could examine the role and effect of each type of operation through SFT. Meanwhile, we used the same prompt for evaluating all models except Tablellama, without optimizing or adapting prompts for different models. Regarding hyperparameters for different models, we used the officially recommended default parameters and do not adjust different hyperparameters for different models. Additionally, inspired by work in other fields (Song et al., 2024a,b), developing long-context table reasoning benchmarks and studying in-context learning for table reasoning are valuable directions for further exploration.

Acknowledgments

We thank the three anonymous reviewers for carefully reading our paper and their insightful comments and suggestions.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *CoRR*, abs/2309.16609.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. 2020. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1026–1036. Association for Computational Linguistics.

Zhoujun Cheng, Haoyu Dong, Zhiruo Wang, Ran Jia, Jiaqi Guo, Yan Gao, Shi Han, Jian-Guang Lou, and Dongmei Zhang. 2022. Hitab: A hierarchical table dataset for question answering and natural language generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1094–1110. Association for Computational Linguistics.

Zhoujun Cheng, Tianbao Xie, Peng Shi, Chengzu Li, Rahul Nadkarni, Yushi Hu, Caiming Xiong, Dragomir Radev, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. 2023. Binding language models in symbolic languages. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, 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, 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, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, 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, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Y. Wu, Y. K. Li, Fuli Luo, Yingfei Xiong, and Wenfeng Liang. 2024. Deepseek-coder: When the large language model meets programming - the rise of code intelligence. *CoRR*, abs/2401.14196.

Xinyi He, Mengyu Zhou, Xinrun Xu, Xiaojun Ma, Rui Ding, Lun Du, Yan Gao, Ran Jia, Xu Chen, Shi Han, Zejian Yuan, and Dongmei Zhang. 2024. Text2analysis: A benchmark of table question answering with advanced data analysis and unclear queries. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 18206–18215. AAAI Press.

Xueyu Hu, Ziyu Zhao, Shuang Wei, Ziwei Chai, Guoyin Wang, Xuwu Wang, Jing Su, Jingjing Xu, Ming Zhu, Yao Cheng, Jianbo Yuan, Kun Kuang, Yang Yang, Hongxia Yang, and Fei Wu. 2024. Infiagent-dabench: Evaluating agents on data analysis tasks. *CoRR*, abs/2401.05507.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Yannis Katsis, Saneem A. Chemmengath, Vishwajeet Kumar, Samarth Bharadwaj, Mustafa Canim, Michael R. Glass, Alfio Gliozzo, Feifei Pan, Jaydeep Sen, Karthik Sankaranarayanan, and Soumen Chakrabarti. 2022. AIT-QA: question answering dataset over complex tables in the airline industry. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track, NAACL 2022, Hybrid: Seattle, Washington, USA + Online, July 10-15, 2022*, pages 305–314. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti,

Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7:452–466.

Yuhang Lai, Chengxi Li, Yiming Wang, Tianyi Zhang, Ruiqi Zhong, Luke Zettlemoyer, Wen-Tau Yih, Daniel Fried, Sida I. Wang, and Tao Yu. 2023. DS-1000: A natural and reliable benchmark for data science code generation. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 18319–18345. PMLR.

Hongxin Li, Jingran Su, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. 2023a. Sheetcopilot: Bringing software productivity to the next level through large language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, Xuanhe Zhou, Chenhao Ma, Guoliang Li, Kevin Chen-Chuan Chang, Fei Huang, Reynold Cheng, and Yongbin Li. 2023b. Can LLM already serve as A database interface? A big bench for large-scale database grounded text-to-sqls. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Zheng Li, Xiang Chen, and Xiaojun Wan. 2024. Wikitableedit: A benchmark for table editing by natural language instruction. *CoRR*, abs/2403.02962.

Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2022. TAPEX: table pre-training via learning a neural SQL executor. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Weizheng Lu, Jiaming Zhang, Jing Zhang, and Yueguo Chen. 2024. Large language model for table processing: A survey. *CoRR*, abs/2402.05121.

Zeyao Ma, Bohan Zhang, Jing Zhang, Jifan Yu, Xiaokang Zhang, Xiaohan Zhang, Sijia Luo, Xi Wang, and Jie Tang. 2024. Spreadsheetbench: Towards challenging real world spreadsheet manipulation. *CoRR*, abs/2406.14991.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Cristian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, and et al. 2024. Gemma: Open models based on gemini research and technology. *CoRR*, abs/2403.08295.

Kai Nakamura, Sharon Levy, Yi-Lin Tuan, Wenhu Chen, and William Yang Wang. 2022. Hybridialogue: An information-seeking dialogue dataset grounded on tabular and textual data. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 481–492. Association for Computational Linguistics.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryscinski, Hailey Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, and Dragomir R. Radev. 2022. Fetaqa: Free-form table question answering. *Trans. Assoc. Comput. Linguistics*, 10:35–49.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Ankur P. Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. Totto: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1173–1186. Association for Computational Linguistics.

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 of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1470–1480. The Association for Computer Linguistics.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy P. Lillicrap, Jean-Baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew M. Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, James Molloy, Jilin Chen, Michael Isard, Paul Barham, Tom Hennigan, Ross McIlroy, Melvin Johnson, Johan Schalkwyk, Eli Collins, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Clemens Meyer, Gregory Thornton,

Zhen Yang, Henryk Michalewski, Zaheer Abbas, Nathan Schucher, Ankesh Anand, Richard Ives, James Keeling, Karel Lenc, Salem Haykal, Siamak Shakeri, Pranav Shyam, Aakanksha Chowdhery, Roman Ring, Stephen Spencer, Eren Sezener, and et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *CoRR*, abs/2403.05530.

Mingyang Song, Mao Zheng, and Xuan Luo. 2024a. Can many-shot in-context learning help llms as evaluators? a preliminary empirical study. *Preprint*, arXiv:2406.11629.

Mingyang Song, Mao Zheng, and Xuan Luo. 2024b. Counting-stars: A multi-evidence, position-aware, and scalable benchmark for evaluating long-context large language models. *Preprint*, arXiv:2403.11802.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. Table meets LLM: can large language models understand structured table data? A benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining, WSDM 2024, Merida, Mexico, March 4-8, 2024*, pages 645–654. ACM.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024. Qwen2 technical report. *CoRR*, abs/2407.10671.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir R. Radev. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3911–3921. Association for Computational Linguistics.

Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. 2023. Tablellama: Towards open large generalist models for tables. *CoRR*, abs/2311.09206.

Xuanliang Zhang, Dingzirui Wang, Longxu Dou, Qingfu Zhu, and Wanxiang Che. 2024. A survey of table reasoning with large language models. *CoRR*, abs/2402.08259.

Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning. *CoRR*, abs/1709.00103.

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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3277–3287. Association for Computational Linguistics.

A Appendix

A.1 Data Statistic of The Language

Table 7 shows the data statistics of the proposed benchmark under different languages, including table number, question number, and the average of question difficulty.

	Table Number	Question Number	Question Difficulty
Overall	428	1719	2.2
English	182	671	2.2
Chinese	246	1048	2.2

Table 7: Data Statistics of Different Languages

A.2 Used Prompts

Table 8, Table 9, and Table 10 show the designed prompts for meta operations classification, model inference, and performance evaluation in this paper.

You are a spreadsheet question classification expert. Given a user's question about an Excel spreadsheet, classify the question according to the requirements and output it in the specified format.

<Requirements>
The following operation classification already exists, presented in the format of operation name: operation description. If the user's question can be classified as some of the operations, output the operation names. One question can be classified into multiple operations.
Lookup: Locate the position of the specific target.
Edit: Modify, delete, or add to a table.
Calculate: The numerical computation, sum, avg, max, etc.
Compare: Compare two or more targets in a table.
Visualize: Show in chart. Reasoning: Inferring information from the table content that is not explicitly included.
</Requirements>
<Output Format>
operation name 1, operation name 2, ...
</Output Format>

<Question>
what country hosted the most tournaments?
</Question>
Lookup, Calculate, Compare

<Question>
QUESTION TO BE CLASSIFIED
</Question>

Table 8: Prompt for Meta Operation Classification

Below is the table content in markdown, please answer the question according to the table content.

<Table>
<Table Name>
SPREADSHEET FILE NAME
</Table Name>
<Table Content>
<Sheet>
<Sheet Name>
SHEET NAME
</Sheet Name>
<Sheet Content>
SHEET CONTENT IN MARKDOWN
</Sheet Content>
</Sheet>
</Table Content>
</Table>

<Question>
THE QUESTION TO BE ASKED
</Question>

Table 9: Prompt for Model Inference

For the following questions, given the correct answer, determine whether the candidate's answer is correct. If it is correct, output "Correct"; if it is incorrect, output "Incorrect"; if it is uncertain whether it is correct, output "Uncertain". As long as the candidate's answer contains the key information that can correctly answer the question, it is considered correct. If the question is open-ended, give a score between 0-1 according to the correct answer. Do not output any other content.

<Question>
THE QUESTION
</Question>
<Correct Answer>
THE CORRECT ANSWER
</Correct Answer>
<Candidate answer>
THE CANDIDATE ANSWER
</Candidate answer>

Table 10: Prompt for Performance Evaluation