TABVERSE: Benchmarking Cross-Format Table Understanding in LLMs and VLMs

Anonymous ACL submission

Abstract

Large Language Models (LLMs) and Vision-Language Models (VLMs) are increasingly evaluated on table reasoning tasks, but the role of table representation remains under-explored. In practice, the same table content may appear in different structural formats, such as HTML, Markdown, and LaTeX, or as rendered images. However, existing evaluations often let content, format, layout, and modality vary together, making it difficult to isolate representation effects. We introduce TABVERSE, a controlled multimodal table benchmark that aligns the same table content across multiple structural formats and rendered images, with question category and difficulty tags. This design enables systematic evaluation of representation effects while holding table content fixed. We evaluate LLMs and VLMs across three tasks: Question Answering (QA), Structural Understanding Capability (SUC), and Structure Reconstruction (SR). Our results show that representation choice substantially affects table understanding. Models generally perform better with structured text than with rendered images, but the size of this gap depends on the task, model, and format. HTML is often the most robust text format, while row-sensitive structural tasks and syntactically usable LaTeX reconstruction remain challenging. These findings show that table representation is a key factor in reliable table evaluation. Data and code are available at `http://anonymous.for.review`.

1 Introduction

Tables are widely used to present structured information in scientific documents, reports, and web content. This makes table understanding critical for AI systems that interpret and verify real-world data (Smock et al., 2022). Despite strong progress in Large Language Models (LLMs) and Vision-Language Models (VLMs), table comprehension remains challenging (Brown et al., 2020; Touvron et al., 2023; Bubeck et al., 2023).

Unlike plain text, tables require models to interpret both content and structure, including headers, merged cells, row and column boundaries, and relevant cells (Deng et al., 2024; Sui et al., 2024a; Kim et al., 2024). The same table content can also be presented in different ways i.e., as HTML, LaTeX, or Markdown, or as a rendered image in a PDF or screenshot. These representations expose different cues. Structured text provides markup and delimiters, while images provide visual layout, so model performance can change even when the underlying table content is identical.

Recent work has introduced table-specialized models (Zhang et al., 2024; Deng and Mihalcea, 2025) and shown that table reasoning is sensitive to serialization, prompting, and modality choices (Deng et al., 2024; Sui et al., 2024a,b; Singha et al., 2023). However, many benchmarks and evaluation pipelines let table content, format, layout, and modality vary together, making it difficult to isolate the effect of representation itself.

We introduce TABVERSE, a benchmark for controlled cross-format and cross-modality table evaluation. TABVERSE aligns identical tables across three structural formats (HTML, LaTeX, Markdown) and their rendered images, enabling comparison while holding table content fixed. Built from held-out evaluation splits of FEVEROUS, HYBRIDQA, TABFACT, SQA, and WIKITABLEQUESTIONS, it includes a full tagged pool and a 700-sample balanced evaluation set balanced by question category and difficulty.

We evaluate LLMs and VLMs on three complementary tasks: Question Answering (QA), Structural Understanding Capability (SUC), and Structure Reconstruction (SR). QA measures answer prediction under different table representations; SUC probes structure understanding through boundary detection, size estimation, and index-based retrieval; and SR measures whether VLMs can reconstruct tables from rendered images.

Our contributions are as follows:

- We formulate cross-format and cross-modality table understanding as a controlled evaluation problem, where table content is fixed while structural format and input modality vary under matched pipelines.
- We introduce TABVERSE, an aligned multimodal table benchmark with HTML, LaTeX, and Markdown representations, corresponding rendered images, category and difficulty tags, and a 700-sample balanced evaluation set from five TableQA sources.
- We benchmark LLMs and VLMs across matched text-only and image-based table inputs on QA, SUC, and SR, revealing how format and modality choices change model behavior across tasks and question groups, and how SR errors separate into table reconstruction quality and output usability.

Our experiments show that representation matters. Structured text often outperforms rendered images, HTML is often the most robust text format, and usable LaTeX reconstruction remains challenging.

2 Related Work

Table 1 summarizes related resources, which cover many table reasoning tasks but rarely isolate representation effects because content, format, layout, and modality often vary together.

Table reasoning benchmarks: Early benchmarks established table QA over semi-structured tables, covering lookup, filtering, aggregation, and simple symbolic operations (Pasupat and Liang, 2015; Zhong et al., 2017). Later datasets expanded to sequential QA, fact verification, multi-hop reasoning over tables and text, open-domain and multi-table QA, and table-grounded generation (Iyyer et al., 2017; Chen et al., 2020a,b; Aly et al., 2021; Chen et al., 2021a; Wu et al., 2025a; Parikh et al., 2020; Nan et al., 2022). Other resources target numerical reasoning (Chen et al., 2021b; Zhu et al., 2021), hierarchical tables (Cheng et al., 2022), long-context cell retrieval (Wang et al., 2026), and complex or multilingual table understanding (Zhu et al., 2025). These benchmarks provide important testbeds, but most evaluate a fixed representation or task setting; TABVERSE instead tests the same table-question pairs across aligned textual formats and rendered images.

Representation and multimodal table evaluation: Prior work shows that table reasoning depends strongly on serialization, prompting, segmentation, and modality. Table Meets LLM and TAP4LLM study prompting, sampling, augmentation, and structural decomposition (Sui et al., 2024a,b), while tables-as-text-versus-image comparisons show that representation choice can substantially change performance (Deng et al., 2024). LongTableBench and RealHiTBench evaluate long or hierarchical tables under multiple input formats (Li et al., 2025; Wu et al., 2025b); related studies examine source-sensitive table understanding, table-image modeling, and cross-domain evaluation behavior (Yang et al., 2025; Chen et al., 2023; Borisova et al., 2025); and multimodal benchmarks cover visual QA, semi-structured tables, rendered table images, and table-image retrieval (Kim et al., 2024; Singh et al., 2025; Mathur et al., 2024; Zheng et al., 2024; Titiya et al., 2025; Talmor et al., 2021; Lompo and Haraoui, 2025; Li et al., 2026; Xu et al., 2026). These works motivate format-aware and image-based table evaluation, but often vary table source, layout, visual complexity, and representation together across different experimental settings; TABVERSE holds table content fixed while systematically varying structural format and input modality.

Table reconstruction and table-focused modeling: Beyond QA, table reconstruction and table-structure recognition studies extract structured representations from rendered tables or document images (Roberts et al., 2025; Li et al., 2020). This is related to our SR task, but prior work usually focuses on recognition accuracy or one target representation rather than reconstruction across aligned input and output formats. Table-specialized pretraining and instruction tuning have been proposed for table manipulation, reasoning, and generation (Herzig et al., 2020; Gong et al., 2020; Zhang et al., 2024; Zha et al., 2023; Li et al., 2024; Su et al., 2024; Zhang et al., 2025; Deng and Mihalcea, 2025); recent systems also combine OCR-style transcription with LLM reasoning for table VQA (Guo et al., 2025) or add supervision through code-driven reasoning traces and structure-aware guidance (Nguyen and Okatani, 2026; Zhu et al., 2026). These efforts are complementary to TABVERSE, which evaluates LLMs and VLMs under controlled QA, SUC, and SR settings across aligned textual and visual table representations.

	Representations		Visual Renders						
Literature	**Text**	**Images**	**HTML**	**LaTeX**	**Markdown**	**Others**	**Q-diff**	**Q-cat**	**Aligned**
Table Reasoning and Multimodal Benchmarks									
TableVQA-Bench (Kim et al., 2024)	✓	✓	✓	✗	✗	✓	✗	✗	✓
MTabVQA (Singh et al., 2025)	✗	✓	✗	✗	✗	✓	✗	✓	✗
MMTabQA (Mathur et al., 2024)	✓	✓	✗	✗	✗	✓	✗	✗	✗
MMTabQA (Mathur et al., 2024)	✗	✓	✗	✗	✗	✓	✗	✓	✗
NeedleInATable (Wang et al., 2026)	✓	✓	✗	✗	✗	✓	✗	✗	✓
TableVLM (Chen et al., 2023)	✓	✓	✓	✗	✗	✗	✗	✗	✓
Evaluation Frameworks / Controlled Studies									
Tables as Texts or Images (Deng et al., 2024)	✓	✓	✗	✗	✗	✓	✗	✗	✓
RealHiTBench (Wu et al., 2025b)	✓	✓	✗	✗	✗	✓	✗	✓	✓
LongTableBench (Li et al., 2025)	✓	✗	✗	✗	✗	✗	✗	✗	✗
Image2Struct (Roberts et al., 2025)	✗	✓	✓	✓	✗	✗	✗	✗	✗
TabVerse (Ours)	✓	✓	✓	✓	✓	-	✓	✓	✓

Table 1: Table understanding resources related to TabVerse. Under *Visual Renders*, HTML, LaTeX, and Markdown indicate whether the work provides or evaluates tables rendered from those source formats, while *Others* covers other visual styles or image sources. *Q-diff* and *Q-cat* denote question difficulty and category annotations. *Aligned* denotes paired textual tables and images with the same table content.

3 Dataset

3.1 Dataset Construction

TabVerse is constructed from filtered held-out splits of five TableQA datasets: FEVEROUS (Aly et al., 2021), TabFact (Chen et al., 2020a), SQA (Iyyer et al., 2017), HybridQA (Chen et al., 2020b), and WikiTableQuestions (WikiTQ) (Pasupat and Liang, 2015). We keep only single-table questions answerable from the table alone and check for overlap with the corresponding training splits where identifiers are available (Appendix B). This yields a full pool of 6,097 question–table pairs from 4,434 unique tables. After category and difficulty tagging, we select a 700-sample balanced evaluation set covering 629 unique tables (Table 2).

3.2 Aligned Formats and Rendered Images

For each table, we create HTML, Markdown, and LaTeX representations and render one image from each. We adapt conversion utilities from Sui et al. (2024a) and extend them for dataset artifacts such as missing values and special characters. Standardized font size, padding, and width keep textual and visual versions aligned by construction. Implementation details are in Appendix B.

3.3 Question Category and Difficulty

Each question is tagged with a *question category* and a binary *difficulty label*. The seven categories are Simple Lookup, Conditional Lookup, Multi-item Lookup, Aggregation/Arithmetic, Comparison/Extremum, Single-step Binary Verification, and Multi-hop Binary Verification. Gemini-3-Flash-Preview assigns the initial category tags, which are manually reviewed and corrected where needed.

Difficulty is estimated from zero-shot QA on rendered table images. GPT-5.2 and Gemini-3-Flash-Preview answer each question using the three aligned image renders produced from HTML, Markdown, and LaTeX, giving six correctness indicators per question. Questions scoring 0–3 are labeled *Hard*; those scoring 4–6 are labeled *Easy*.

3.4 Balanced Evaluation Set

We balance the final evaluation set by difficulty and question category. It contains 700 question–table pairs: 350 Easy and 350 Hard questions, with 50 examples per category within each difficulty level. These questions reference 629 unique tables.

3.5 Supported Tasks

TabVerse supports three tasks: QA, Structural Understanding Capability (SUC), and Structure Reconstruction (SR).

QA: Given a question and a table as structured text or a rendered image, the model predicts an answer following the source dataset conventions.

SUC: SUC probes table structure through boundary detection, size estimation, and index-based retrieval. We adapt the templates from Sui et al. (2024a) and extend them with probes for table grounding and document-style tables. Prompt templates appear in Appendix C.

SR: Given a rendered table image, the model reconstructs the table in HTML, Markdown, or LaTeX.

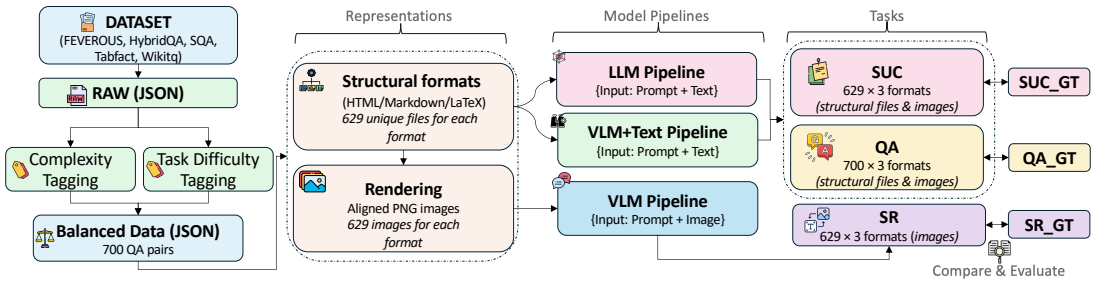

Figure 1: Overview of TABVERSE. From the balanced evaluation set, each table is represented in three structural formats (HTML, Markdown, LaTeX) with corresponding rendered images. These aligned multimodal pairs enable evaluation on QA, SUC, and SR tasks across VLMs and LLMs for cross-format and cross-modality analysis.

Split	Question–table pairs	Unique tables
Full tagged pool	6,097	4,434
Balanced set	700	629

Table 2: Dataset statistics for TABVERSE. The full tagged pool contains all filtered question-table pairs after category and difficulty tagging. The balanced evaluation set is used for all experiments, with equal coverage across difficulty and question category.

4 Experimental Settings

We evaluate how structural format and input modality affect table understanding while keeping table content fixed. Using TABVERSE, we vary the table format (HTML, Markdown, LaTeX) and input modality (structured text vs. rendered image), allowing performance differences to be attributed to representation rather than content variation.

4.1 Evaluation Pipelines

We evaluate LLMs and VLMs on TABVERSE using three different pipelines:

VLM-Image. The VLM receives each question with a *rendered table image*. To measure visual format effects, we use three aligned image renderings per instance, rendered from HTML, Markdown, and LaTeX sources, while keeping the question and table content unchanged.

VLM-Text. The VLM receives the question prompt with a *structured text-based table* in one of the three formats, without an image. Comparing VLM-Text with VLM-Image isolates the impact of visual input within the same model.

LLM-Text. The LLM receives the question prompt with a *structured text-based table* in one of the three formats. Comparing LLM-Text with VLM-Text highlights differences between language-only and multimodal models on identical text inputs.

4.2 Models

We evaluate several LLMs and VLMs, including both general-purpose and table-specialized models. For LLMs, we use Qwen2.5-7B-Instruct (Yang et al., 2024), Qwen3-30B-A3B-Instruct (Team, 2025b), TableGPT2-7B[1] (Su et al., 2024), and TAMA-QWen3[1] (Xing et al., 2025). For VLMs, we use SmolVLM2-2.2B-Instruct (Marafioti et al., 2025), Gemma-3-12B-IT (Team, 2025a), Gemma-3-27B-IT (Team, 2025a), InternVL3.5-14B (Wang et al., 2025), InternVL3.5-30B-A3B (Wang et al., 2025), Qwen3-VL-8B-Instruct (Team, 2025b), Qwen3-VL-30B-A3B-Instruct (Team, 2025b), Ministral-3-14B-Instruct (Liu et al., 2026), LLaVA-1.6-7B and LLaVA-1.6-13B (Liu et al., 2024), TableLLaVA-v1.5-7B[1] (Zheng et al., 2024), GPT-5.2 (OpenAI, 2025), and Gemini-3-Flash-Preview (Google, 2024).

4.3 Evaluation Protocol

All experiments follow a uniform zero-shot setup across the three tasks[2]. Models generate outputs using greedy decoding (`temperature=0`, `top_p=1`) with task-specific output limits: short for SUC, medium for QA, and long for SR. We apply minimal output normalization, including removing common answer prefixes such as *the answer is* and normalizing whitespace and casing where appropriate.

For QA and SUC, we report exact-match accuracy following the dataset conventions after light normalization. For SUC, we additionally report Field Accuracy and Relaxed Accuracy as diagnostics for structured answers with pipe-separated fields. Field Accuracy compares each gold field with the prediction at the same position, while Re-

[1]Table-specialized model designed for table understanding tasks.

[2]The prompt templates are provided in Appendix C.

laxed Accuracy checks whether each gold field appears anywhere in the prediction. These diagnostics are used for additional analysis, while exact match remains the main SUC metric.

For SR, we report GRITS (Smock et al., 2023) using both GriTS-Topology and GriTS-Content. GriTS-Topology measures structural similarity between the reconstructed and reference tables, while GriTS-Content measures cell-text fidelity. We also measure syntactic usability for each requested target format using HTML parse success, Markdown render/parse success, and LaTeX compilation success.

To connect usability with reconstruction quality, Appendix D.3 reports two usability-aware variants of GRITS. Valid-only GRITS averages scores only over syntactically usable outputs when at least one output is usable:

$$\mathrm{GriTS}_{\mathrm{valid}} = \frac{\sum_{i=1}^{N} u_i \cdot \mathrm{GriTS}_i}{\sum_{i=1}^{N} u_i},$$

where $u_i = 1$ if the generated output is syntactically usable and $u_i = 0$ otherwise. Zero-penalized GRITS instead assigns unusable outputs a score of zero before averaging:

$$\mathrm{GriTS}_0 = \frac{1}{N} \sum_{i=1}^{N} u_i \cdot \mathrm{GriTS}_i.$$

5 Results

This section evaluates model performance across different table formats, input modalities, and table comprehension tasks.

5.1 Question Answering

Table 3 reports strict exact-match (EM) accuracy on 700 QA questions across three aligned table representations.

Overall performance. Gemini-3-Flash-Preview obtains the highest scores across all formats and modalities, followed by GPT-5.2. Among open-weight VLMs, Qwen3-VL-8B-IT is strongest under strict EM, while Qwen3-30B-A3B-IT is the strongest text-only LLM. Larger models are not always better under strict EM: the 8B Qwen3-VL variant outperforms the 30B-A3B variant across modalities and formats in Table 3. However, Appendix 14 shows that the 30B-A3B variant recovers substantially under relaxed matching, especially in VLM-Text. This suggests that part of the strict EM gap comes from answer-formatting behavior rather than answer retrieval alone. InternVL3.5 shows a milder version of the same pattern.

Modality and format effects. Structured table text does not consistently outperform rendered images. Gemma-3 benefits from text inputs, SmolVLM2 and InternVL3.5 perform similarly or better on images, and Gemini-3-Flash-Preview remains nearly unchanged across modalities. At the format level, rendered-image scores are usually close across HTML, LaTeX, and Markdown sources, while text pipelines show larger format gaps. This suggests that symbolic table format affects text-based reasoning more strongly than rendered-image reasoning.

Task factors. Beyond modality and format, QA performance varies by question category and difficulty. Verification questions tend to be easier, while multi-item lookup and aggregation/counting questions remain challenging, as illustrated by the category-wise profiles in Figure 9. Accuracy also drops substantially from Easy to Hard questions. We provide the full category and difficulty breakdowns in Appendix D.1.

5.2 Structural Understanding Capability

Tables 4–6 evaluate Structural Understanding Capability (SUC) across ten structure-oriented subtasks defined in Section 3.5. Tables 4 and 15 report VLM-Image EM results, while Tables 17 and 12 report VLM-Text and LLM-Text results.

Overall performance. GPT-5.2 obtains the highest overall VLM-Image scores, while Qwen3-VL-30B and InternVL3.5-14B are the strongest open-weight VLMs in Table 4. Gemini-3-Flash performs well on boundary and column-oriented subtasks, but weak size detection, cell lookup, and row retrieval reduce its overall score. Overall, SUC is highly task-dependent. Column counting and column retrieval are among the easiest subtasks, whereas row retrieval, cell lookup, table partitioning, and size detection remain difficult, as shown in Figure 10.

Field Accuracy reveals partial structural recovery. Field Accuracy shows that low EM often reflects incomplete or shifted localization rather than entirely wrong table understanding. In Table 5, the largest gains occur on row/column-sensitive subtasks: InternVL3.5-14B gains +30.8 on table par-

Model	Params.	Table Image Render			Table Text Format		
		HTML	LaTeX	Markdown	HTML	LaTeX	Markdown
Language Models (text-only)							
Qwen2.5-IT (Yang et al., 2024)	7B	–	–	–	44.57	42.71	45.43
Qwen3-IT (Team, 2025b)	30B (A3B)	–	–	–	**51.14**	**48.43**	**46.57**
TableGPT2 (Su et al., 2024)	7B	–	–	–	44.43	41.57	42.14
TAMA-QWen3 (Xing et al., 2025)	–	–	–	–	18.29	19.14	20.71
Vision-Language Models							
SmolVLM2-IT (Marafioti et al., 2025)	2.2B	29.71	28.71	25.86	21.57	17.63	15.75
Gemma-3-IT (Team, 2025a)	12B	38.86	39.57	38.57	50.29	49.00	48.57
Gemma-3-IT (Team, 2025a)	27B	46.14	45.29	45.43	**53.43**	51.29	53.14
InternVL3.5 (Wang et al., 2025)	14B	48.57	48.14	48.00	47.14	47.29	44.86
InternVL3.5 (Wang et al., 2025)	30B (A3B)	47.86	**50.00**	48.29	45.86	45.71	47.00
Qwen3-VL-IT (Bai et al., 2023)	8B	**50.29**	49.29	**49.71**	**53.43**	**52.14**	**53.29**
Qwen3-VL-IT (Bai et al., 2023)	30B (A3B)	41.14	42.14	41.43	45.29	43.57	39.71
Ministral-3-IT (Liu et al., 2026)	14B	44.43	39.14	42.71	40.00	35.43	36.57
LLaVA-1.6* (Liu et al., 2023)	7B	31.86	31.43	32.00	27.37	29.55	26.50
LLaVA-1.6* (Liu et al., 2023)	13B	25.14	23.71	25.00	23.91	22.13	24.89
TableLLaVA-v1.5* (Zheng et al., 2024)	7B	1.29	1.00	4.00	23.61	27.37	28.40
Proprietary Models							
GPT-5.2 (OpenAI, 2025)	–	54.57	54.52	56.14	57.43	57.29	58.00
Gemini-3-Flash-Preview (Google, 2024)	–	**65.43**	**65.14**	**65.43**	**65.71**	**65.00**	**65.43**

Table 3: **TaskQA results (700 questions).** Exact-match accuracy (%) across three aligned table representations (HTML/LaTeX/Markdown). Text-only models (LLM-Text) operate on structured table text (right block). Vision-language models are evaluated both on rendered table images (left block) and on structured table text (right block), which isolates modality effects while keeping the underlying table content identical.

titioning and +30.2 on cell lookup, while Qwen3-VL-30B gains +30.1 on size detection. These gaps are especially informative for row-sensitive outputs, where models often recover part of the structure but miss exact localization.

Header and indexing cues affect SUC scores. Table 6 compares the explicit SUC prompt, which states header exclusion and 0-indexed row/column coordinates, with an implicit prompt that removes these details. The effect is strongest on index-dependent subtasks. For first-cell detection, the explicit prompt gives much higher EM for GPT-5.2 (+83.8) and Gemini-3-Flash (+94.8), with similar gains for the strongest open models; table partitioning shows the same trend for GPT-5.2 (+80.5) and Gemini-3-Flash (+88.3). The effect is much weaker for last-cell detection, suggesting that many first-cell errors come from treating the header as the first row. In contrast, reverse lookup improves under the implicit prompt for GPT-5.2 (-52.7) and Gemini-3-Flash (-15.4). Thus, SUC also tests whether models follow the intended row-indexing and header-inclusion convention.

Pipeline and format effects. Tables 4, 15, 17, and 12 show that SUC is generally stronger with structured table text than with rendered images. The gains are clearest for GPT-5.2 and Gemini-3-Flash, while open-weight models improve less uniformly. The largest image-to-text improvements occur on row-boundary and header-sensitive tasks. Row retrieval and cell lookup remain very low for most VLM-Image models despite strong column counting and column retrieval, but improve for several models in VLM-Text; Figure 11 shows that they remain the main bottlenecks across pipelines.

Table 12 shows that text alone does not solve SUC for LLMs. Text-only Qwen3-30B is strong on HTML text but drops on LaTeX and Markdown, while smaller and table-specialized LLMs remain weak on coordinate lookup and row retrieval. Format effects are present but not universal. HTML is often the safest text format for text-input pipelines, while rendered-image results show smaller and less consistent format differences. Overall, SUC depends on input modality, table format, and row/column indexing behavior.

5.3 Structure Reconstruction

Table 7 evaluates Structure Reconstruction (SR), where models receive rendered table images and reconstruct the table into HTML, Markdown, or LaTeX. We report GRITS-Topology for structural fidelity, GRITS-Content for cell-text fidelity, and output usability in Tables 8 and 13. Appendix D.3 further reports valid-only and zero-penalized GRITS, separating syntactic usability failures from reconstruction errors among usable outputs.

Models	Formats	T.P.	F.C.	L.C.	S.D.	# Rows	# Cols	C.Lu.	R.Lu.	Co.Rt.	Ro.Rt.	Overall
Open Models												
Gemma-3-27B-IT	HTML	29.6	63.9	52.3	21.0	39.9	50.1	22.6	32.6	70.9	10.5	39.3
	LaTeX	28.3	59.6	52.0	23.5	39.7	50.2	22.7	32.6	70.3	9.1	38.8
	Markdown	30.2	59.0	51.5	27.3	44.4	55.8	24.2	33.7	74.2	**11.3**	41.2
InternVL3.5-14B	HTML	35.3	86.8	68.4	22.3	55.3	89.8	15.6	15.9	84.6	4.1	47.8
	LaTeX	38.0	88.6	71.9	16.4	28.6	83.1	18.9	22.7	90.1	6.4	46.5
	Markdown	31.6	83.0	65.3	32.3	51.0	86.6	20.3	23.5	88.9	5.4	48.8
Qwen3-VL-30B-A3B-IT	HTML	50.2	92.5	85.2	32.9	44.2	91.9	21.1	24.3	71.1	0.2	51.4
	LaTeX	45.6	92.7	82.2	34.2	40.2	87.4	22.3	27.3	83.8	4.0	52.0
	Markdown	40.1	87.9	78.7	32.0	41.7	87.4	21.6	25.1	88.9	1.1	50.4
Ministral-3-14B-IT	HTML	36.1	40.5	61.5	22.7	35.1	46.4	26.6	**39.6**	79.7	10.7	39.9
	LaTeX	36.1	41.8	53.3	38.2	35.6	36.1	**26.9**	39.4	72.7	11.1	39.1
	Markdown	30.7	34.3	49.4	38.5	35.6	52.5	26.6	38.8	74.2	10.7	39.1
LLaVA-1.6-13B	HTML	0.2	31.5	27.5	4.5	19.4	28.0	1.4	6.4	37.5	0.0	15.6
	LaTeX	0.2	32.9	29.6	3.8	20.0	20.5	2.2	5.6	36.7	0.2	15.2
	Markdown	0.5	35.6	31.6	4.3	19.1	22.6	2.1	6.2	36.6	0.0	15.9
Proprietary Models												
GPT-5.2	HTML	**93.0**	**97.1**	**94.9**	32.9	78.4	98.1	2.7	15.7	95.2	1.7	61.0
	LaTeX	85.4	93.3	88.1	57.7	85.7	96.5	3.5	19.2	93.5	2.5	62.5
	Markdown	87.4	93.6	91.6	**80.3**	**91.3**	98.9	3.0	20.8	96.0	5.1	**66.8**
Gemini-3-Flash-Preview	HTML	91.7	97.0	88.7	0.2	0.5	94.9	0.6	14.8	94.4	0.3	48.3
	LaTeX	86.1	94.8	87.4	0.0	6.5	99.2	1.6	15.1	95.7	0.8	48.7
	Markdown	87.9	94.8	87.7	0.2	1.9	**99.7**	1.3	14.9	**97.3**	0.3	48.6

Table 4: **Structural Understanding Capability (SUC) results for the VLM-Image pipeline.** Exact-match accuracy (%) across ten structure-oriented subtasks, including table partitioning (T.P.), boundary detection (F.C., L.C.), size estimation (S.D., #Rows, #Cols), coordinate lookup (C.Lu., R.Lu.), and index-based retrieval (Co.Rt., Ro.Rt.). Models receive rendered table images derived from HTML, LaTeX, and Markdown sources.

	Δ (Field Acc. - EM)			
Model	**T.P.**	**S.D.**	**C.Lu.**	**Ro.Rt.**
Gemma-3-27B-IT	+25.7	+21.3	+20.9	+13.7
InternVL3.5-14B	+30.8	+29.5	+30.2	+11.8
Qwen3-VL-30B-A3B-IT	+28.3	+30.1	+27.0	+20.6
Ministral-3-14B-IT	+18.2	+25.5	+20.2	+11.7
LLaVA-1.6-13B	+6.2	+17.2	+7.8	+3.6
GPT-5.2	+4.9	+21.0	+42.1	+13.1
Gemini-3-Flash-Preview	+7.3	+48.9	+35.2	+13.4

Table 5: **Field-level gaps on selected SUC subtasks.** Values report $\Delta = \text{Field Accuracy} - \text{EM}$, averaged over HTML, LaTeX, and Markdown image renders. Larger gaps indicate subtasks where models often recover part of the structured answer but fail exact match.

	Δ_{EM} (Explicit – Implicit)					
Model	**T.P.**	**F.C.**	**C.Lu.**	**R.Lu.**	**Co.Rt.**	**Ro.Rt.**
Gemma-3-27B-IT	+13.2	+28.4	+22.0	+7.0	+9.7	+4.1
InternVL3.5-14B	+17.0	+65.8	+18.0	+15.9	+26.7	+1.7
Qwen3-VL-30B*	+33.1	+79.6	+21.7	+9.8	+5.1	+0.0
Ministral-3-14B-IT	-8.2	+6.6	+24.4	+23.6	+0.3	+3.6
LLaVA-1.6-13B	+0.2	-21.1	+1.6	-0.7	+3.2	+0.1
GPT-5.2	+80.5	+83.8	+1.4	-52.7	+16.8	-1.9
Gemini*	+88.3	+94.8	+0.7	-15.4	-0.7	+0.5

Table 6: **Effect of prompt explicitness on changed SUC prompts.** Values report $\Delta_{\text{EM}} = \text{EM}_{\text{explicit}} - \text{EM}_{\text{implicit}}$ for VLM results, averaged over HTML, LaTeX, and Markdown image renders. Positive values indicate that the explicit prompt achieved higher exact-match accuracy, while negative values indicate that the implicit prompt improved exact-match accuracy. Qwen3-VL-30B* denotes Qwen3-VL-30B-A3B-IT, and Gemini* denotes Gemini-3-Flash-Preview.

SR errors separate into fidelity and usability failures. Across models and formats, GRITS-Topology is consistently higher than GRITS-Content, showing that models recover broad table layout more reliably than exact cell text. Strong VLMs such as Qwen3, InternVL3.5, and GPT-5.2 obtain high topology scores across most source and target formats, but content scores drop more often, especially for LaTeX targets. This indicates that SR failures are not only structural alignment errors, but also include cell-text degradation during reconstruction.

Usability exposes syntax-level failures. Tables 8 and 13 show that high GRITS does not always imply usable output. Strong open VLMs usually produce valid HTML and Markdown, but LaTeX usability is less stable. For example, Qwen3-VL-30B reaches perfect usability for HTML and Markdown targets, while its LaTeX usability ranges from 0.77 to 0.95. This separates unusable syntax from usable but inaccurate reconstructions. We therefore report zero-penalized GRITS in Table 18, where unusable outputs receive zero before averaging.

LaTeX and content preservation remain the main bottlenecks. LaTeX reconstruction is difficult in two ways: models must recover the table structure and also produce compilable syntax. This

	HTML image						Markdown image						LaTeX image					
	GriTS-Topology			GriTS-Content			GriTS-Topology			GriTS-Content			GriTS-Topology			GriTS-Content		
Models	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**
Open Models																		
SmolVLM2-2.2B	0.87	0.86	0.81	0.74	0.74	0.69	0.66	0.79	0.78	0.57	0.68	0.68	0.79	0.85	0.57	0.65	0.69	0.47
Gemma3-12B	0.94	0.95	0.91	0.79	0.79	0.76	0.95	0.96	0.92	0.80	0.81	0.77	0.94	0.95	0.90	0.78	0.77	0.75
Gemma3-27B	0.97	0.97	0.94	0.86	0.85	0.82	0.98	0.97	0.94	0.85	0.85	0.81	0.97	0.96	0.94	0.83	0.82	0.81
InternVL3.5-14B	**0.99**	0.99	**0.96**	0.95	0.94	**0.92**	**0.99**	0.98	0.96	0.93	0.93	0.91	0.96	0.97	0.95	0.93	0.91	0.93
InternVL3.5-30B	0.98	0.99	0.95	0.95	0.94	**0.92**	**0.99**	0.99	0.96	0.93	0.94	0.92	0.96	0.97	0.95	0.93	0.92	0.92
Qwen3-VL-8B	**0.99**	**1.00**	0.95	**0.98**	**0.98**	**0.92**	**0.99**	**1.00**	**0.97**	**0.97**	**0.98**	**0.93**	**0.98**	**0.98**	**0.97**	**0.95**	**0.95**	**0.95**
Qwen3-VL-30B	0.98	0.99	0.84	**0.98**	**0.98**	0.80	**0.99**	**1.00**	**0.97**	**0.97**	**0.98**	**0.93**	0.97	**0.98**	0.96	**0.95**	**0.95**	**0.95**
Ministral3-14B	0.98	0.95	0.95	0.94	0.90	**0.92**	0.98	0.95	0.95	0.92	0.90	0.90	0.95	0.93	0.93	0.87	0.86	0.88
LLaVA1.6-7B	0.70	0.66	0.05	0.43	0.40	0.02	0.70	0.54	0.04	0.46	0.35	0.02	0.71	0.58	0.05	0.44	0.37	0.03
LLaVA1.6-13B	0.66	0.81	0.23	0.46	0.51	0.15	0.62	0.75	0.22	0.45	0.51	0.15	0.65	0.76	0.27	0.46	0.49	0.19
Proprietary Models																		
GPT-5.2	0.98	0.98	0.81	0.97	0.94	0.78	0.98	0.99	0.95	0.96	0.97	0.91	**0.98**	0.97	0.93	0.94	0.94	0.89
Gemini-3-Flash	0.05	0.96	0.65	0.05	0.94	0.64	0.86	0.97	0.51	0.85	0.96	0.51	0.65	0.93	0.58	0.63	0.91	0.57
Table-specialised Models																		
TableLLaVA-7B	0.73	0.71	0.68	0.33	0.33	0.31	0.73	0.72	0.70	0.43	0.43	0.41	0.58	0.57	0.54	0.29	0.29	0.28

Table 7: **Structure Reconstruction (SR) from table images.** Models receive table images rendered from HTML, Markdown, or LaTeX sources and reconstruct them into three target structural formats (HTML/Markdown/LaTeX). We report GriTS-Topology (structural fidelity) and GriTS-Content (cell-text fidelity). Higher scores indicate better reconstruction. Best scores per column (rounded to two decimals) are shown in bold.

	HTML image			Markdown image			LaTeX image		
Models	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**
Open Models									
Gemma3-27B-IT	**1.00**	**1.00**	0.89	**1.00**	**1.00**	0.88	**1.00**	**1.00**	0.87
Qwen3-VL-30B-A3B-IT	**1.00**	**1.00**	0.77	**1.00**	**1.00**	0.89	**1.00**	**1.00**	0.95
LLaVA1.6-Vicuna-13B	0.99	0.90	0.00	0.93	0.84	0.00	0.99	0.90	0.01
TableLLaVA-v1.5-7B	**1.00**	0.96	0.71	**1.00**	0.97	0.74	**1.00**	0.98	0.80
Proprietary Models									
GPT-5.2	0.99	**1.00**	**0.94**	0.99	**1.00**	**0.96**	**1.00**	**1.00**	**0.98**
Gemini-3-Flash-Preview	0.06	0.99	0.76	0.91	**1.00**	0.80	0.72	0.99	0.75

Table 8: **Output usability for Structure Reconstruction (SR).** Values report the fraction of syntactically usable reconstructed outputs across source render formats and target output formats. Best scores per column are shown in bold.

is most visible for weaker VLMs such as LLaVA variants, where LaTeX usability is near zero despite non-trivial HTML or Markdown usability. Among open models, Qwen3-VL-8B is highly competitive and often exceeds larger models on both topology and content, while TableLLaVA remains much weaker. Overall, modern VLMs can recover table topology reliably from images, but exact content preservation and syntactically usable LaTeX generation remain challenging.

6 Conclusion and Future Work

We introduce TABVERSE, a controlled multimodal table benchmark for studying representation effects in table understanding. TABVERSE provides aligned HTML, Markdown, and LaTeX representations with corresponding rendered images, category and difficulty tags, and a 700-sample balanced evaluation set drawn from five TableQA sources. This setup supports matched evaluation of QA, SUC, and SR across text and image inputs while keeping table content fixed.

Our results show that table representation strongly affects model behavior. Structured text often outperforms rendered images, especially for structure-sensitive tasks, while HTML is often the most robust format for text inputs and LaTeX remains challenging. In QA, verification questions are easier than Multi-Item Lookup and Aggregation/Arithmetic. In SUC, models handle column-oriented subtasks better than row and cell indexing, and header/indexing conventions can substantially shift answers. In SR, strong VLMs recover broad table layout reliably, but exact cell content and syntactically usable LaTeX generation remain difficult. Overall, scaling alone does not guarantee better table understanding.

Future work can extend TABVERSE to more realistic document settings, such as PDFs with noisy or multi-table layouts. It can also study methods that make models more consistent across formats and modalities. explore representation-aware decoding to improve structural consistency.

Limitations

TABVERSE targets controlled cross-format and cross-modality evaluation, so we make a few scope choices. We build the benchmark from five established English TableQA sources (FEVEROUS, HYBRIDQA, SQA, TABFACT, WIKITQ) and focus on single-table questions answerable from the table alone; extending coverage to additional languages, scripts, and document-level settings is a natural next step. For VLM-Image, we render table images from clean markup under a standardized layout. This keeps the image inputs aligned across HTML, Markdown, and LaTeX and helps us isolate representation effects, but it does not capture noise from scanned or photographed documents.

Ethical Statement and Broad Impact

TABVERSE is constructed from publicly available datasets: FEVEROUS, HYBRIDQA, SQA, TABFACT, and WIKITQ. The benchmark is intended for academic research on multimodal and cross-format table understanding. We use released table-question pairs and do not add any private or proprietary data. Because TABVERSE builds on existing datasets, it may inherit domain, linguistic, or annotation biases present in the original sources. TABVERSE aims to support transparent and reproducible evaluation of LLMs and VLMs, and is not designed for commercial deployment or decision-making in sensitive domains.

References

Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. FEVEROUS: Fact Extraction and VERification Over Unstructured and Structured information. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *Preprint*, arXiv:2308.12966.

Ekaterina Borisova, Fabio Barth, Nils Feldhus, Raia Abu Ahmad, Malte Ostendorff, Pedro Ortiz Suarez, Georg Rehm, and Sebastian Möller. 2025. Table understanding and (multimodal) LLMs: A cross-domain case study on scientific vs. non-scientific data. In *Proceedings of the 4th Table Representation Learning Workshop*, pages 109–142, Vienna, Austria. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. 2023. Sparks of artificial general intelligence: Early experiments with GPT-4. *Preprint*, arXiv:2303.12712.

Leiyuan Chen, Chengsong Huang, Xiaoqing Zheng, Jinshu Lin, and Xuanjing Huang. 2023. TableVLM: Multi-modal pre-training for table structure recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2437–2449, Toronto, Canada. Association for Computational Linguistics.

Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Yang Wang, and William W. Cohen. 2021a. Open question answering over tables and text. In *International Conference on Learning Representations*.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020a. TabFact: A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations*, ICLR'20.

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

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.

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)*, pages 1094–1110, Dublin, Ireland. Association for Computational Linguistics.

Naihao Deng and Rada Mihalcea. 2025. Rethinking table instruction tuning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 21757–21780, Vienna, Austria. Association for Computational Linguistics.

Naihao Deng, Zhenjie Sun, Ruiqi He, Aman Sikka, Yulong Chen, Lin Ma, Yue Zhang, and Rada Mihalcea. 2024. Tables as texts or images: Evaluating the table reasoning ability of LLMs and MLLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 407–426, Bangkok, Thailand. Association for Computational Linguistics.

Heng Gong, Yawei Sun, Xiaocheng Feng, Bing Qin, Wei Bi, Xiaojiang Liu, and Ting Liu. 2020. TableGPT: Few-shot table-to-text generation with table structure reconstruction and content matching. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1978–1988, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Google. 2024. Gemini 3 Flash model card. *Google*.

Yutong Guo, Wanying Wang, Yue Wu, Zichen Miao, and Haoyu Wang. 2025. TALENT: Table VQA via augmented language-enhanced natural-text transcription. In *2025 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 1404–1410. IEEE.

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.

Mohit Iyyer, Wen-tau Yih, and Ming-Wei Chang. 2017. Search-based neural structured learning for Sequential Question Answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1821–1831, Vancouver, Canada. Association for Computational Linguistics.

Yoonsik Kim, Moonbin Yim, and Ka Yeon Song. 2024. TableVQA-Bench: A visual question answering benchmark on multiple table domains. *ArXiv preprint*, abs/2404.19205.

Da Li, Keping Bi, Jiafeng Guo, Wei Yuan, Fan Yang, Tingting Gao, and Xueqi Cheng. 2026. Beyond text-only: Towards multimodal table retrieval in open-world. In *The Fourteenth International Conference on Learning Representations*.

Liyao Li, Jiaming Tian, Hao Chen, Wentao Ye, Chao Ye, Haobo Wang, Ningtao Wang, Xing Fu, Gang Chen, and Junbo Zhao. 2025. LongTableBench: Benchmarking long-context table reasoning across real-world formats and domains. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 11927–11965, Suzhou, China. Association for Computational Linguistics.

Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, and Zhoujun Li. 2020. TableBank: Table benchmark for image-based table detection and recognition. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1918–1925, Marseille, France. European Language Resources Association.

Peng Li, Yeye He, Dror Yashar, Weiwei Cui, Song Ge, Haidong Zhang, Danielle Rifinski Fainman, Dongmei Zhang, and Surajit Chaudhuri. 2024. Table-GPT: Table fine-tuned GPT for diverse table tasks. *Proc. ACM Manag. Data*, 2(3).

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, Alexandre Sablayrolles, Amélie Héliou, Amos You, Andy Ehrenberg, Andy Lo, Anton Eliseev, Antonia Calvi, Avinash Sooriyarachchi, Baptiste Bout, and 101 others. 2026. Ministral 3. *Preprint*, arXiv:2601.08584.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26286–26296.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Boammani Aser Lompo and Marc Haraoui. 2025. Visual-TableQA: Open-domain benchmark for reasoning over table images. *Preprint*, arXiv:2509.07966.

Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Vaibhav Srivastav, Joshua Lochner, Hugo Larcher, Mathieu Morlon, Lewis Tunstall, Leandro von Werra, and Thomas Wolf. 2025. SmolVLM: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*.

Suyash Vardhan Mathur, Jainit Sushil Bafna, Kunal Kartik, Harshita Khandelwal, Manish Shrivastava, Vivek Gupta, Mohit Bansal, and Dan Roth. 2024. Knowledge-aware reasoning over multimodal semi-structured tables. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14054–14073, Miami, Florida, USA. Association for Computational Linguistics.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński,

Hailey Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, and Dragomir Radev. 2022. FeTaQA: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.

Van-Quang Nguyen and Takayuki Okatani. 2026. CoReTab: Improving multimodal table understanding with code-driven reasoning. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6498–6523, Rabat, Morocco. Association for Computational Linguistics.

OpenAI. 2025. GPT-5 system card. *OpenAI*.

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'20, 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.

Josselin Somerville Roberts, Tony Lee, Chi Heem Wong, Michihiro Yasunaga, Yifan Mai, and Percy Liang. 2025. Image2Struct: Benchmarking structure extraction for vision-language models. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Anshul Singh, Chris Biemann, and Jan Strich. 2025. MTabVQA: Evaluating multi-tabular reasoning of language models in visual space. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 19866–19891, Suzhou, China. Association for Computational Linguistics.

Ananya Singha, José Cambronero, Sumit Gulwani, Vu Le, and Chris Parnin. 2023. Tabular representation, noisy operators, and impacts on table structure understanding tasks in LLMs. In *Proceedings of the Table Representation Learning Workshop at Neural Information Processing Systems (NeurIPS) 2023*.

Brandon Smock, Rohith Pesala, and Robin Abraham. 2022. PubTables-1M: Towards comprehensive table extraction from unstructured documents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4634–4642.

Brandon Smock, Rohith Pesala, and Robin Abraham. 2023. GriTS: Grid Table Similarity metric for table structure recognition. In *Document Analysis and Recognition - ICDAR 2023: 17th International Conference, San José, CA, USA, August 21–26, 2023, Proceedings, Part V*, page 535–549, Berlin, Heidelberg. Springer-Verlag.

Aofeng Su, Aowen Wang, Chao Ye, Chen Zhou, Ga Zhang, Gang Chen, Guangcheng Zhu, Haobo Wang, Haokai Xu, Hao Chen, Haoze Li, Haoxuan Lan, Jiaming Tian, Jing Yuan, Junbo Zhao, Junlin Zhou, Kaizhe Shou, Liangyu Zha, Lin Long, and 14 others. 2024. TableGPT2: A large multimodal model with tabular data integration. *Preprint*, arXiv:2411.02059.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024a. Table Meets LLM: Can large language models understand structured table data? A benchmark and etabmpirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, page 645–654, New York, NY, USA. Association for Computing Machinery.

Yuan Sui, Jiaru Zou, Mengyu Zhou, Xinyi He, Lun Du, Shi Han, and Dongmei Zhang. 2024b. TAP4LLM: Table provider on sampling, augmenting, and packing semi-structured data for large language model reasoning. In *Findings of the Association for Computational Linguistics*, EMNLP'24, pages 10306–10323, Miami, Florida, USA. Association for Computational Linguistics.

Alon Talmor, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannaneh Hajishirzi, and Jonathan Berant. 2021. Multimodal{qa}: complex question answering over text, tables and images. In *International Conference on Learning Representations*.

Gemma Team. 2025a. Gemma 3 Technical Report.

Qwen Team. 2025b. Qwen3 Technical Report. *Preprint*, arXiv:2505.09388.

Prasham Yatinkumar Titiya, Jainil Trivedi, Chitta Baral, and Vivek Gupta. 2025. MMTBENCH: A unified benchmark for complex multimodal table reasoning. *Preprint*, arXiv:2505.21771.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aur'elien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

Lanrui Wang, Mingyu Zheng, Hongyin Tang, Zheng Lin, Yanan Cao, Jingang Wang, Xunliang Cai, and Weiping Wang. 2026. NeedleinATable: Exploring long-context capability of large language models towards long-structured tables. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, and 1 others. 2025. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*.

Jian Wu, Linyi Yang, Dongyuan Li, Yuliang Ji, Manabu Okumura, and Yue Zhang. 2025a. MMQA: Evaluating LLMs with multi-table multi-hop complex questions. In *The Thirteenth International Conference on Learning Representations*.

Pengzuo Wu, Yuhang Yang, Guangcheng Zhu, Chao Ye, Hong Gu, Xu Lu, Ruixuan Xiao, Bowen Bao, Yijing He, Liangyu Zha, Wentao Ye, Junbo Zhao, and Haobo Wang. 2025b. RealHiTBench: A comprehensive realistic hierarchical table benchmark for evaluating LLM-based table analysis. In *Findings of the Association for Computational Linguistics*, ACL'25, pages 7105–7137, Vienna, Austria. Association for Computational Linguistics.

Junjie Xing, Yeye He, Mengyu Zhou, Haoyu Dong, Shi Han, Lingjiao Chen, Dongmei Zhang, Surajit Chaudhuri, and H. V. Jagadish. 2025. MMTU: A massive multi-task table understanding and reasoning benchmark. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Zhuoyan Xu, Haoyang Fang, Boran Han, Bonan Min, Bernie Wang, Cuixiong Hu, and Shuai Zhang. 2026. Efficient table retrieval and understanding with multimodal large language models. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 4327–4340, Rabat, Morocco. Association for Computational Linguistics.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2024. Qwen2.5 technical report. *ArXiv preprint*, abs/2412.15115.

Bohao Yang, Yingji Zhang, Dong Liu, André Freitas, and Chenghua Lin. 2025. Does table source matter? benchmarking and improving multimodal scientific table understanding and reasoning. *Preprint*, arXiv:2501.13042.

Liangyu Zha, Junlin Zhou, Liyao Li, Rui Wang, Qingyi Huang, Saisai Yang, Jing Yuan, Changbao Su, Xiang Li, Aofeng Su, and 1 others. 2023. TableGPT: Towards unifying tables, nature language and commands into one GPT. *arXiv preprint arXiv:2307.08674*.

Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. 2024. TableLlama: Towards open large generalist models for tables. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6024–6044, Mexico City, Mexico. Association for Computational Linguistics.

Xiaokang Zhang, Sijia Luo, Bohan Zhang, Zeyao Ma, Jing Zhang, Yang Li, Guanlin Li, Zijun Yao, Kangli Xu, Jinchang Zhou, Daniel Zhang-Li, Jifan Yu, Shu Zhao, Juanzi Li, and Jie Tang. 2025. TableLLM: Enabling tabular data manipulation by LLMs in real office usage scenarios. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10315–10344, Vienna, Austria. Association for Computational Linguistics.

Mingyu Zheng, Xinwei Feng, Qingyi Si, Qiaoqiao She, Zheng Lin, Wenbin Jiang, and Weiping Wang. 2024. Multimodal table understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9102–9124, Bangkok, Thailand. Association for Computational Linguistics.

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 (Volume 1: Long Papers)*, pages 3277–3287, Online. Association for Computational Linguistics.

Junnan Zhu, Jingyi Wang, Bohan Yu, Xiaoyu Wu, Junbo Li, Lei Wang, and Nan Xu. 2025. TableEval: A real-world benchmark for complex, multilingual, and multi-structured table question answering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 7126–7146, Suzhou, China. Association for Computational Linguistics.

Yingjie Zhu, Xuefeng Bai, Kehai Chen, Yang Xiang, Youcheng Pan, Xiaoqiang Zhou, and Min Zhang. 2026. Decoupling skeleton and flesh: Efficient multimodal table reasoning with disentangled alignment and structure-aware guidance. *Preprint*, arXiv:2602.03491.

A Implementation and Evaluation Details

A.1 Model list

We evaluate a set of text-only language models (LLMs) and vision–language models (VLMs), including both general-purpose and table-oriented models. The evaluated models are:

Language Models (text-only): *Qwen2.5-7B-Instruct*, *Qwen3-30B-A3B-Instruct*, *TableGPT2-7B*, *TAMA-QWen3*.

Vision–Language Models. *SmolVLM2-2.2B-Instruct*, *Gemma-3-12B-IT*, *Gemma-3-27B-IT*, *InternVL3.5-14B*, *InternVL3.5-30B-A3B*, *Qwen3-VL-8B-Instruct*, *Qwen3-VL-30B-A3B-Instruct*, *Ministral-3-14B-Instruct*, *LLaVA-1.6-7B**, *LLaVA-1.6-13B**, *TableLLaVA-v1.5-7B**, *GPT-5.2*, *Gemini-3-Flash-Preview*.

A.2 Pipelines and configurations

We evaluate: (i) *LLM* (structural-text) for text-only models, (ii) *VLM-Image* (rendered table images) for VLMs, and (iii) *VLM-Text* (structural-text) for VLMs when the model interface supports text-only operation. For starred models (*), we report results only for the configurations that are supported reliably by the model interface and context window.

A.3 Context length and coverage

Structural-text evaluation includes the full table markup in the prompt and can require long context. When a prompt exceeds a model's supported context length (or fails to run reliably), we mark that instance as out of coverage for that model–configuration and compute metrics over the remaining evaluable instances. We report per-model coverage statistics alongside results. We ran open-weight model experiments on NVIDIA A100 SXM GPUs, using GPU execution for inference and supervised fine-tuning. Closed-model experiments were conducted through the OpenAI and Google Gemini APIs.

A.4 Decoding and post-processing

All evaluations use zero-shot prompting with greedy decoding (`temperature=0`, `top_p=1`) and task-specific output limits. We apply minimal normalization for scoring consistency (e.g., stripping boilerplate prefixes, whitespace normalization, and simple label extraction for verification tasks). Prompt templates and exact decoding limits are listed in Appendix C.

A.5 Additional Metric Details

For QA, we report exact-match accuracy after light normalization. For single-answer questions, a prediction is counted correct if it exactly matches the gold answer after normalization, or if the gold answer appears as a leading complete answer span. For multi-item lookup questions, we split predictions on commas, semicolons, or newlines and require the normalized set of predicted items to match the gold set. We also report a relaxed QA diagnostic that counts a single-answer prediction as correct when the gold answer appears as a complete normalized span in the prediction. For numeric answers, we extract numeric strings and compare them after comma normalization.

For SUC, exact match remains the main metric. We additionally report Field Accuracy and Relaxed Accuracy to diagnose structured-answer failures. Field Accuracy compares pipe-separated fields position-wise, giving partial credit when only some fields are correct:

$$\mathrm{FieldAcc}_i = \frac{1}{K_i} \sum_{k=1}^{K_i} \mathbb{1}[\hat{y}_{i,k} = y_{i,k}],$$

where K_i is the number of gold fields. Relaxed Accuracy measures how many gold fields appear somewhere in the model output:

$$\mathrm{RelaxedAcc}_i = \frac{1}{K_i} \sum_{k=1}^{K_i} \mathbb{1}[y_{i,k} \in \hat{y}_i].$$

These diagnostics help distinguish fully wrong predictions from outputs that contain the right values but use the wrong format or field order.

For SR, output usability is computed separately for each requested target format. HTML outputs are considered usable when a table can be parsed and at least one cell can be extracted. Markdown outputs are rendered with a table extension and then checked for a recoverable table. LaTeX outputs are considered usable when the generated table compiles successfully with a fixed wrapper. These checks are applied without repairing malformed model outputs.

B Dataset Details

B.1 Source splits, filtering, and overlap checks

We construct the raw pool from the official held-out split for each source dataset (test when available; otherwise dev) and filter to retain single-table, table-grounded instances. For datasets with mixed table/passage supervision (e.g., HYBRIDQA and FEVEROUS), we drop instances whose gold evidence requires non-table context or multiple tables. We also check for overlaps against the corresponding training splits and remove duplicated question–table pairs when detected.

Dataset	Split used	Tagged pool	#Q	#Tables
FEVEROUS	dev	794		525
HYBRIDQA	dev	1608		1608
TABFACT	test	1695		1695
SQA	test	1000		185
WIKITQ	unseen tables	1000		421
Total	–	6097		4434

Table 9: Tagged pool composition after filtering and normalization. #Tables counts unique underlying tables.

B.2 Tagged pool normalization

The source datasets use different schemas for tables and supervision. We normalize each example into a common JSON format (table content, question, answer/label, and metadata) and assign a stable `table_id` to each unique table so that multiple questions can reference the same underlying table.

B.3 Format conversion to HTML, Markdown, and LaTeX

We convert each table into three structural formats. We adapt conversion utilities released with Sui et al. (2024a) and extend them to (i) target our held-out splits, (ii) enforce consistent row/-column ordering across formats, and (iii) produce syntactically valid outputs under dataset-specific artifacts (e.g., missing values and special characters). We generate HTML markup with standard `<table>`/`<tr>`/`<th>`/`<td>` tags, Markdown tables with pipe-delimited syntax, and compilable LaTeX `tabular` code with appropriate escaping.

B.4 Rendering pipelines

We render a table image from each structural representation under a standardized layout (font size, padding, width). We render HTML tables in a controlled browser environment, convert Markdown to HTML before rendering, and compile LaTeX to PDF before converting to PNG. Because we render images from the generated markup, the rendered images are aligned with the textual tables by construction.

B.5 Question category taxonomy

We use seven question categories for analysis and stratification (the same set used in the Results section). For each category, we provide a short definition and one example prompt in Appendix C.

C Prompts

System Prompt
You are an expert at classifying table-question pairs into Question Categories. You must follow the rules exactly and output exactly ONE line in the required format.

Input
TABLE (as JSON with header + rows):
`{table_json}`
QUESTION
`{query}`

TASK
Assign exactly ONE Question Category using the definitions below.

CRITICAL RULE (TABLE-REQUIREDNESS)
A question can be assigned a predefined Question Category ONLY IF the table is REQUIRED to produce the final answer.
External knowledge policy:
You MAY use external knowledge to:

- Map a description or condition to an entity present in the table (e.g., "country with population 67.02 million" → "France")
- Resolve aliases, real-world facts, or descriptive constraints ONLY for the purpose of identifying the correct row(s)

You MUST NOT:

- Output a final answer that does not come directly from table cells
- Compute the final answer using non-table facts
- Label as answerable if the table is not necessary

If the final answer can be obtained without using the table at all, output:
`None of the above – Not table-required`
If the required answer value does not exist in or cannot be computed from the table cells, output:
`None of the above – Missing answer attribute in table`

PROCEDURE
Step A — Table-Requiredness Check

1. Identify what the question asks for:
   - single value
   - list/set
   - number
   - yes/no
2. Identify which table cell(s) must be read or aggregated to produce the final answer.
3. Verify that the final answer is directly read from or computed using ONLY table cells.
4. If the table is not needed to produce the final answer → None of the above — Not table-required
5. If the answer value is not present or cannot be computed from table cells → None of the above — Missing answer attribute in table

Otherwise, proceed to classification.

QUESTION CATEGORIES (Choose EXACTLY ONE)

- **Simple Lookup** Identify ONE row and read ONE cell. (No filtering beyond locating that row.)
- **Conditional Lookup** Apply one or more conditions to select row(s), then read ONE resulting value.
- **Multi-Item Lookup** Return multiple values/rows from the table (a list/set).
- **Aggregation / Counting / Arithmetic** Compute a number from table values (count/sum/avg/difference/ratio/percent/etc.).
- **Comparison & Extremum** Choose max/min/earliest/latest by comparing table values.
- **Single-step Binary Verification** Verify one statement directly using the table.
- **Multi-hop Binary Verification** Verify a statement requiring multiple reasoning steps across the table.

OUTPUT FORMAT (STRICT)
Output exactly ONE line and nothing else:
`Question Category: <CATEGORY NAME>`
OR (if not answerable under rules):
`Question Category: None of the above`

Figure 2: Prompt used to classify table-question pairs into structured reasoning categories.

General Question Answering Prompt
Look at the given table and answer the following question directly. Do not include introductions, explanations, or extra text. Provide only the exact, precise final answer.
`{query}`

Binary Verification Prompt
Look at the given table and answer the following question with only a single digit: 1 if the statement is true, 0 if the statement is false. Do not include any explanations or extra text.
`{query}`

Figure 3: Prompts used for table-based question answering and binary verification tasks.

HTML Generation Prompt
Generate the complete HTML code that exactly represents this image. Provide only the code without any explanations.

LaTeX Generation Prompt
Generate the complete LaTeX code that exactly represents this image. Provide only the code without any explanations.

Markdown Generation Prompt
Generate the complete Markdown code that exactly represents this image. Provide only the code without any explanations.

Figure 4: Prompts used for generating structured table representations from images in HTML, LaTeX, and Markdown formats.

table_partition
What is the **first cell value** (not including headers) of the given table? What is the **last cell value** (not including headers) of the given table? Answer questions one by one and use | to split the answer. Answer the question without having any introduction or explanations.

table_first_cell
What is the **first cell value** (not including headers) of the given table? Answer the question without having any introduction or explanations.

table_last_cell
What is the **last cell value** (not including headers) of the given table? Answer the question without having any introduction or explanations.

size_detection
How many rows in the table? How many columns in the table? Answer the questions one by one and use | to split the answer. Answer the question without having any introduction or explanations.

number_of_rows
How many rows in the table? Answer the question without having any introduction or explanations.

number_of_columns
How many columns in the table? Answer the question without having any introduction or explanations.

cell_lookup
Row/column indices start at 0 (top-left is 0|0). What is the position of the cell value `{cell_value}`? Use row index and column index to answer. Use | to split the answer. Answer the question without having any introduction or explanations.

reverse_lookup
Row/column indices start at 0 (top-left is 0|0). What is the cell value of row index `{reverse_lookup_row}`, column index `{reverse_lookup_col}` ? Only output the cell value without other information. Answer the question without having any introduction or explanations.

column_retrieval
Row/column indices start at 0 (top-left is 0|0). What is the column name with the index `{column_idx}` of the given table image? Only give the column name without any explanation. Answer the question without having any introduction or explanations.

row_retrieval
Row/column indices start at 0 (top-left is 0|0). What are the cell values of the `{row_idx}` row in following table? Only list the cell values one by one using | to split the answers. Answer the question without having any introduction or explanations.

Figure 5: Prompts used for evaluating structural understanding

D Result Discussion

D.1 TaskQA: Additional Analyses

This appendix provides additional TaskQA analyses that support the main results. We include modality gaps, strict-vs-relaxed matching, question-category results, and Easy/Hard breakdowns.

Modality gap. Figure 6 reports Δ = VLM-Text avg − VLM-Image avg, averaged over HTML, LaTeX, and Markdown. Positive values mean that structured table text helps more than rendered images, while negative values mean that rendered images help more. The direction of the gap depends on the model family. Gemma-3 shifts toward structured text, with gains of +10.3 points for the 12B model and +7.0 points for the 27B model. In contrast, SmolVLM2 and InternVL3.5 shift toward rendered images. Gemini-3-Flash-Preview stays near zero, which is consistent with its stable performance across modalities in Table 3.

Strict vs relaxed matching. Table 14 reports both strict exact-match accuracy and relaxed accuracy. Relaxed accuracy counts a prediction as correct when the normalized gold answer appears as a complete answer span inside a longer response. This diagnostic does not replace strict EM. Instead, it separates answer retrieval from answer-only formatting. For example, Qwen3-VL-8B-IT is stronger under strict EM, while Qwen3-VL-30B-A3B-IT recovers substantially under relaxed matching, especially for structured table-text inputs. This indicates that some lower strict-EM scores are partly caused by verbose answer formatting rather than missing the answer completely.

Question categories. Figure 8 reports accuracy by question category, averaged over models and formats. Verification-style questions are easier across pipelines, while multi-item lookup and aggregation/counting questions are more difficult. This suggests that models handle binary or localized evidence better than questions requiring multiple retrieved items, counting, or arithmetic operations.

Easy vs Hard split. Figure 7 reports Easy and Hard accuracy for each model and pipeline, averaged over HTML, LaTeX, and Markdown. All pipelines show a clear drop from Easy to Hard questions. This confirms that the difficulty annotation captures a meaningful increase in reasoning or evidence-composition demands. We use this split as a diagnostic view of question difficulty, not as a model-independent definition of reasoning complexity.

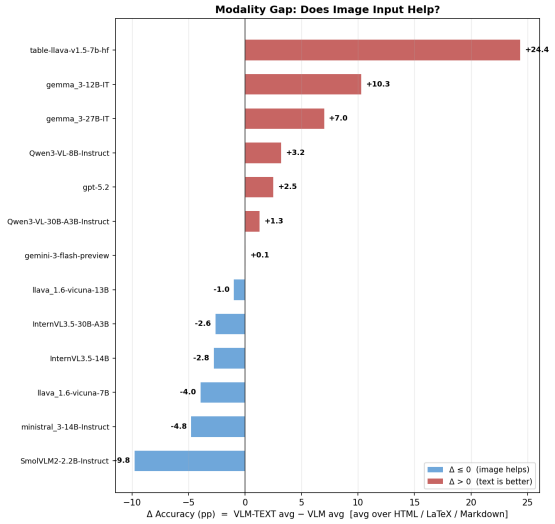

Figure 6: **TaskQA modality gap.** Δ accuracy (pp) = VLM-Text avg − VLM-Image avg, averaged over HTML/LaTeX/Markdown. Negative means images help more than text.

Evaluation notes. TaskQA is scored with strict exact-match accuracy in the main results. We apply the same normalization and post-processing to all models. Strict EM is intentionally conservative: answers with extra explanatory text are counted as incorrect even when they contain the gold answer. For this reason, Table 14 is included as a diagnostic relaxed-matching view. Models marked with * have shorter context windows; when they fail to return an answer on large-table cases, the output is counted as incorrect under the same scoring rule.

D.2 Structural Understanding Capability: Additional Analyses

This appendix provides additional SUC analyses that support Section 5.2. We include subtask-level difficulty, pipeline comparisons, diagnostic metrics, prompt-sensitivity results, format effects, and evaluation notes.

Subtask difficulty. Figure 10 shows that SUC difficulty is highly subtask-dependent. Column counting and column retrieval are consistently among the easiest subtasks across pipelines. In contrast, row retrieval, cell lookup, table partition-

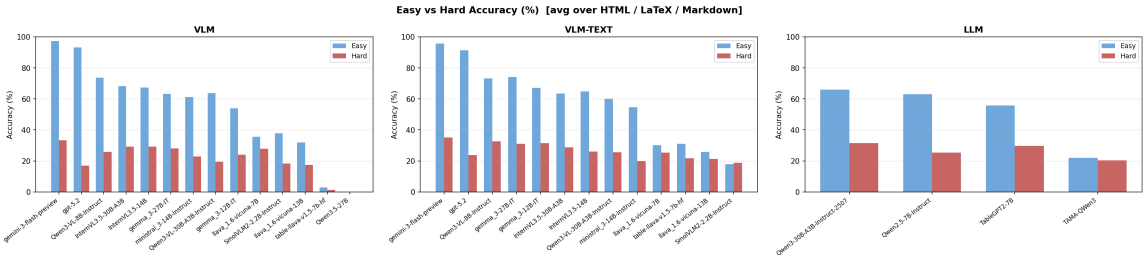

Figure 7: **TaskQA Easy vs Hard.** Easy and Hard exact-match accuracy per model and pipeline (VLM-Image / VLM-Text / LLM-Text), averaged over HTML/LaTeX/Markdown.

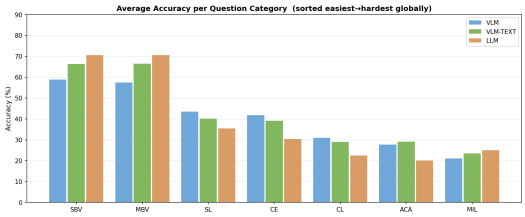

Figure 8: **TaskQA category averages.** Accuracy by question category, averaged over models and formats, shown per pipeline (VLM-Image / VLM-Text / LLM-Text).

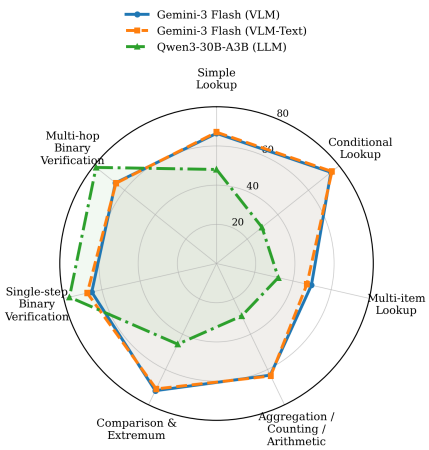

Figure 9: Category-wise Task QA accuracy for the strongest model from each pipeline.

ing, and size detection remain difficult. This pattern shows that models are better at identifying global column structure than at recovering precise row-level or coordinate-based structure.

Pipeline comparison. Figure 11 compares SUC accuracy across VLM-Image, VLM-Text, and LLM-Text. Structured text generally improves SUC over rendered images, especially for subtasks that depend on row boundaries and header handling. This is also visible in Tables 4, 15, and 17. For example, several models improve on row retrieval, reverse lookup, and size detection when the table structure is provided as text. However, text input does not remove the bottleneck entirely. Cell lookup and row retrieval remain difficult across pipelines, showing that SUC requires more than access to explicit table markup.

Field accuracy and relaxed accuracy diagnostics. Tables 5 and 16 show that strict EM can hide partial structural recovery. For multi-field tasks such as table partitioning, size detection, cell lookup, and row retrieval, a model may recover one part of the requested structure while missing another. Field Accuracy captures this behavior by scoring individual fields, while relaxed accuracy captures outputs that contain the correct answer but fail strict string matching. The large gaps between EM and Field Accuracy show that many errors are incomplete or shifted structural predictions rather than completely unrelated answers.

Prompt explicitness and header handling. Table 6 compares the explicit SUC prompt with an implicit prompt on selected VLM-Image subtasks. The explicit prompt states conventions such as excluding headers for first/last-cell tasks and using 0-indexed row/column coordinates for lookup and retrieval tasks. The implicit prompt removes these details. The largest differences appear on index-dependent subtasks, especially first-cell detection and table partitioning. This suggests that rendered-table models often confuse header rows with table body rows, producing answers under a different row-indexing convention. The effect is weaker for last-cell detection because the last cell is usually identified from the table end and is less affected by whether headers are counted. Reverse lookup shows a different pattern for some models, where the implicit prompt performs better. This confirms that SUC is sensitive not only to visual structure recognition, but also to how models interpret indexing and header conventions.

Format effects. Tables 4, 15, 17, and 12 show that format effects are present but not uniform. For VLM-Image, differences across HTML, LaTeX, and Markdown renders are usually smaller than the differences across models and subtasks. For text-input pipelines, format effects are more visible. HTML is often the safest structured-text format, especially for VLM-Text and LLM-Text, while LaTeX and Markdown can be less stable for some models. This suggests that SUC depends jointly on the input modality, the table representation, and the model's ability to follow row/column conventions.

Evaluation notes. We report strict exact-match accuracy as the main SUC metric because the tasks require exact structured outputs, such as row–column coordinates, table size fields, or ordered cell sequences. We use the same post-processing across models and formats. Field Accuracy and relaxed accuracy are reported only as diagnostics. They help identify partial structural recovery and answer-formatting issues, but they do not replace EM. For models with shorter context windows or weak instruction following, some failures come from incomplete outputs, extra explanations, or missing delimiters. These cases are counted as incorrect under strict EM to keep evaluation consistent across pipelines.

D.3 Structure Reconstruction: Additional Analyses

Validity-adjusted SR scores. Table 18 reports zero-penalized GRITS, where syntactically unusable outputs receive a score of zero before averaging. This combines the two SR failure modes: unusable target syntax and usable but inaccurate reconstruction. The gap between raw and zero-penalized scores is small for strong models on HTML and Markdown targets, showing that most remaining errors are fidelity errors rather than syntax failures. The gap is larger for LaTeX targets, especially for weaker VLMs and TableLLaVA, confirming that LaTeX failures often come from invalid or non-compilable outputs rather than only low structural similarity.

Format-pair difficulty. Figure 13 averages GRITS-Topology and GRITS-Content over all models for each input–output format pair. Markdown targets are slightly easier on average for both topology and content, while LaTeX targets are consistently harder. The lowest scores occur when the target is LaTeX, suggesting that the output format is a larger source of difficulty than the input render format.

Same-format versus cross-format reconstruction. Figure 14 compares same-format reconstruction with cross-format conversion for each model. Strong models show only small differences between the two settings, which indicates that cross-format reconstruction is not inherently much harder once the model can parse the table image and generate the target syntax. Larger gaps appear for weaker models, where errors are dominated by unstable parsing or invalid output generation rather than by format conversion alone.

E Error Analysis

Error analysis (TaskQA). Table 10 summarizes the main ways models miss TaskQA under exact match. On lookup-style questions, many errors come from selecting a reasonable-looking cell from the right column but the wrong row (or missing a filter in Conditional Lookup), which also shows up in Comparison/Extremum questions when the model compares within the wrong subset. Multi-Item Lookup adds two separate issues: models often miss one item (or add an extra one), and even when they list the right items, small differences in separators can break matching after normalization. For Aggregation/Arithmetic, we mostly see small counting or summation mistakes, which usually happen when the model skips or double-counts one qualifying row. Verification errors typically appear as 0/1 label flips, often around negation or multi-row conditions. Finally, exact match penalizes answers that include extra text before the answer token; this affects smaller models more often and also shows up when a model returns an explanation instead of a single answer span.

Error analysis (SUC). To understand low accuracy on the index-based SUC probes under exact match, we inspected outputs for strong proprietary VLMs (GPT-5.2, Gemini-3-Flash-Preview), a strong open VLM (Qwen3-VL-8B), and a table-specialised model (TableLLaVA-v1.5-7B), across VLM-Image and VLM-Text (and text-only baselines in LLM-Text). We keep the same post-processing and exact-match scoring for every model, so small differences in indexing conventions or answer templates count as errors.

Header offset explains many indexing/lookup errors, especially in VLM-Image. Our gold labels

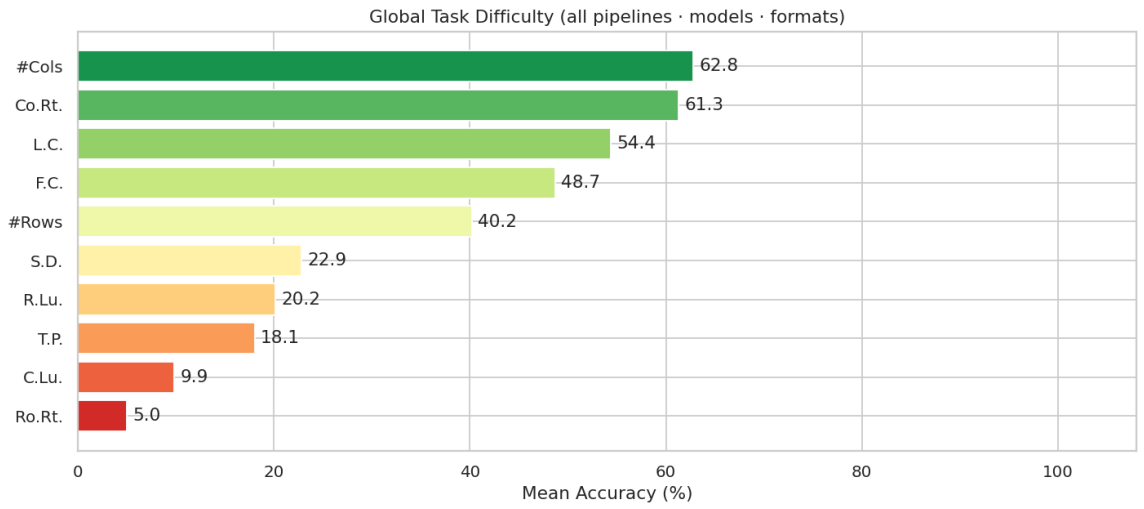

Figure 10: **SUC task difficulty.** We average exact-match accuracy over all models, pipelines, and formats for each SUC subtask. Higher is better.

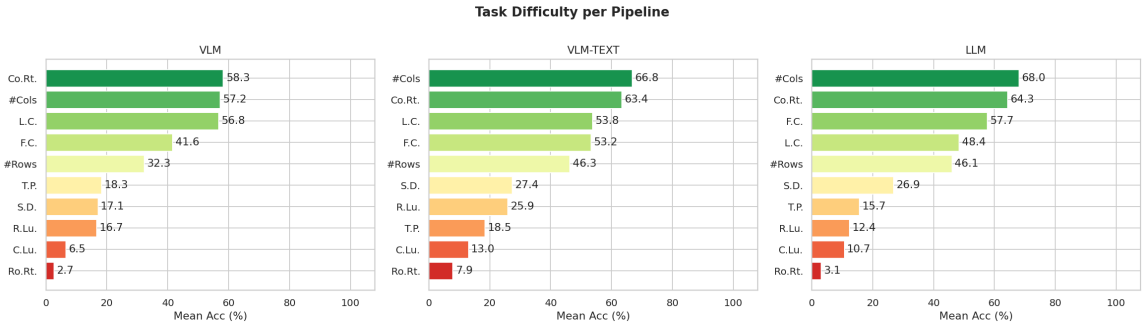

Figure 11: **Pipeline comparison.** We average exact-match accuracy per SUC subtask for VLM-Image, VLM-Text, and LLM-Text. Higher is better.

define indices over *data cells* (headers excluded). Under a fixed coordinate convention, some models treat the header as part of the indexable grid, which produces a consistent $+1$ row shift. When this happens, size-related probes report one extra row (e.g., gold S.D. = $10|3$ vs. prediction $11|3$), coordinate probes return row indices that are one larger than gold, and row retrieval often returns the row *one position above* the gold row for the same queried index. This pattern makes models look strong on boundary and column probes while scoring poorly on C.Lu. and Ro.Rt. under exact match. We keep one coordinate convention across pipelines to surface this sensitivity rather than tuning prompts separately per modality or per format.

Repeated values make coordinate probes harder than they appear. Many tables contain repeated values (especially short strings and common numbers), so more than one coordinate can look reasonable for C.Lu. and R.Lu. Models then either pick a different occurrence or return multiple coordinates. Exact match counts both cases as incorrect, even when the output remains consistent with the table content.

Ro.Rt. often fails on output formatting, not only row selection. Ro.Rt. requires emitting the full row as a pipe-separated string. Models sometimes add leading/trailing bars, change spacing, omit empty cells, or output Markdown-style rows. TableLLaVA also tends to produce multi-line structured outputs instead of a single row span. These differences fail exact match even when the chosen row is close.

Verbosity and uniform post-processing. Some models (especially TableLLaVA, and occasionally smaller VLMs) generate verbose answers that include the correct information inside extra text or inside a larger structured block. We intentionally keep one uniform post-processing rule for all models to evaluate end-to-end, machine-readable relia-

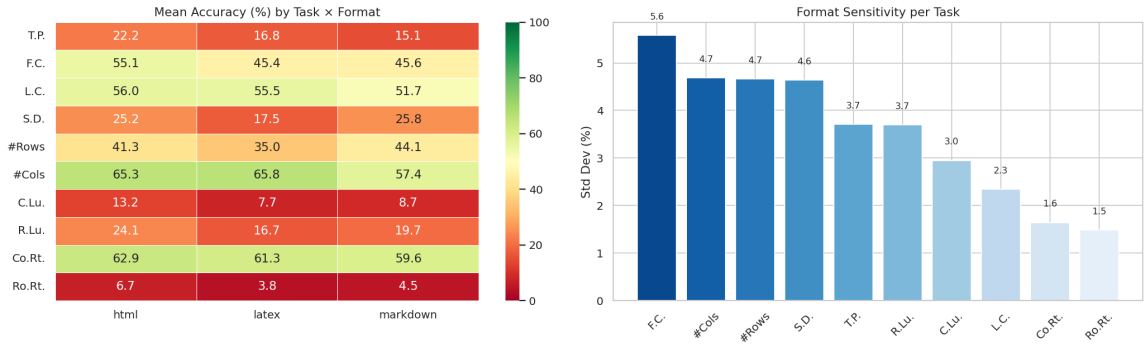

Figure 12: **Format sensitivity.** We show mean exact-match accuracy by SUC subtask and format (HTML, LaTeX, Markdown) and the variation across formats. Higher is better.

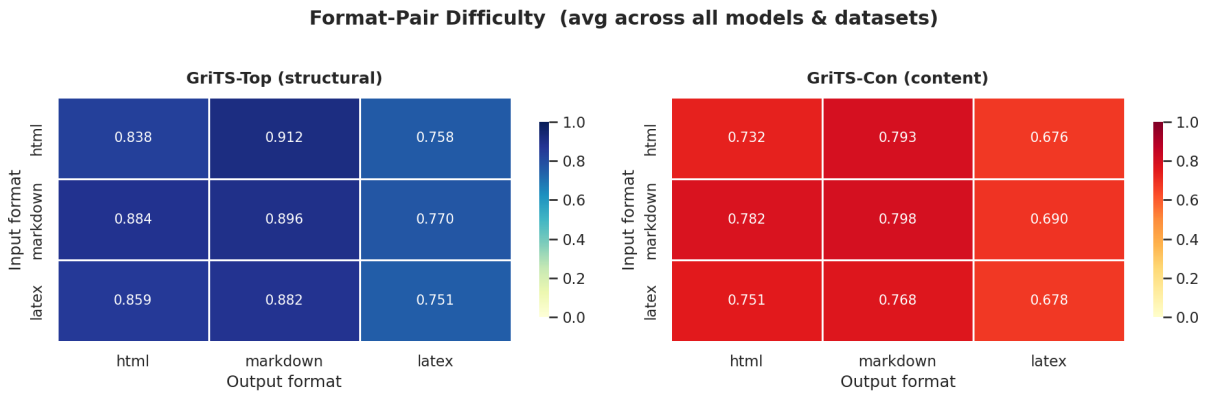

Figure 13: **Format-pair difficulty (SR).** We average GriTS-Topology and GriTS-Content over all models for each input→output format pair.

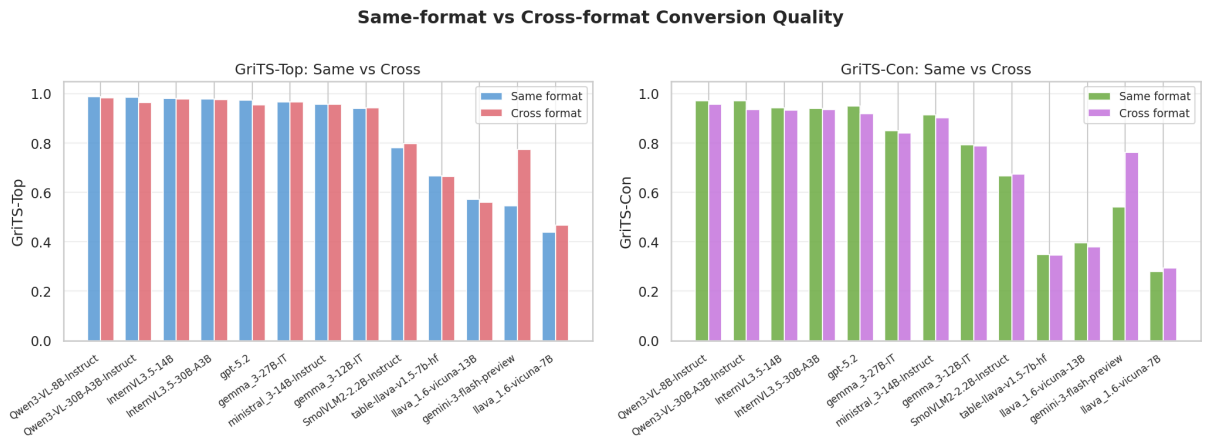

Figure 14: **Same-format vs cross-format SR.** We compare per-model averages for same-format reconstruction against cross-format conversion.

Error type	What it looks like	Common in
Wrong table cell	Prediction matches a plausible cell/header, but it comes from the wrong row/column (often satisfies only part of the condition).	Lookup / Conditional / Comparison
Off-table answer	Prediction does not match any table cell and does not equal the gold (e.g., generic "cannot determine" replies or free-form values).	VLM-Image (more)
Multi-item mismatch	Missing items, extra items, or mixed sets when the question expects a specific set of values.	Multi-Item Lookup
Multi-item formatting	Gold items appear in the output, but separators/punctuation break set matching (e.g., commas that belong to an entity name).	Multi-Item Lookup
Arithmetic near-miss	Wrong count/sum; off-by-one errors are common when one row is missed or double-counted.	Aggregation/Arithmetic
Binary label flip (0/1)	Model outputs a valid 0/1 token but flips the label relative to gold.	Verification
Answer not isolated	Gold answer appears in the output but not as the first token (e.g., extra prefix or short descriptor).	Smaller models; LLM-Text

Table 10: Common TaskQA failure modes under our exact-match scoring (light normalization; set match for Multi-Item Lookup).

Error type	What it looks like	Often affects
Header offset (+1 row)	Model treats the header row as part of the indexed grid (header becomes row 0), shifting row indices by 1.	S.D., #Rows, C.Lu., Ro.Rt.
Repeated-value ambiguity	The queried value appears multiple times; the model returns a different valid coordinate or lists several coordinates.	C.Lu., R.Lu.
Answer-template mismatch	Extra text, multi-line answers, or a delimiter different from the required template (e.g., "Row = 3, Col = 2").	C.Lu., S.D., #Rows, Ro.Rt.
Row serialization drift	Row retrieval returns the right row content but with small formatting differences (e.g., leading/trailing '|', missing cells, or Markdown-style rows).	Ro.Rt.
Verbose structured outputs	Model outputs a full Markdown table or explanation instead of a single required answer span.	Mostly Ro.Rt. (also C.Lu./S.D.)

Table 11: Common SUC error types under strict exact match.

bility. This choice can undercount models that do not follow the requested answer template without using model-specific extraction rules.

F Figures

The figures give a quick overview of the tasks and the evaluation setup. Figure 16 shows how the SUC tasks are grouped into four types: partitioning, size estimation, lookup, and retrieval. Figure 15 shows the structure recognition (SR) pipeline. A table x is turned into an image, the model predicts a structure x', and this prediction is compared with the original table.

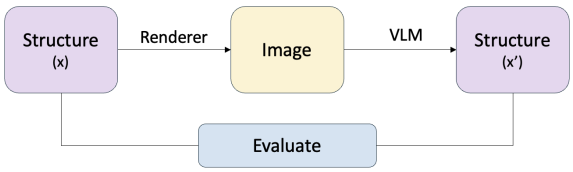

Figure 15: Pipeline for SR. A ground-truth table x is rendered into an image, the model predicts a structure x', and evaluation compares x' with x.

G Result Tables

This section presents detailed results for SUC tasks across different settings.

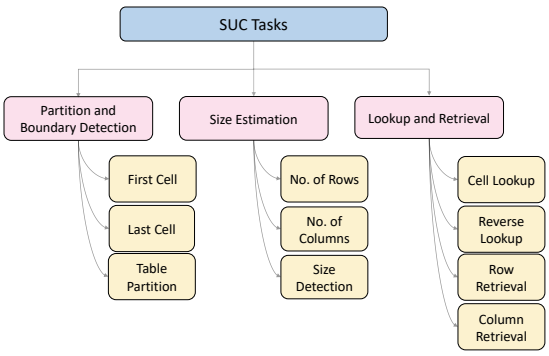

Figure 16: Taxonomy of SUC tasks. Ten tasks are grouped into partitioning, size estimation, lookup, and retrieval.

Models	Formats	T.P.	F.C.	L.C.	S.D.	# Rows	# Cols	C.Lu.	R.Lu.	Co.Rt.	Ro.Rt.
Open Models											
Qwen2.5-7B-Instruct	HTML	16.7	**95.1**	57.7	20.5	52.3	80.0	25.4	14.3	75.4	**5.7**
	LaTeX	11.1	44.4	54.8	3.3	34.3	**99.0**	7.3	5.2	61.4	5.4
	Markdown	5.9	55.6	53.7	15.1	62.5	43.7	8.1	9.2	52.6	2.7
Qwen3-30B-A3B-Instruct	HTML	**45.0**	94.8	**68.0**	**72.2**	**81.1**	89.8	**44.7**	**25.0**	**82.2**	4.8
	LaTeX	25.6	91.3	60.9	45.5	51.2	85.4	28.3	9.4	66.9	2.2
	Markdown	23.4	83.5	56.0	57.6	75.5	44.4	28.9	9.2	63.4	1.4
Table-specialised Models											
TableGPT2-7B	HTML	**20.8**	**94.1**	68.4	31.0	49.0	78.7	**13.5**	**12.9**	66.8	3.2
	LaTeX	18.1	89.7	**69.0**	16.7	38.8	**98.7**	6.2	7.2	57.4	**4.1**
	Markdown	4.9	67.2	58.0	**39.0**	**66.1**	38.0	6.7	9.2	49.0	1.0
TAMA-QWen3	HTML	7.2	48.5	4.9	6.8	15.1	44.2	6.4	9.4	**86.0**	0.6
	LaTeX	12.1	35.1	15.9	0.0	0.0	51.0	2.1	2.1	59.3	0.2
	Markdown	4.6	38.0	6.0	0.0	3.2	54.2	1.7	3.5	83.6	0.3

Table 12: SUC results for the LLM pipeline. Exact-match accuracy (%) across subtasks is reported.

	HTML image			Markdown image			LaTeX image		
Models	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**	**HTML**	**Md**	**TeX**
Open Models									
SmolVLM2-2.2B-IT	0.97	0.90	0.35	0.75	0.81	0.37	0.92	0.92	0.60
Gemma3-12B-IT	**1.00**	0.99	0.81	**1.00**	**1.00**	0.81	**1.00**	0.99	0.80
InternVL3.5-14B	**1.00**	**1.00**	**0.91**	**1.00**	0.99	**0.89**	**1.00**	0.99	0.91
InternVL3.5-30B-A3B	**1.00**	**1.00**	0.89	**1.00**	**1.00**	0.88	**1.00**	**1.00**	0.89
Qwen3-VL-8B-IT	**1.00**	**1.00**	0.88	**1.00**	**1.00**	0.87	**1.00**	**1.00**	**0.95**
Ministral3-14B-Instruct	**1.00**	0.97	0.84	**1.00**	0.97	0.84	**1.00**	0.98	0.88
LLaVA1.6-Vicuna-7B	0.98	0.76	0.00	0.99	0.68	0.00	0.99	0.73	0.00

Table 13: **Additional output usability results for Structure Reconstruction (SR).** Values report the fraction of syntactically usable reconstructed outputs across source render formats and target output formats. Best scores per column are shown in bold.

Model	Table Image Render						Table Text Format					
	HTML		LaTeX		Markdown		HTML		LaTeX		Markdown	
	EM	Rel.	EM	Rel.	EM	Rel.	EM	Rel.	EM	Rel.	EM	Rel.
Language Models (text-only)												
Qwen2.5-IT	–	–	–	–	–	–	44.57	50.29	42.71	48.43	45.43	51.43
Qwen3-IT	–	–	–	–	–	–	51.14	61.43	48.43	60.86	46.57	62.57
TableGPT2	–	–	–	–	–	–	44.43	57.29	41.57	56.00	42.14	58.57
TAMA-QWen3	–	–	–	–	–	–	18.29	52.00	19.14	53.71	20.71	52.71
Vision-Language Models												
SmolVLM2-IT	29.71	31.57	28.71	30.71	25.86	30.00	21.57	39.80	17.63	34.39	15.75	33.67
Gemma-3-IT 12B	38.86	41.86	39.57	43.29	38.57	43.00	50.29	55.57	49.00	54.14	48.57	53.71
Gemma-3-IT 27B	46.14	50.29	45.29	48.57	45.43	49.43	53.43	59.14	51.29	56.57	53.14	58.86
InternVL3.5 14B	48.57	52.57	48.14	52.43	48.00	52.57	47.14	55.86	47.29	54.29	44.86	55.14
InternVL3.5 30B-A3B	47.86	54.29	50.00	55.43	48.29	54.71	45.86	57.57	45.71	56.14	47.00	61.00
Qwen3-VL-IT 8B	50.29	55.43	49.29	54.71	49.71	55.14	53.43	59.14	52.14	57.43	53.29	59.14
Qwen3-VL-IT 30B-A3B	41.14	53.29	42.14	54.29	41.43	53.29	45.29	62.86	43.57	60.86	39.71	63.00
Ministral-3-IT	44.43	55.29	39.14	49.29	42.71	53.00	40.00	50.86	35.43	46.86	36.57	46.57
LLaVA-1.6 7B	31.86	35.71	31.43	35.29	32.00	35.71	27.37	43.91	29.55	41.05	26.50	42.17
LLaVA-1.6 13B	25.14	27.57	23.71	26.43	25.00	27.43	23.91	42.56	22.13	43.67	24.89	44.07
Table-specialised Vision-Language Models												
TableLLaVA-v1.5	1.29	20.86	1.00	19.71	4.00	23.71	23.61	36.24	27.37	37.41	28.40	38.80
Proprietary Models												
GPT-5.2	54.57	61.71	54.52	61.12	56.14	63.29	57.43	66.00	57.29	65.29	58.00	66.57
Gemini-3-Flash-Preview	65.43	72.00	65.14	71.16	65.43	71.57	65.71	72.29	65.00	71.86	65.43	72.86

Table 14: **TaskQA strict and relaxed matching diagnostic.** Exact-match accuracy (EM) and relaxed accuracy (Rel.) are reported across HTML, LaTeX, and Markdown inputs. Relaxed accuracy counts a prediction as correct when the normalized gold answer appears as a complete answer span inside a longer response. Underlined values indicate the highest score within each selected model-variant group and column. This diagnostic does not replace EM; it highlights cases where models retrieve the correct answer but fail the answer-only format required by strict exact match.

Models	Formats	T.P.	F.C.	L.C.	S.D.	# Rows	# Cols	C.Lu.	R.Lu.	Co.Rt.	Ro.Rt.	Overall
Open Models												
SmolVLM2-2.2B-IT	HTML	0.2	48.6	31.6	2.1	26.9	8.9	1.1	6.2	35.0	0.0	16.1
	LaTeX	0.0	49.4	34.5	0.2	18.9	8.4	1.1	7.0	30.4	0.0	15.0
	Markdown	0.0	19.6	17.0	0.5	2.7	0.6	0.5	6.2	5.6	0.0	5.3
Gemma-3-12B-IT	HTML	10.3	37.5	45.8	13.7	30.0	35.0	10.3	14.5	56.3	2.9	25.6
	LaTeX	13.8	41.2	48.2	13.5	30.2	37.2	11.0	15.1	62.8	3.0	27.6
	Markdown	10.7	40.1	46.7	18.1	32.8	42.0	14.1	15.4	66.8	2.4	28.9
InternVL3.5-30B-A3B	HTML	0.0	88.9	79.0	8.7	39.9	88.6	17.8	9.2	53.7	2.2	38.8
	LaTeX	0.0	89.2	80.6	12.6	44.5	83.5	21.5	15.9	69.8	4.0	42.1
	Markdown	0.0	84.9	76.5	25.1	56.1	85.5	24.5	13.0	64.9	4.5	43.5
Qwen3-VL-8B-IT	HTML	23.8	93.3	80.9	8.1	25.6	89.3	27.0	26.6	90.1	1.0	46.6
	LaTeX	20.8	93.2	78.9	14.3	36.1	88.7	32.4	30.0	89.5	3.0	48.7
	Markdown	24.6	87.8	78.4	14.6	31.8	90.1	28.0	34.2	89.8	2.2	48.2
LLaVA-1.6-7B	HTML	0.0	14.3	14.5	3.2	18.9	30.2	1.7	1.3	29.6	0.0	11.4
	LaTeX	0.3	13.8	18.1	1.3	20.5	23.5	1.9	2.9	28.3	0.0	11.1
	Markdown	0.2	14.1	18.1	1.9	17.6	23.8	1.6	3.3	29.9	0.0	11.1
Table-specialised Models												
TableLLaVA-v1.5-7B	HTML	0.0	6.5	4.3	0.0	0.0	0.3	0.0	1.6	1.0	0.0	1.4
	LaTeX	0.0	2.2	4.3	0.0	2.1	2.9	0.0	1.7	3.7	0.0	1.7
	Markdown	0.0	6.7	9.2	0.0	13.0	14.5	0.0	2.7	9.2	0.0	5.5

Table 15: **Additional SUC results for the VLM-Image pipeline.** Exact-match accuracy (%) across ten structure-oriented subtasks. Models receive rendered table images derived from HTML, LaTeX, and Markdown sources.

Model	Format	EM	Field Acc.	Relaxed Acc.	Δ_{Field}	Δ_{Relaxed}
VLM						
InternVL3.5-14B	HTML	47.8	57.5	58.3	+9.7	+10.5
	LaTeX	46.5	57.1	57.8	+10.6	+11.3
	Markdown	48.8	59.2	59.8	+10.4	+11.0
InternVL3.5-30B-A3B	HTML	38.8	52.8	53.7	+14.0	+14.9
	LaTeX	42.1	55.4	56.4	+13.3	+14.3
	Markdown	43.5	56.8	57.8	+13.3	+14.3
VLM-TEXT						
InternVL3.5-14B	HTML	53.9	66.0	67.5	+12.1	+13.6
	LaTeX	48.9	61.2	62.4	+12.3	+13.5
	Markdown	47.9	60.8	62.5	+12.9	+14.6
InternVL3.5-30B-A3B	HTML	46.9	64.2	72.0	+17.3	+25.1
	LaTeX	42.8	57.3	60.4	+14.5	+17.6
	Markdown	42.1	57.5	65.8	+15.4	+23.7

Table 16: **Diagnostic comparison of InternVL3.5-14B and InternVL3.5-30B-A3B on SUC.** Values report overall exact-match accuracy (EM), Field Accuracy, and Relaxed Accuracy across formats for the VLM-Image and VLM-Text pipelines. $\Delta_{\text{Field}} = \text{Field Accuracy} - \text{EM}$ and $\Delta_{\text{Relaxed}} = \text{Relaxed Accuracy} - \text{EM}$.

Model	Format	T.P.	F.C.	L.C.	S.D.	# Rows	# Cols	C.Lu.	R.Lu.	Co.Rt.	Ro.Rt.	Overall
Open Models												
InternVL3.5-14B	HTML	8.4	95.5	78.7	8.4	50.6	98.4	50.9	44.2	95.2	8.6	53.9
	LaTeX	11.1	94.1	76.0	11.0	44.2	99.4	36.9	22.3	87.1	6.8	48.9
	Markdown	1.4	76.3	71.4	14.5	65.7	85.9	43.6	33.2	83.9	3.3	47.9
InternVL3.5-30B-A3B	HTML	0.0	91.6	75.5	0.0	11.6	99.5	42.9	31.0	87.0	30.2	46.9
	LaTeX	0.0	85.1	75.8	0.0	9.1	99.4	28.8	21.0	86.8	21.9	42.8
	Markdown	0.0	64.4	75.4	0.0	15.7	97.1	32.4	28.5	86.2	21.5	42.1
Qwen3-VL-30B-A3B-IT	HTML	49.9	94.6	76.8	68.5	73.4	96.7	40.5	37.5	89.7	14.3	64.2
	LaTeX	29.4	95.5	69.6	46.9	64.5	97.5	29.7	19.7	83.0	8.6	54.5
	Markdown	26.4	87.8	64.9	62.3	84.6	71.7	29.6	20.8	72.2	5.9	52.6
Qwen3-VL-8B-IT	HTML	26.9	93.5	80.0	27.8	69.0	98.3	45.3	47.9	84.9	3.2	57.7
	LaTeX	18.6	92.1	76.3	15.7	36.2	**100.0**	30.2	17.5	74.1	5.6	46.6
	Markdown	10.7	68.5	64.5	30.2	66.9	92.8	28.1	18.8	69.0	8.3	45.8
Gemma-3-12B-IT	HTML	18.9	68.5	81.4	64.9	77.9	90.1	31.3	28.9	85.1	5.6	55.3
	LaTeX	14.5	53.1	76.5	30.5	35.6	98.7	19.1	18.4	76.9	3.7	42.7
	Markdown	4.3	31.6	67.4	51.4	73.3	67.2	19.4	20.7	76.3	2.5	41.4
Gemma-3-27B-IT	HTML	53.7	82.5	82.2	72.2	73.8	98.1	45.0	55.6	94.9	16.2	67.4
	LaTeX	33.2	62.6	76.2	60.4	60.1	99.5	42.0	32.3	89.0	13.2	56.9
	Markdown	16.4	45.3	70.6	59.5	83.3	76.5	40.5	38.5	91.6	11.6	53.4
Ministral-3-14B-IT	HTML	20.0	95.5	57.2	22.4	51.2	99.0	41.2	37.7	88.4	22.7	53.5
	LaTeX	14.1	79.0	62.3	30.2	40.5	99.4	19.2	17.0	68.5	12.2	44.3
	Markdown	1.7	69.0	49.4	6.5	44.7	80.0	23.8	21.3	62.0	11.1	37.0
SmolVLM2-2.2B-IT	HTML	0.0	11.3	9.2	0.0	6.8	0.2	0.2	2.6	2.9	0.0	3.3
	LaTeX	0.0	0.5	0.2	0.0	0.0	0.0	0.0	2.7	2.6	0.0	0.6
	Markdown	0.0	8.3	5.9	0.0	17.9	1.8	0.0	4.5	4.2	0.0	4.3
LLaVA-1.6-13B*	HTML	0.0	21.5	5.2	1.0	12.3	31.2	2.5	0.7	16.0	0.2	9.1
	LaTeX	0.0	24.7	7.6	0.3	1.0	1.6	2.3	2.4	19.4	0.2	5.9
	Markdown	0.0	12.5	4.9	0.0	8.1	5.5	1.6	1.3	12.5	0.0	4.6
LLaVA-1.6-7B*	HTML	0.0	27.5	2.7	0.0	0.0	0.0	0.0	1.0	14.4	0.0	4.6
	LaTeX	0.0	38.1	2.7	0.0	10.8	8.4	1.0	0.8	13.6	0.0	7.5
	Markdown	0.0	16.1	2.9	0.0	1.8	0.3	0.2	0.6	3.9	0.0	2.6
Table-specialised Models												
TableLLaVA-v1.5-7B	HTML	0.0	19.7	2.8	0.0	7.0	0.5	0.2	1.7	7.3	0.0	3.9
	LaTeX	0.0	33.8	5.7	0.0	23.9	9.5	0.5	1.5	7.8	0.0	8.3
	Markdown	0.0	21.3	3.9	0.0	24.5	3.1	0.0	1.0	5.4	0.0	5.9
Proprietary Models												
GPT-5.2	HTML	**94.9**	**99.4**	**95.5**	**98.4**	97.3	**100.0**	**53.3**	**86.0**	94.8	**84.4**	**90.4**
	LaTeX	90.5	98.4	92.5	29.3	84.4	**100.0**	7.0	17.2	**97.6**	3.5	62.0
	Markdown	88.6	94.9	93.3	87.3	96.0	99.8	14.8	34.8	96.2	25.1	73.1
Gemini-3-Flash-Preview	HTML	92.7	97.3	90.0	86.5	**97.8**	**100.0**	16.9	79.0	97.3	59.9	81.7
	LaTeX	90.6	97.1	89.3	8.9	89.8	**100.0**	0.6	19.4	97.5	5.9	59.9
	Markdown	88.9	95.7	89.0	75.8	95.1	**100.0**	15.9	67.1	97.5	49.8	77.5

Table 17: SUC results for the VLM-Text pipeline. Exact-match accuracy (%) across ten structure-oriented subtasks. Models receive table text extracted from HTML, LaTeX, and Markdown sources.

Models	HTML image						Markdown image						LaTeX image					
	Topology-Zero			Content-Zero			Topology-Zero			Content-Zero			Topology-Zero			Content-Zero		
	HTML	Md	TeX	HTML	Md	TeX	HTML	Md	TeX	HTML	Md	TeX	HTML	Md	TeX	HTML	Md	TeX
Open Models																		
SmolVLM2-2.2B-IT	0.87	0.85	0.31	0.74	0.73	0.27	0.66	0.76	0.30	0.57	0.66	0.26	0.79	0.83	0.57	0.66	0.68	0.47
Gemma3-12B-IT	0.94	0.95	0.75	0.79	0.79	0.63	0.95	0.96	0.75	0.80	0.81	0.63	0.94	0.94	0.74	0.78	0.77	0.62
Gemma3-27B-IT	0.97	0.97	0.84	0.86	0.84	0.75	0.98	0.97	0.84	0.85	0.85	0.74	0.97	0.96	0.83	0.83	0.82	0.73
InternVL3.5-14B	0.99	0.99	0.89	0.95	0.94	0.86	0.99	0.98	0.87	0.93	0.93	0.83	0.96	0.97	0.89	0.93	0.91	0.87
InternVL3.5-30B-A3B	0.98	0.99	0.88	0.95	0.94	0.85	0.99	0.99	0.87	0.93	0.94	0.84	0.96	0.97	0.87	0.93	0.92	0.84
Qwen3-VL-8B-IT	0.99	1.00	0.87	0.98	0.98	0.85	0.99	1.00	0.86	0.97	0.98	0.84	0.98	0.98	0.92	0.95	0.95	0.91
Qwen3-VL-30B-A3B-IT	0.98	0.99	0.69	0.98	0.98	0.67	0.99	1.00	0.88	0.97	0.98	0.85	0.97	0.98	0.93	0.95	0.95	0.92
Ministral3-14B-Instruct	0.98	0.94	0.83	0.94	0.90	0.81	0.98	0.94	0.83	0.92	0.90	0.80	0.95	0.92	0.86	0.87	0.86	0.82
LLaVA1.6-Vicuna-7B	0.70	0.64	0.00	0.43	0.39	0.00	0.70	0.52	0.00	0.46	0.35	0.00	0.71	0.57	0.00	0.44	0.36	0.00
LLaVA1.6-Vicuna-13B	0.66	0.80	0.00	0.46	0.51	0.00	0.62	0.74	0.00	0.45	0.50	0.00	0.65	0.74	0.01	0.46	0.49	0.01
Proprietary Models																		
GPT-5.2	0.98	0.98	0.78	0.97	0.94	0.75	0.98	0.99	0.91	0.96	0.97	0.88	0.98	0.97	0.91	0.94	0.94	0.88
Gemini-3-Flash-Preview	0.05	0.96	0.63	0.05	0.94	0.62	0.86	0.97	0.50	0.85	0.96	0.49	0.65	0.93	0.57	0.63	0.91	0.57
Table-specialised Models																		
TableLLaVA-v1.5-7B	0.73	0.69	0.55	0.33	0.32	0.26	0.73	0.71	0.56	0.43	0.42	0.33	0.58	0.56	0.49	0.29	0.29	0.26

Table 18: **Usability-aware Structure Reconstruction (SR) scores.** We report zero-penalized `GriTS`-Topology and `GriTS`-Content, where syntactically unusable outputs are assigned a score of zero before averaging. This captures both reconstruction fidelity and whether the requested output format is usable.