Tables as Images? Exploring the Strengths and Limitations of LLMs on Multimodal Representations of Tabular Data

Anonymous ACL submission

Abstract

In this paper, we investigate the effectiveness of various LLMs in interpreting tabular data through different prompting strategies and data formats. Our analysis extends across six benchmarks for table-related tasks such as question-answering and fact-checking. We introduce for the first time the assessment of LLMs' performance on image-based table representations. Specifically, we compare five text-based and three image-based table representations, demonstrating the influence of representation and prompting on LLM performance. Our study provides insights into the effective use of LLMs on table-related tasks.

1 Introduction

Recent years have witnessed an explosion of Large Language Models (LLMs), with impressive performances on various Natural Language Processing (NLP) tasks (Brown et al., 2020; Touvron et al., 2023; Team et al., 2023). Research to date has examined the performance of LLMs for various aspects and abilities (Bang et al., 2023b; Bubeck et al., 2023; Akter et al., 2023), but their effectiveness on structured data such as tables is less explored.

Unlike unstructured text, tables are systematically organized structures of a large amount of information. This characteristic makes tabular data serve as the foundations for numerous applications, including medical diagnostics, virtual personal assistants, customer relationship management (Hemphill et al., 1990; Dahl et al., 1994; Akhtar et al., 2022; Xie et al., 2022), etc.

The evaluation of LLMs on processing tabular data involves many challenges. First, there are many ways to represent the information in tables. If we represent the table in pure text, we may use naive linearization or insert brackets to better represent table structures. Meanwhile, emerging multimodal LLMs like GPT-4 (Achiam et al., 2023) and Gemini (Team et al., 2023) offer image-based approaches, where we can pass the table as images to the LLMs. In such cases, visual cues like color highlighting in tables can influence outcomes. Second, diverse prompting methods for text may also apply to tabular data, which can yield varied results (Wei et al., 2022). Furthermore, the tasks involving tabular data are diverse, including table fact-checking (Chen et al., 2019) and table question answering (Pasupat and Liang, 2015), and table-to-text generation (Novikova et al., 2017), etc.

In this paper, we systematically evaluate model performance on tabular data for both textual LLMs and multi-modal LLMs. Specifically, we investigate several research questions, including the effectiveness of image-based representation of tabular data and how different text-based or image-based prompt methods affect LLMs' performance on table-related tasks. Our findings include:

- LLMs maintain decent performance when we use image-based table representations. Sometimes, image-based table representations can make LLMs perform better.
- There are nuances in the prompting design for table-related tasks, revealed by our comparisons of various prompting methods for text- and image-based table representations.

To the best of our knowledge, we are the first to study how LLMs perform with image-based table representations. We believe this paper draws new insights into optimizing table-based information processing.

2 Related Work

Table-Related Tasks. Tasks involving structured data have attracted interest in various tasks from

[*] Contributed equally to this work.

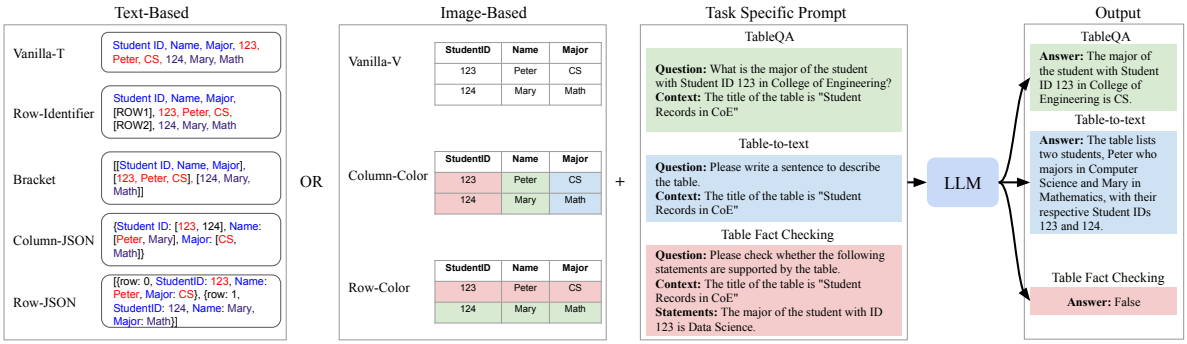

Figure 1: Concept diagram. In this paper, we study differences in table representations. For each example, we prompt LLMs with the question and the context information, as well as one of the table representations.

diverse communities (Deng et al., 2020; Chen et al., 2021a, 2022; Deng et al., 2022), among which there is a huge focus on tabular data (Yin et al., 2020; Herzig et al., 2020).

Researchers have investigated various ways to encode tabular data. Hwang et al. (2019); Liu et al. (2021) linearize the table content. Others employ model-specific techniques such as adapting the attention mechanism to better align transformer-based models with the tabular data (Zhang et al., 2020; Yang et al., 2022) or designing hierarchical encoding to capture the table structure (Wang et al., 2021), fine-tuning LLMs on tabular data (Zha et al., 2023), etc. In contrast, our work focuses on exploring various table representations and prompts LLMs directly.

Prompting LLMs. Researchers have prompted LLMs to evaluate LLMs' performance on traditional NLP tasks (Bang et al., 2023a) as well as on various complex reasoning tasks (Jin et al., 2022; Wu et al., 2023). On the contrary, to the best of our knowledge, few works have prompted these LLMs on tasks involving tabular data.

For closed-source LLMs, researchers adopt hard prompts to manually craft text prompts with discrete tokens (Qiao et al., 2022; Bahng et al., 2022; Liu et al., 2023). Wei et al. (2022) develop chain-of-though prompting, Xu et al. (2023a) develop expert prompting. In our work, we include the comparison between vanilla, chain-of-thought, and expert prompting for LLMs on table-related tasks.

3 Experiment Setups

3.1 Experimented LLMs

Table 1 describes the LLMs we use for our experiments. We use closed-source models such as GPT-3.5 and GPT-4 (Brown et al., 2020; Ouyang et al., 2022), and Gemini (Team et al., 2023). We note that GPT-4 and Gemini are multimodal models, which can take tables as images. For open-source models, we use the chat models from LLaMa-2 (Touvron et al., 2023) families from the 7 billion to the 70 billion parameter version as they are claimed to perform on par with closed-source models like ChatGPT.*

Models	# P(B)	🔓/🔒	+V?	Company
LLaMa-2	7/13/70	🔓	✗	Meta
GPT-3.5	–	🔒	✗	OpenAI
GPT-4	–	🔒	✓	OpenAI
Gemini$_{\text{pro}}$	–	🔒	✓	Google

Table 1: Comparison of LLMs used in our experiments. "# P" represents the number of parameters in billions (B). Note that we do not include the number of parameters for the closed-source models as there are no official documents revealing this information. "🔓/🔒" indicates whether the LLM is open-source (🔓) or closed-source (🔒). "+V?" indicates whether the visual input is allowed for the LLM. "Company" indicates which company the LLM is from.

3.2 Prompting Strategies

We explore two ways to represent tables in the prompt, **Text-Based** and **Image-Based**.

Text-Based. Apart from the information contained in the cells of tables, the structure of the table maintains information such as what cell values are in the same row or column, and what cell values correspond to a particular column. Therefore, we explore various ways to incorporate such structure information into the text prompt.

- **Vanilla-T** lists column names followed by cell values in each row sequentially, an approach

*https://huggingface.co/meta-llama/Llama-2-70b-chat

Method Name	Table Representation
Vanilla-T	$c_1, c_2, \cdots, c_n, v_{(1,1)}, v_{(1,2)}, \cdots, v_{(1,n)}, v_{(2,1)}, v_{(2,2)}, \cdots, v_{(2,n)}, \cdots, v_{(m,1)}, v_{(m,2)}, \cdots, v_{(m,n)}$.
Row-Identifier	$c_1, c_2, \cdots, c_n$, [ROW1] $v_{(1,1)}, v_{(1,2)}, \cdots, v_{(1,n)}$, [ROW2] $v_{(2,1)}, v_{(2,2)}, \cdots, v_{(2,n)}, \cdots$, [ROW$m$] $v_{(m,1)}, v_{(m,2)}, \cdots, v_{(m,n)}$.
Bracket	$[[c_1, c_2, \cdots, c_n], [v_{(1,1)}, v_{(1,2)}, \cdots, v_{(1,n)}], [v_{(2,1)}, v_{(2,2)}, \cdots, v_{(2,n)}], \cdots, [v_{(m,1)}, v_{(m,2)}, \cdots, v_{(m,n)}]]$.
Column-JSON	$\{c_1: [v_{(1,1)}, v_{(2,1)}, \cdots, v_{(m,1)}], c_2: [v_{(1,2)}, v_{(2,2)}, \cdots, v_{(m,2)}], \cdots, c_n: [v_{(1,n)}, v_{(2,n)}, \cdots, v_{(m,n)}]\}$.
Row-JSON	[{ Row: 1, $c_1: v_{(1,1)}, c_2: v_{(1,2)}, \cdots, c_n: v_{(1,n)}$}, { Row: 2, $c_1: v_{(2,1)}, c_2: v_{(2,2)}, \cdots, c_n: v_{(2,n)}$ }, $\cdots$, { Row: m, $c_1: v_{(m,1)}, c_2: v_{(m,2)}, \cdots, c_n: v_{(m,n)}$ }].

Table 2: Text-based table representation examples. We construct the examples assuming a table of m rows and n columns, where c_i denotes the column name of column i and $v_{(i,j)}$ denotes the cell value at row i and column j. We use colored text to indicate different rows in the table to assist readers.

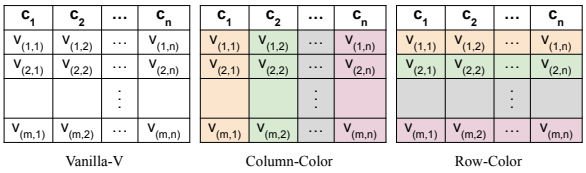

Figure 2: Image-based table representation examples. We construct these examples based on the same table described in Table 2.

adopted in various prior works (Hwang et al., 2019; Liu et al., 2021).

- **Row-Identifier** adds an identifier as the prefix for each row to distinguish different rows in the linearized table sequence.
- **Bracket** encloses the column names and their values in brackets to distinguish each row.
- **Column-JSON** represents the table in JSON format, where column names are the keys that map to the list of cell values corresponding to that column.
- **Row-JSON** represents each row as a JSON object, within which the column names and their corresponding cell values are represented as key-value pairs.

Table 2 shows examples of these text-based table representations.

Image-Based. Alternatively, we can pass the table as an image to the recent multimodal LLMs such as GPT-4 and Gemini. In this way, LLMs would "view" the table in a similar way as how we human beings view the table. We explore various table-highlighting methods as different visual cues may influence the model outcomes as shown by Shtedritski et al. (2023) who study how highlighting can influence CLIP model (Radford et al., 2021)'s performance on vision and language tasks. We pass these images of the table to LLMs.

- **Vanilla-V** feeds the table image without any colors or highlighting to LLMs.
- **Column-Color** uses a single color for each table column. Therefore, the LLM may easily distinguish columns as cells in the same column are annotated by the same color, whereas different colors annotate cells from different columns.
- **Row-Color** uses a single color for each row in the table. The same color annotates cells in the same row, whereas different colors annotate cells in different rows.

Figure 2 show examples for these image-based table representations.

On top of different methods to represent tables, we test the vanilla prompting, chain-of-thought prompting (Wei et al., 2022), and expert prompting (Xu et al., 2023a) by adding "let's pretend you are an expert in reading and understanding tables" to the prompt. Appendix C provides an example for each table representation and prompting method.

3.3 Datasets

We make use of six previously introduced datasets that cover different table sources such as Wikipedia and financial reports, examine model abilities such as information extraction and arithmetic reasoning, and cover table-related tasks such as table question answering, table fact-checking, and table-to-text generation. Table 3 provides information for each dataset we use. Considering the limited access to LLMs' APIs and the scale of the comparison, we randomly select 100 examples from the test set for each of these datasets to conduct our analysis.

3.4 Metrics

Following Pasupat and Liang (2015); Chen et al. (2019, 2020, 2021b), we compute accuracy scores on WikiTQ, TabFact, LogicNLG, FinQA.

Task Family	Name	Domain	Input	Output	Metrics
Table QA	WikiTQ (Pasupat and Liang, 2015)	Wikipedia	Table	Text	Acc
	FinQA (Chen et al., 2021b)	Finance	Table + Text	Text	Acc
Table Fact Checking	TabFact (Chen et al., 2019)	Wikipedia	Table	Boolean	Acc
Table-to-text	E2E (Novikova et al., 2017)	Restaurants	Table	Text	ROUGE, Human
	ToTTo (Parikh et al., 2020)	Wikipedia	Table + Text	Text	ROUGE
	LogicNLG (Chen et al., 2020)	Wikipedia	Table + Text	Entity	Acc

Table 3: Dataset descriptions. For Input, we refer to the input information other than the question, the statement for fact-checking, or the statement that requires the model to describe the table content.

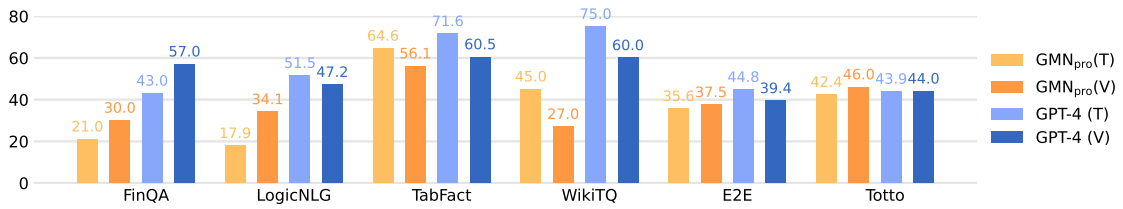

Figure 3: Performance comparison between passing the text versus image representations of tables to GPT-4 and Gemini$_{Pro}$ across FinQA, LogicNLG, TabFact, and WikiTQ by accuracy, and E2E and ToTTo by ROUGE-L scores. We feed the linearized table (Vanilla-T) as the text-based representation, and the original table image (Vanilla-V) as the image-based representation to these LLMs.

We adopt the automatic ROUGE evaluation for table-to-text generation datasets ToTTo and E2E. In addition, the authors manually investigate the generation quality on the E2E dataset by whether the generation encapsulates the table information without any additional information that cannot be inferred from the table.

4 Research Questions

Using the setup described previously, we can now seek answers to several research questions concerning the use of LLMs for tabular data.

RQ1. Are image-based representations of tabular data effective?

Test: We compare using the linearized table representation (Vanilla-T in text-based table representation) and the table image (Vanilla-V in image-based table representation) as the input for both GPT-4 model and Gemini$_{pro}$. We use vanilla prompting in this comparison and plot Figure 3. We report the results for other prompting methods in Figures 9 and 10 in Appendix A.1.

TL;DR Answer: Yes.

Full Answer: Figure 3 shows that in most cases, LLMs perform comparably if we represent tables as images versus text. On datasets such as FinQA, passing image representation of tables to Gemini$_{pro}$ and GPT-4 outperform passing text representations of the tables significantly. As FinQA focuses on financial question answering with long context and many numerical relations, we hypothesize that *representing tables as images can help LLMs in complex reasoning*. Since these multimodal LLMs have a strong capability over visual input (Yang et al., 2023), representing tables as images may reduce the cognitive load for LLMs to parse and understand dense text. This is especially beneficial when the context involves long passages of text that may also contain numerous numerical relations. As shown in Figure 4, since the context is long (around 416 English words, approximately 556 tokens for GPT models) and involves various numerical relations, GPT-4 ignores the relevant clues in text when we pass text representation of the table. In contrast, when we pass the table image, GPT-4 can effectively leverage information from both the text and visual modality for its reasoning process.

On WikiTQ and TabFact, both Gemini$_{pro}$ and GPT-4 perform better with the text than the image representation of the table significantly. We notice that both datasets are sourced from Wikipedia and the texts from Wikipedia are commonly used to pre-train LLMs (Brown et al., 2020; Touvron et al., 2023). GPT-4 and Gemini$_{pro}$ may have encountered these tables in their pre-training phase in the text format rather than the image format, leading

Question: What percentage of the intangible assets is related to the license of the realtor.com ae trademark?
Context: …the license of the realtor.com ae trademark , which has a fair value of approximately $116 million…

Cash	$108
Other current assets	28
Intangible assets	216
…	…

Gold: 0.53704
GPT-4 (T): The text does not provide information… ✗
GPT-4 (V): …(Trademark value / Total intangible assets) * 100 = ($116 million / $216 million) * 100 = 53.7037% ✓

Figure 4: An example from FinQA. We highlight the relevant parts from the context and the table and omit irrelevant parts to help readers. We feed the linearized table (Vanilla-T) as the text-based representation (GPT-4 (T)), and the original table image (Vanilla-V) as the image-based representation to GPT-4 (GPT-4 (V)).

	GPT 3.5	GPT 4	GMN_{pro}	LLaMa-2 7B	LLaMa-2 13B	LLaMa-2 70B
Vanilla-T						
V	52.5	60.3	37.1	28.8	35.3	42.7
E	51.0	**63.8**	39.5	29.0	35.1	46.7
CoT	55.2	62.6	53.5	32.1	37.6	48.3
Brackt						
V	50.9	60.1	38.4	28.4	36.6	42.2
E	47.9	62.8	39.5	28.1	34.5	45.8
CoT	51.4	61.9	57.3	34.2	39.3	50.0

Table 4: For text-based table representations, averaged accuracy scores across FinQA, LogicNLG, TabFact, and WikiTQ for different LLMs. "GMN_{pro}" represents $Gemini_{pro}$ model, "V", "E", and "CoT" represent vanilla, expert and chain-of-thought prompting, respectively.

to the performance disparity between text and image representation of tables for both $Gemini_{pro}$ and GPT-4 on WikiTQ and TabFact.

RQ2. How do different text-based prompt methods affect LLMs' performance on table-related tasks?

Test: We compare the five text-based table representations introduced in Section 3.2. On top of the five representations, we also compare how vanilla, chain-of-thought, and expert prompting affect the model performance. We conduct the comparison using all six LLMs in Section 3.1 and average their accuracy scores across FinQA, LogicNLG, TabFact, and WikiTQ. Appendix A.2 reports LLMs' performance on E2E and ToTTo datasets.

TL;DR Answer 2.1: Expert prompting works the best when the LLM is an "expert".

Full Answer 2.1: With respect to vanilla, CoT, and expert prompting, for GPT-4, we note that expert prompting outperforms the other two prompting methods consistently. For instance, for the vanilla linearized table representations (Vanilla-T), expert prompting outperforms the CoT and the vanilla prompting method by 1.2% and 3.5%, respectively (Table 4). In contrast, CoT prompting instead of expert prompting leads to the best performance for all other models. For instance, for GPT-3.5 with Vanilla-T table representation, CoT prompting outperforms vanilla and expert prompting by 2.7% and 4.2% (Table 4).

On the other hand, GPT-4 outperforms all other models, as the best average score GPT-4 achieves is 63.8%, compared to 55.2% by GPT-3.5 and 50.0% by LLaMa-2-70B. We suspect that because of GPT-4's "expertise" on these tasks, expert prompting can further enhance its reasoning ability as GPT-4 can "pretend they are an expert in reading and understanding tables". In contrast, expert prompting may not fit the less capable LLMs as they may not "pretend an expert" well.

TL;DR Answer 2.2: CoT prompting can sometimes boost up the performance significantly.

Full Answer 2.2: We notice that CoT prompting significantly improves $Gemini_{pro}$'s performance from 38.4% to 57.3% using the bracket table representation (Table 4), which outperforms the best performance 55.2% by GPT-3.5. This suggests that proper prompting can make a big difference in LLMs' performance and unleash the potential within the LLM. On the other hand, it underscores the complexity of LLMs' evaluation and the importance of prompt engineering, as we may underestimate an LLMs' performance because of an improper prompt.

TL;DR Answer 2.3: Bracket representation can help LLMs better understand tables.

Full Answer 2.3: Compared to linearizing tables directly (Vanilla-T), adding brackets to distinguish rows in the table boosts up model performances for $Gemini_{pro}$ and different versions of LLaMa-2 models (Table 4).

Figure 5 shows a WikiTQ example from $Gemini_{pro}$, where the vanilla prompting fails to count the number of "1st" place. We suspect that the simple linearized table representation does not have a clear boundary between rows, which may

Question: How many times did Salvatore Bettiol win first place across competitions?

Year		Position
1986	…	1st
1987		1st
1991		6th

Gold: 2
Vanilla-T: three times. ✗
Row-Identifier: 3 times. ✗
Bracket: 2 ✓

Figure 5: An example from WikiTQ. We use $\text{Gemini}_{\text{pro}}$ with vanilla prompting and show its prediction when we use the linearized table representation (Vanilla-T), insert "Row-Identifier" or "Bracket" in the representation.

	GPT-4			$\text{Gemini}_{\text{pro}}$		
	V	E	CoT	V	E	CoT
VV	56.2	54.9	57.8	36.8	37.2	46.0
CC	53.3	52.8	58.0	37.1	37.8	45.1
RC	51.8	51.6	**60.2**	39.4	38.7	46.2

Table 5: For image-based table representations, averaged accuracy scores across FinQA, LogicNLG, TabFact, and WikiTQ for GPT-4 and $\text{Gemini}_{\text{pro}}$. For the headers, "V", "E", and "CoT" represent vanilla, expert, and chain-of-thought prompting, respectively. For the row names, "VV", "CC', and "RC" represent Vanilla-V, Column-Color, and Row-Color, respectively.

lead to confusion or misinterpretation of data relationships. In addition, adding the row identifier in the sequence does not help while the LLM answers correctly with the bracket representation. We conjecture that LLMs may be familiar with brackets from their pre-training exposure. Since brackets are fundamental components of many programming languages, and Github which contains rich code is often used as a source for pre-training corpora (Touvron et al., 2023), LLMs may have acquired proficiency in recognizing and interpreting bracketed structures.

TL;DR Answer 2.4: Different table representations do not affect the performance of GPT models much.

Full Answer 2.4: Even without any sophisticated prompting methods, the GPT-3.5 and GPT-4 achieve a decent performance (52.5% and 60.3% respectively using the vanilla prompting and linearized table representation from Table 4), demonstrating their strong table understanding abilities. In such cases, brackets or other kinds of table representations may add extra "workload" to the model, which dilutes the models' attention to the original table content and thus leads to worse performance.

Question: How many games did the team score at least 30 points?

Week		Score
4	…	34-6
5		38-12
6		45-0
10		30-9

Gold: 4
Vanilla-V: 3 games. ✗
Row-Color: 4 games. ✓

Figure 6: An example from WikiTQ. We use $\text{Gemini}_{\text{pro}}$ with vanilla prompting and show its prediction when we use the original table image (Vanilla-V) and the table image that uses different colors to distinguish rows in the table (Row-Color).

RQ3. How do different image-based prompt methods affect LLMs' performance on table-related tasks?

Test: We test the three image-based table representations in Section 3.2 together with vanilla, chain-of-thought, and expert prompting. We test the $\text{Gemini}_{\text{pro}}$ and GPT-4 model which can take images as the input. We average the accuracy scores across FinQA, LogicNLG, TabFact, and WikiTQ. Appendix A.3 reports LLMs' performance on E2E and ToTTo datasets.

TL;DR Answer 3.1: CoT prompting helps LLMs reason over images of the table.

Full Answer 3.1: In Table 5, we observe that chain-of-thought prompting helps multimodal LLMs in all image-based table representations. For instance, when using different colors to distinguish rows in the table (Row-Color), the average accuracy score for GPT-4 improves from 51.8% by vanilla prompting to 60.2% by chain-of-thought prompting. By explicitly outlining the reasoning process, chain-of-thought prompting may help LLMs better understand the context and relationships between different rows and columns in the table, therefore better aligning this visual information with the question text. Such consistent performance improvements suggest that chain-of-thought prompting may enhance information fusion across the text and vision modality.

TL;DR Answer 3.2: Distinguishing rows may lead to better performance for LLMs to reason over images of the table.

Full Answer 3.2: In Table 5, under CoT prompting, GPT4 performs slightly better when using colors to distinguish different rows, which also yields the overall best performance using images of the table. In contrast, under CoT prompting, using

Rep	Cues	GPT 3.5	GPT 4	GMN_{pro}	LLaMa-2 7B	LLaMa-2 13B	LLaMa-2 70B
T	N/A	34	43	21	10	20	41
T	T	30	51	25	14	16	37
V	N/A	-	57	30	-	-	-
V	T	-	58	34	-	-	-
V	V	-	57	28	-	-	-
V	V+T	-	**61**	38	-	-	-

Table 6: Accuracy scores of LLMs on FinQA. We use vanilla prompting across experiments in this table. GMN_{pro} represents $Gemini_{pro}$ model. We denote text and image-based table representations as "T" and "V" in the "Rep" column, respectively. The "Cues" column indicates how we highlight the relevant cells, where "N/A" indicates no information about relevant cells, "T" indicates referring to relevant cells in the text, "V" indicates highlighting relevant cells on the table image, "V + T" indicates both highlighting relevant cells on the table image and referring to them in the text.

colors to distinguish columns yields similar performance to vanilla image (58.0% to 57.8% for GPT-4 and 45.1% to 46.0% for $Gemini_{pro}$), suggesting that these advanced LLMs may not capture row information as well as column information.

Figure 6 shows a WikiTQ example with $Gemini_{pro}$ model's predictions. Since the question asks about the number of games, it requires the model to count how many rows satisfy such a condition. Using colors to distinguish rows may help models visually segment and categorize the data. This visual differentiation may act as a cognitive aid, which reduces the complexity of parsing and interpreting the tabular data.

TL;DR Answer 3.3: The more capable LLM does not necessarily benefit more from the colored images.

Full Answer 3.3: In addition, if we use the vanilla prompt, the different coloring methods may even hurt the performance of GPT-4 (for GPT-4, coloring rows with different colors yields 51.8% compared to 56.2% without adding any color), but helpful for $Gemini_{pro}$ (for $Gemini_{pro}$, coloring rows with different colors yields 39.4% compared to 36.8% without adding any color). This suggests that the effectiveness of how different LLMs can leverage colored images varies, and does not depend on the model's overall performance.

RQ 4. Does highlighting relevant cells yield a better performance?

Test: We test all six LLMs in Section 3.1 on FinQA which provides relevant cells in the table for each instance. We refer to the relevant cells by adding "Please pay attention to the highlighted cells: (row index, column index, cell value)" in the text prompt, or mark them on the table image directly. Appendix C provides our prompt examples. We use vanilla prompting in this comparison.

TL;DR Answer: Yes.

Full Answer: In Table 6, we notice that in most cases, referring LLMs to specific cells helps LLMs better attend to them, thereby helping LLMs reason over the example. However, LLMs' performance may get hurt when we refer to the relevant cells through text such as LLaMa-2-13B and 70B. This may be due to the inherent limitations of textual descriptions for conveying spatial or relational information. In order to relate the mentioned cells in the text, the model needs to figure out the connection between the mentioned cell and the cell in the linearized table, which can be challenging to the model given the complicated table structure.

In addition, *LLMs best attend to the table items when there are clues from both text and image.* In Table 6, we observe that marking the relevant cells on the image while mentioning them through text leads to the most correctly answered examples (61 examples by GPT-4 and 38 by $Gemini_{pro}$ at the last row in Table 6). Such a dual-modality approach that combines visual cues with text references, enhances LLMs' overall reasoning ability over the tabular data.

5 Open Problems to Increase the Performance of LLMs on Tabular Data

Mathematical reasoning. We observe that *LLMs are not good at arithmetic reasoning* similar to the findings in prior works (Hendrycks et al., 2021; Imani et al., 2023). As shown in Figure 7, simple arithmetic computing like counting the total number of rows that satisfy certain conditions ('1st' in Figure 7) still poses challenges even for GPT-4. *This suggests that these previously proposed benchmarks are still valuable in evaluating LLMs*, as many of these datasets involve arithmetic reasoning

[*] Except for ToTTo, where the task is to generate the sentence based on the highlighted cells. On ToTTo, we include the highlight information just in text.

	🔓	🔒	Δ	Metric
FinQA	47.0	57.0	+10.0	Acc
LogicNLG	43.4	58.5	+15.1	
TabFact	51.8	74.7	+22.9	
WikiTQ	69.0	86.0	+17.0	
E2E	37.1	46.0	+ 8.9	ROUGE-L
ToTTo	30.1	47.7	+17.6	

Table 7: Performance scores of the best performed open-source (🔓) LLM we test, LLaMa-2-70B versus closed-source (🔒) LLM we test, GPT-4 on different datasets. The closed-source LLMs always outperform the open-source LLMs and we report the performance difference Δ between them. For consistency across different datasets, we **do not include the performances with highlighting cells**[*] in this comparison. Table 15 in Appendix B provides what combinations of table representation and prompting method yield this performance.

Date	Country	Placing	…
10/31/2008	UK	1st	…
…			

Question: What is the number of 1st place finishes across all events?

Gold: 17
GPT-4: 15 ✗

Figure 7: An example from WikiTQ where GPT-4 cannot answer it correctly with bracket table representation and chain-of-thought prompting. In addition, for most of the table representations and prompting styles, GPT-4 cannot answer this question correctly.

such as FinQA and WikiTQ, which is challenging for even the most sophisticated LLMs like GPT-4. For datasets like E2E for table-to-text generation, though relatively easy for advanced models like GPT-4, smaller models cannot describe the table fully correctly as shown in Figure 8. Therefore, these datasets are still valuable for benchmarking the progress of smaller LLMs.

Appendix D provides more examples from each dataset where LLMs can and cannot answer well.

Closing the gap between open-source and closed-source LLMs In Table 7, we report the best performance among different prompting methods for the best performed open-source LLaMa-2 models versus the best performed closed-source GPT models on FinQA, LogicNLG, TabFact, and WikiTQ. We note that on these tasks, GPT-4 always performs the best among all the closed-source LLMs we test.

We observe that *open-source LLaMa-2 models obtain decent performances across these benchmarks.* as shown in Table 15. Figure 8 shows an example from E2E dataset. The smaller LLaMa-2 models such as LLaMa-2-7B or LLaMa-2-13B make mistakes in counting rows. However, they capture almost all the information in the table including the name, eat type, and area of the restaurant. As the model scales up, the LLaMa-2 70B model can describe the table accurately.

Question: Write a sentence to describe the table.

Name	EatType	Area
Blue Spice	pub	Riverside

Model	Generated	
L-7B	The table contains three columns and three rows, with the data "Blue Spice", "pub", and "riverside" in the first column, respectively.	✗
L-13B	The table contains information about three different restaurants, including their names, eat types (pub or restaurant), and locations (riverside or not).	✗
L-70B	The table shows the name, eatType, and area of a restaurant, with one entry for Blue Spice, a pub located in the riverside area.	✓

Figure 8: Table-to-text generation from E2E. We use vanilla prompting and bracket table representation across all the models. "L-7B/13B/70B" represents LLaMa-2-7B/13B/70B, respectively. We highlight the part where the model generates incorrectly in red.

However, *significant performance gaps exist between open-source LLaMa-2 models and closed-source GPT-4 models.* In Table 7, the gap between open-source LLaMa-2 models and GPT-4 can be as large as 15% on FinQA and 22.9% on TabFact. Even on LogicNLG which has the smallest performance gap, there is an 8.4% difference between the LLaMa-2 and GPT models. As LLaMa models often serve as the foundation models for a wide range of NLP research (Roziere et al., 2023; Xu et al., 2023b), we need the effort from the open-source community to keep developing stronger LLMs to close the gap between open-source and closed-source LLMs.

6 Conclusion

We have explored various representation strategies, including both text-based and innovative image-based approaches, to understand how to use LLMs effectively in tasks involving tabular data. We demonstrate the effectiveness of image-based representations and reveal the impact of prompting strategies on the performance of LLMs. We believe our insights contribute to the understanding of LLMs and how to optimize LLMs for tabular data processing.

7 Ethical Statement

We conduct our studies on six pre-existing and publically available datasets using various existing LLMs. Prior works have pointed out the potential bias in these LLMs (Bender et al., 2021) which practitioners need to be aware of.

8 Limitations

In this study, we do not intend to exhaust every possible text representation, image representation of tables, or every possible LLM. Moreover, we do not have access to the closed-source LLMs behind their API. We hope our findings and insights in this paper can inspire future research on table-related tasks.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Mubashara Akhtar, Oana Cocarascu, and Elena Simperl. 2022. PubHealthTab: A public health table-based dataset for evidence-based fact checking. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1–16, Seattle, United States. Association for Computational Linguistics.

Syeda Nahida Akter, Zichun Yu, Aashiq Muhamed, Tianyue Ou, Alex Bäuerle, Ángel Alexander Cabrera, Krish Dholakia, Chenyan Xiong, and Graham Neubig. 2023. An in-depth look at gemini's language abilities. *arXiv preprint arXiv:2312.11444*.

Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. 2022. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023a. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–718, Nusa Dua, Bali. Association for Computational Linguistics.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023b. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Miao Chen, Xinjiang Lu, Tong Xu, Yanyan Li, Zhou Jingbo, Dejing Dou, and Hui Xiong. 2022. Towards table-to-text generation with pretrained language model: A table structure understanding and text deliberating approach. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8199–8210, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sanxing Chen, Xiaodong Liu, Jianfeng Gao, Jian Jiao, Ruofei Zhang, and Yangfeng Ji. 2021a. HittER: Hierarchical transformers for knowledge graph embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10395–10407, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, and William Yang Wang. 2020. Logical natural language generation from open-domain tables. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7929–7942, Online. Association for Computational Linguistics.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2019. Tabfact: A large-scale dataset for table-based fact verification. *arXiv preprint arXiv:1909.02164*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. 2021b. 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, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Deborah A. Dahl, Madeleine Bates, Michael Brown, William Fisher, Kate Hunicke-Smith, David Pallett, Christine Pao, Alexander Rudnicky, and Elizabeth Shriberg. 1994. Expanding the scope of the ATIS task: The ATIS-3 corpus. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Naihao Deng, Yulong Chen, and Yue Zhang. 2022. Recent advances in text-to-SQL: A survey of what we have and what we expect. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2166–2187, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. 2020. Turl: table understanding through representation learning. *Proceedings of the VLDB Endowment*, 14(3):307–319.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Wonseok Hwang, Jinyeong Yim, Seunghyun Park, and Minjoon Seo. 2019. A comprehensive exploration on wikisql with table-aware word contextualization. *arXiv preprint arXiv:1902.01069*.

Shima Imani, Liang Du, and Harsh Shrivastava. 2023. MathPrompter: Mathematical reasoning using large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 37–42, Toronto, Canada. Association for Computational Linguistics.

Zhijing Jin, Sydney Levine, Fernando Gonzalez Adauto, Ojasv Kamal, Maarten Sap, Mrinmaya Sachan, Rada Mihalcea, Josh Tenenbaum, and Bernhard Schölkopf. 2022. When to make exceptions: Exploring language models as accounts of human moral judgment. In *Advances in Neural Information Processing Systems*, volume 35, pages 28458–28473. Curran Associates, Inc.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2021. Tapex: Table pre-training via learning a neural sql executor. *arXiv preprint arXiv:2107.07653*.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206, Saarbrücken, Germany. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Ankur 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)*, pages 1173–1186, Online. 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 (Volume 1: Long Papers)*, pages 1470–1480, Beijing, China. Association for Computational Linguistics.

Shuofei Qiao, Yixin Ou, Ningyu Zhang, Xiang Chen, Yunzhi Yao, Shumin Deng, Chuanqi Tan, Fei Huang, and Huajun Chen. 2022. Reasoning with language model prompting: A survey. *arXiv preprint arXiv:2212.09597*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. 2023. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. 2023. What does clip know about a red circle? visual prompt engineering for vlms. *arXiv preprint arXiv:2304.06712*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Zhiruo Wang, Haoyu Dong, Ran Jia, Jia Li, Zhiyi Fu, Shi Han, and Dongmei Zhang. 2021. Tuta: Tree-based transformers for generally structured table pre-training. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1780–1790.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Yufan Wu, Yinghui He, Yilin Jia, Rada Mihalcea, Yulong Chen, and Naihao Deng. 2023. Hi-ToM: A benchmark for evaluating higher-order theory of mind reasoning in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10691–10706, Singapore. Association for Computational Linguistics.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I. Wang, Victor Zhong, Bailin Wang, Chengzu Li, Connor Boyle, Ansong Ni, Ziyu Yao, Dragomir Radev, Caiming Xiong, Lingpeng Kong, Rui Zhang, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. UnifiedSKG: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 602–631, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Benfeng Xu, An Yang, Junyang Lin, Quan Wang, Chang Zhou, Yongdong Zhang, and Zhendong Mao. 2023a. Expertprompting: Instructing large language models to be distinguished experts. *arXiv preprint arXiv:2305.14688*.

Yiheng Xu, Hongjin Su, Chen Xing, Boyu Mi, Qian Liu, Weijia Shi, Binyuan Hui, Fan Zhou, Yitao Liu, Tianbao Xie, et al. 2023b. Lemur: Harmonizing natural language and code for language agents. *arXiv preprint arXiv:2310.06830*.

Jingfeng Yang, Aditya Gupta, Shyam Upadhyay, Luheng He, Rahul Goel, and Shachi Paul. 2022. TableFormer: Robust transformer modeling for table-text encoding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 528–537, Dublin, Ireland. Association for Computational Linguistics.

Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. 2023. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*, 9(1):1.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8413–8426, Online. Association for Computational Linguistics.

Liangyu Zha, Junlin Zhou, Liyao Li, Rui Wang, Qingyi Huang, Saisai Yang, Jing Yuan, Changbao Su, Xiang Li, Aofeng Su, et al. 2023. Tablegpt: Towards unifying tables, nature language and commands into one gpt. *arXiv preprint arXiv:2307.08674*.

Hongzhi Zhang, Yingyao Wang, Sirui Wang, Xuezhi Cao, Fuzheng Zhang, and Zhongyuan Wang. 2020. Table fact verification with structure-aware transformer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1624–1629, Online. Association for Computational Linguistics.

	GPT 3.5	GPT 4	GMN_{pro}	LLaMa-2 7B	LLaMa-2 13B	LLaMa-2 70B
Vanilla-T						
V	52.5	60.3	37.1	28.8	35.3	42.7
E	51.0	63.8	39.5	29.0	35.1	46.7
CoT	55.2	62.6	53.5	32.1	37.6	48.3
Bracket						
V	50.9	60.1	38.4	28.4	36.6	42.2
E	47.9	62.8	39.5	28.1	34.5	45.8
CoT	51.4	61.9	57.3	34.2	39.3	50.0
Column-JSON						
V	48.3	59.5	32.6	24.9	28.8	39.2
E	48.8	62.8	34.0	26.4	28.2	42.5
CoT	51.2	59.6	53.6	28.0	34.8	42.8
Row-JSON						
V	49.7	62.3	41.2	27.9	32.6	40.9
E	53.7	63.8	39.4	26.4	31.6	45.4
CoT	53.3	62.0	52.1	31.0	35.7	48.4
Row-Identifier						
V	52.0	61.2	38.6	27.9	38.5	43.2
E	53.2	63.0	38.2	26.1	34.0	41.8
CoT	51.6	62.1	56.5	30.6	33.0	45.9

Table 8: For text-based table representations, averaged accuracy scores across FinQA, LogicNLG, TabFact, and WikiTQ for different LLMs. "GMN_{pro}" represents $Gemini_{pro}$ model, "V", "E", and "CoT" represent vanilla, expert and chain-of-thought prompting, respectively.

	GPT 3.5	GPT 4	GMN_{pro}	LLaMa-2 7B	LLaMa-2 13B	LLaMa-2 70B
Vanilla-T						
V	28.6	44.8	35.6	21.2	20.6	20.0
E	29.3	44.9	35.9	15.8	20.6	16.8
CoT	21.3	44.6	21.1	16.7	17.7	18.1
Bracket						
V	42.0	45.2	26.1	23.0	23.7	21.2
E	41.7	43.2	29.6	19.4	24.7	21.4
CoT	31.3	42.4	19.5	18.8	21.0	18.3
Column-JSON						
V	45.5	45.6	41.6	37.1	26.2	31.4
E	43.5	45.1	41.9	27.2	25.4	29.0
CoT	43.7	46.0	22.3	30.0	23.4	23.2
Row-JSON						
V	44.6	45.7	28.8	32.4	21.5	25.9
E	43.3	45.0	21.9	27.6	28.0	27.5
CoT	43.6	44.7	22.1	27.0	24.1	19.3

Table 9: For text-based table representations, ROUGE-L scores on E2E for different LLMs. "GMN_{pro}" represents $Gemini_{pro}$ model, "V", "E", and "CoT" represent vanilla, expert and chain-of-thought prompting, respectively. We do not include the Row-Identifier here as all the tables in E2E dataset only contains one row other than the header row.

A Research Questions Cont'd

A.1 RQ1 Cont'd. Can we use image-based representations of tabular data?

Figure 9 and Figure 10 show the performance comparison between feeding text representations versus image representations of the table to GPT-4 and $Gemini_{pro}$ for chain-of-thought and expert prompting, respectively. The results resemble similar trends as Figure 3.

A.2 RQ2 Cont'd. How do different text-based prompt methods affect LLMs' performance on tabular-related tasks?

Table 8 reports the averaged accuracy scores across FinQA, LogicNLG, TabFact and WikiTQ that use accuracy as the metric. Table 9 and Table 11 report the ROUGE-L scores of LLMs' generation on E2E and ToTTo dataset, respectively. Table 10 reports the scores annotated manually by the authors. As discussed in Section 3.4, the authors manually check whether the generated sentence captures all the information from the table and does not include any additional or misinformation. We assign "1" for sentences who satisfy the criteria and "0" otherwise.

A.3 RQ3 Cont'd. How do different image-based prompt methods affect LLMs' performance on tabular-related tasks?

Tables 12 and 13 report the ROUGE-L scores of GPT-4 and $Gemini_{pro}$ when we use image representations of tables on E2E and ToTTo dataset, respectively. Table 14 reports the scores annotated manually by the authors.

B Comparison of LLaMa Models and GPT-4 Models

Table 15 provides the details of what combination of table representation and prompting method yields the best performance with respect to the LLaMa-70B and GPT-4 models.

C Prompt Examples

Figure 11 gives an example of how we construct our prompt for an instance in WikiTQ.

D LLMs' Generation Examples on Each Dataset

Figure 12 gives examples for WikiTQA, TabFact, LogicNLG, and FinQA datasets we use, how many combinations of LLMs, table representations, and

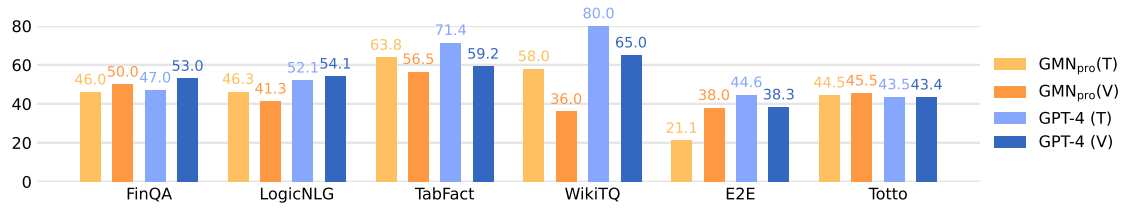

Figure 9: Performance comparison between passing the text versus image representations of tables to GPT-4 and $\text{Gemini}_{\text{Pro}}$ across FinQA, LogicNLG, TabFact, and WikiTQ by accuracy, and E2E and ToTTo by ROUGE-L scores. We use the linearized table (Vanilla-T) as the text-based representation, the original table image (Vanilla-V) as the image-based representation, and CoT prompting.

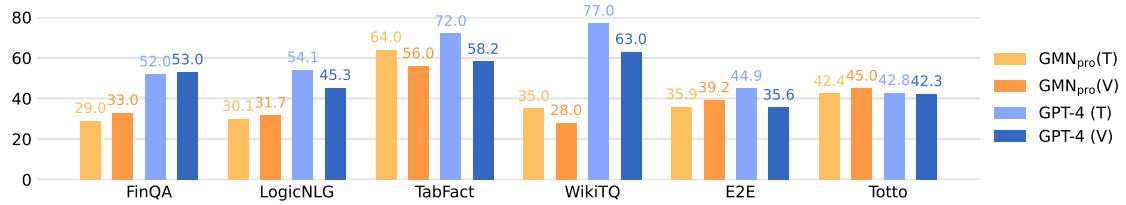

Figure 10: Performance comparison between passing the text versus image representations of tables to GPT-4 and $\text{Gemini}_{\text{Pro}}$ across FinQA, LogicNLG, TabFact, and WikiTQ by accuracy, and E2E and ToTTo by ROUGE-L scores. We use the linearized table (Vanilla-T) as the text-based representation, the original table image (Vanilla-V) as the image-based representation, and expert prompting.

prompting techniques can answer the question correctly. We notice that LLMs tend to answer well in general if the example focuses on extracting information from the table, but answer poorly if the question involves some arithmetic reasoning such as counting rows and comparing with others (examples from TabFact and LogicNLG), or complicated calculation that involves several steps (the example from FinQA). Figure 13 provides examples from E2E dataset.

Figure 14 provides examples from the ToTTo dataset, the models generally describe information better when there is less information.

	GPT 3.5	GPT 4	GMN_{pro}	LLaMa-2 7B	LLaMa-2 13B	LLaMa-2 70B
Vanilla-T						
V	28	79	60	23	15	24
E	26	81	50	8	22	12
CoT	23	86	32	17	14	19
Bracket						
V	90	94	33	28	69	32
E	94	92	39	29	78	34
CoT	74	94	36	26	62	37
Column-JSON						
V	91	82	88	63	63	77
E	91	84	85	56	76	73
CoT	90	84	60	55	67	75
Row-JSON						
V	94	93	57	54	41	48
E	94	94	33	58	72	65
CoT	95	96	62	62	49	60

Table 10: For text-based table representations, manual annotation scores (whether the generation contains all the information from the table without any additional or mis-information) on E2E for all LLMs. "GMN_{pro}" represents $\text{Gemini}_{\text{pro}}$ model, "V", "E", and "CoT" represent vanilla, expert and chain-of-thought prompting, respectively.

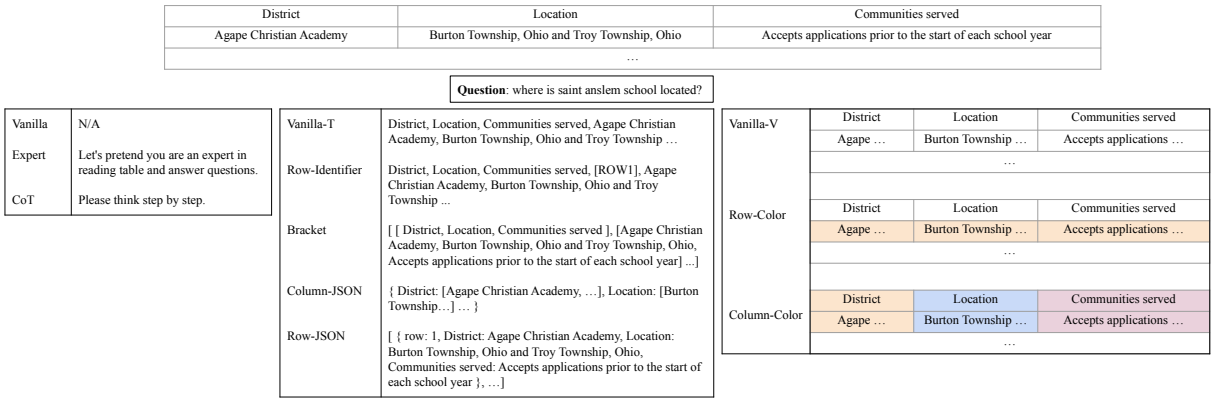

Figure 11: An example of how we construct the prompt for WikiTQ. Given the table and question, we choose from the three prompting methods, and combine with either the text-based or the image-based table representation.

	GPT 3.5	GPT 4	GMN_{pro}	LLaMa-2 7B	LLaMa-2 13B	LLaMa-2 70B
Vanilla-T						
V	43.3	43.9	42.4	21.6	22.9	27.2
E	41.4	42.8	42.4	20.7	22.2	26.3
CoT	42.7	43.5	44.5	19.8	22.8	27.4
Bracket						
V	44.2	44.9	44.8	23.6	24.8	28.9
E	41.7	43.0	44.1	22.3	23.1	29.1
CoT	43.9	45.5	43.9	23.3	24.0	29.6
Column-JSON						
V	45.5	45.5	44.5	22.1	22.7	30.1
E	41.1	43.1	44.8	19.9	20.8	10.3
CoT	43.9	44.2	45.1	22.1	21.5	28.8
Row-JSON						
V	43.1	45.1	43.7	22.3	21.9	29.4
E	40.8	43.1	43.4	21.5	21.4	27.0
CoT	42.3	45.2	45.1	22.9	22.5	26.9
Row-Identifier						
V	42.2	44.6	43.2	19.1	22.2	28.3
E	40.1	42.7	42.5	21.3	21.3	26.5
CoT	42.0	43.7	44.2	19.1	22.3	28.0

Table 11: For text-based table representations, ROUGE-L scores on ToTTo for all LLMs. "GMN_{pro}" represents $Gemini_{pro}$ model, "V", "E", and "CoT" represent vanilla, expert and chain-of-thought prompting, respectively.

	GPT-4 V	GPT-4 E	GPT-4 CoT	$Gemini_{pro}$ V	$Gemini_{pro}$ E	$Gemini_{pro}$ CoT
VV	44.0	42.3	43.4	46.0	45.0	45.5
CC	44.8	41.7	44.1	47.7	44.8	45.1
RC	44.5	42.8	43.7	46.3	44.6	45.0

Table 12: For image-based table representations, ROUGE-L scores on E2E for GPT-4 and $Gemini_{pro}$. For the headers, "V", "E", and "CoT" represent vanilla, expert, and chain-of-thought prompting, respectively. For the row names, "VV", "CC', and "RC" represent Vanilla-V, Column-Color, and Row-Color, respectively.

	GPT-4 V	GPT-4 E	GPT-4 CoT	$Gemini_{pro}$ V	$Gemini_{pro}$ E	$Gemini_{pro}$ CoT
VV	44.0	42.3	43.4	46.0	45.0	45.5
CC	44.8	41.7	44.1	47.7	44.8	45.1
RC	44.5	42.8	43.7	46.3	44.6	45.0

Table 13: For image-based table representations, ROUGE-L scores on ToTTo for GPT-4 and $Gemini_{pro}$. For the headers, "V", "E", and "CoT" represent vanilla, expert, and chain-of-thought prompting, respectively. For the row names, "VV", "CC', and "RC" represent Vanilla-V, Column-Color, and Row-Color, respectively.

	GPT-4 V	GPT-4 E	GPT-4 CoT	$Gemini_{pro}$ V	$Gemini_{pro}$ E	$Gemini_{pro}$ CoT
VV	86	83	90	77	74	78
CC	86	69	93	70	61	72
RC	85	70	89	61	57	60

Table 14: For image-based table representations, manual annotation scores (whether the generation contains all the information from the table without any additional or mis-information) on E2E for GPT-4 and $Gemini_{pro}$. For the headers, "V", "E", and "CoT" represent vanilla, expert, and chain-of-thought prompting, respectively. For the row names, "VV", "CC', and "RC" represent Vanilla-V, Column-Color, and Row-Color, respectively.

	Table Repr	Prompting	🔓	🔒	Δ	Table Repr	Prompting	Metric
FinQA	Vanilla-T	CoT	47.0	57.0	+10.0	Vanilla-V	Vanilla	Acc
LogicNLG	Vanilla-T	CoT	43.4	58.5	+15.1	Row-Color	CoT	
TabFact	Column-JSON	Expert	51.8	74.7	+22.9	Row-JSON	Expert	
WikiTQ	Row-JSON	CoT	69.0	86.0	+17.0	Row-Identifier	CoT	
E2E	Column-JSON	Vanilla	37.1	46.0	+8.9	Column-JSON	CoT	ROUGE-L
ToTTo	Column-JSON	Vanilla	30.1	47.7	+17.6	Column-Color	Vanilla	

Table 15: Performance scores of the best performed open-source (🔓) LLM we test, LLaMa-2-70B versus closed-source (🔒) LLM we test, GPT-4 on different datasets. Four datasets uses accuracy as metrics and two datasets (E2E and ToTTo) uses ROUGE-L as metrics. The closed-source LLMs always outperform the open-source LLMs and we report the performance difference Δ between them. We include the table representation ("Table Repr") and prompting methods that yield the best performance next to the columns that report open-source and closed-source LLM scores, respectively. For consistency across different datasets, we **do not include the performances with highlighting cells** in this comparison.

Dataset	Question / Statement	Table	Gold	Correct (out of 108)
WikiTQ	Who won the most gold medals?	Rank | Nation | Gold | … 1 | Brazil | 7 | …	Brazil	107
	What is the number of 1st place finishes across all events?	Date | Plaing | Event | … … | 1 | Sprint | …	17	8
TabFact	Ben Curtis , J B Holmes , Steve Flesch, and David Tom be from the united state.	Place | Player | Country | … 1 | Ben Curties | 7 | …	True	76
	March be feature more often as a month in the date than any other month.	Date | Result | … 3 March 2009 | 7 | …	True	13
LogicNLG	In each of the event there were 4 [ENT] on a team.	Rank | Rowers | … 4 | vitasek , dolecek , hanak , irka | …	rowers	102
	[ENT] won more medal than anyone else.	Rank | Nation | Total | … 1 | Marit BjØrgen (Nor) | 5 | …	Marit BjØrgen (Nor)	29
FinQA	What was the average net revenue between 2016 and 2017 in millions?	Amount (in millions) | … 2016 | $ 705.4 | …	705.25	102
	What are the nuclear realized price changes as a percentage of the decrease in net revenue from 2011 to 2012	Amount (in millions) 2011 Net Revenue | $ 2045 Nuclear Realized Price Change | -194 (194) 2012 Net Revenue | 1854	101.571%	21

Figure 12: Examples from WikiTQ, TabFact, LogicNLG, FinQA with the number of correctly answered cases. For each example, we have 108 cases corresponding to the three prompting methods, five text-based table representations, and six LLMs, together with three prompting methods, three image-based table representations, and two LLMs. We omit some table content to assist readers.

Dataset	Question / Statement	Table				Gold	Correct
E2E	Write a sentence to describe the table.	Name	EatType	Customer Rating	Near	Near Burger King is the Blue Spice coffee shop. It has average customer ratings.	78
		Blue Spice	Coffee Shop	Average	Burger King		
	Write a sentence to describe the table.	Name	EatType	Customer Rating	Near	The pub Blue Spice is based near Crowne Plaza Hotel and has a high customer rating of 5 out of 5.	60
		Blue Spice	Pub	5 out of 5	Crowne Plaza Hotel		

Figure 13: Examples from the E2E dataset with the number of generations that capture all the table information without any false information (manually annotated by the authors). For each example, we have 102 cases as we exclude the Row-Identifier because there is one row for each table.

Dataset	Question / Statement	Table				Gold	Avg ROUGE-L
ToTTo	Write a sentence with respect to the corresponding cells in the table. Title: Baudette Air Force Station; Awards	Award Steamer	Award	Dates	Notes	Baudette Air Force Station was awarded the Air Force Outstanding Unit Award for the period, 1 June 1971 through 31 May 1973.	58.4
		-	Air Force Outstanding Unit Award	...			
	Write a sentence with respect to the corresponding cells in the table. Title: Sora Amamiya	Year	Title	Role		In 2015, Sora Amamiya was cast as Isla in Plastic Memories and as Miia in Monster Musume.	40.9
		2015	Plastic Memories	Isla			
		2015	Monster Musume	Miia			
		...					

Figure 14: Examples from the ToTTo dataset with the average ROUGE L scores for the generation. For each example, we have 108 cases similar to Figure 12. Since ToTTo requires to generate information about relevant cells in the table, we provide the relevant cells' information through text across all the experiments on ToTTo.