TABLEKIE: A BENCHMARK AND LEARNING FRAMEWORK TELETABLE FOR KEY INFORMATION EXTRACTION FROM TABLES

Anonymous authors
Paper under double-blind review

ABSTRACT

Key information extraction (KIE) from tables goes beyond recognizing textual content and associating values with entities. It requires synthesizing hierarchical row and column headers with contextual information into composite semantic keys that precisely characterize each value. Complex layouts and repeated field labels further challenge vision-language models in linking these keys to the correct values. We introduce TableKIE, a benchmark and learning framework for evaluating and improving table-centric KIE. The benchmark covers two complementary settings: closed-schema extraction, where target fields are predefined, and open-schema extraction, where models identify and organize key information explicitly present in a table without a fixed field inventory. Our primary evaluation focuses on field-value correctness and requested-field coverage. On the subset with verified structural annotations, we additionally evaluate row- and column-path agreement and value-box localization to assess the grounding behavior induced by auxiliary structural supervision. To improve extraction capabilities, we introduce a three-stage training framework that combines general KIE supervised fine-tuning, table-structure-aware fine-tuning, and reinforcement learning tailored to structured table extraction through Table-GRPO. Experiments show consistent gains across diverse table structures and reveal how schema design and training strategies affect extraction performance. TableKIE provides a unified testbed for studying reliable key information extraction from visually rich tables.

1 INTRODUCTION

Key information extraction (KIE) converts document content into structured key–value records, supporting applications such as scientific analysis and financial reporting. In complex tables, however, the key associated with a value is often not stated in a single text span. Instead, its meaning is jointly

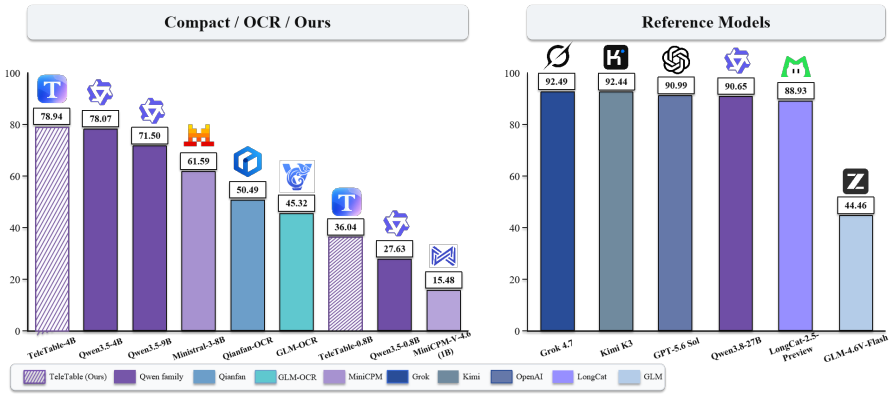


Figure 1: **Overall results on TableKIEBench.** Performance of representative models under the Open and Close evaluation settings.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

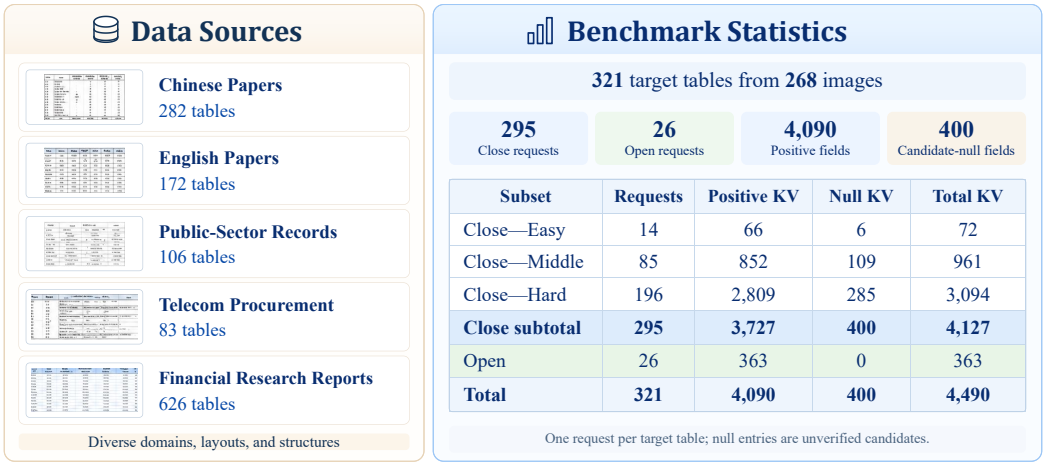


Figure 2: Overview of TableKIEBench.

defined by row groups, entity identifiers, and multiple levels of column headers. For example, a financial value such as "12.6" becomes a meaningful record only after its structural context identifies the business segment, revenue measure, and fiscal year.

Table KIE therefore requires aligning table structure with field semantics. In *open extraction*, models must recover the relevant row and column conditions and compose them into complete semantic keys. In *closed-key extraction*, models receive these keys and must resolve their conditions in the table to locate the corresponding values. Both settings require distinguishing fields whose labels or values may recur under different contexts. In particular, a correct answer string alone cannot establish correct localization when several cells contain the same value.

Existing research addresses important parts of this problem. Document KIE benchmarks such as DocILE and UNIKIE-BENCH evaluate extraction across diverse document scenarios (Šimsa et al., 2023; Ji et al., 2026). HiTab and TableVista examine hierarchical or visually complex tables through question answering and reasoning (Cheng et al., 2022; Yang et al., 2026), while schema-driven extraction and Statements organize table content into semantically meaningful records (Bai et al., 2024; Mishra et al., 2024). Building on these foundations, we focus on a specific evaluation need: connecting natural-language keys to the hierarchical row–column contexts and source cells that define them, under both open and closed-key extraction from table images.

We introduce **TableKIEBench**, a benchmark for evaluating the correspondence between field semantics and table structure. It covers diverse tables and links each key–value pair to its hierarchical context and source-cell location. The benchmark supports two complementary settings: *Open* evaluates the discovery and description of fields, while *Close* evaluates extraction conditioned on supplied keys. Together, these settings assess semantic field recovery and value association, with auxiliary annotations enabling analysis of structural and spatial alignment. An overview of the benchmark results is shown in Figure 1.

We further develop **TeleTable-0.8B** and **TeleTable-4B**, two table KIE models trained with table-specific supervised fine-tuning followed by **TableGRPO**. TableGRPO leverages structural supervision by rewarding value correctness together with hierarchical structure paths and source-cell bounding boxes, encouraging predictions that are both semantically correct and structurally grounded. Experiments further show that TeleTable retains strong generalization ability on general document KIE benchmarks. Detailed related work is provided in Appendix A.

2 TABLEKIEBENCH

We introduce **TableKIEBench** in Figurer 2 , a benchmark for key information extraction from visual tables in which field semantics are jointly determined by hierarchical row and column context. It

supports two complementary settings: discovering table-supported fields and extracting values for explicitly specified fields.

2.1 TASK SETTINGS

Both settings take a document image I and a target-table region b^{tab} as input. A semantic key describes a field through its relevant row and column conditions. Positive answers may be a string or a list of strings when multiple values satisfy the same conditions; the current release contains only string-valued references.

Open extraction (Open). The model discovers the fields in the target table and generates their semantic keys and values:

$$(I, b^{\text{tab}}) \rightarrow \hat{Y} = \big((\hat{k}_i, \hat{v}_i)\big)_{i=1}^{\hat{N}}, \tag{1}$$

where $\hat{k}_i$ and $\hat{v}_i$ denote the predicted semantic key and value, respectively, and $\hat{N}$ is the number of predicted records. This setting evaluates both field discovery and key–value association.

Closed-key extraction (Close). The model additionally receives m natural-language keys $K = (k_j)_{j=1}^{m}$ and predicts their corresponding values:

$$(I, b^{\text{tab}}, K) \rightarrow \hat{V} = (\hat{v}_j)_{j=1}^{m}, \tag{2}$$

where $\hat{v}_j$ is the prediction associated with key k_j. If a requested field is not supported by the target table, the model should return `null`. This setting evaluates conditioned value extraction and rejection of unsupported requests.

2.2 DATA SOURCES AND COMPOSITION

TableKIEBench is built from a diverse collection of financial, public-sector, and research documents in both Chinese and English, covering a wide range of domain vocabularies, table layouts, and hierarchical structures. Each positive field links a semantic key to its value, hierarchical row–column paths, and source-cell location; the latter two are used only for auxiliary alignment analysis. The benchmark contains **321 tables from 268 images**, including **295 Close** and **26 Open** requests, with **4,090 positive fields** and **400 candidate-null fields**. Close spans Easy, Middle, and Hard tables with 10%, 50%, and 100% field-coverage settings, while Open retains only tables with complete positive annotations.

Evaluation metrics. For Close requests, we report field-level accuracy across Easy, Middle, and Hard subsets, together with separate accuracies for positive fields and candidate-null fields. Positive-field accuracy measures exact value extraction, whereas candidate-null accuracy measures correct abstention on fields annotated as absent. Close F1 additionally penalizes unmatched and extra predictions. For Open requests, semantic KV-F1 measures whether models identify the correct fields and extract their values: a predicted pair receives credit only when its key is semantically equivalent to a reference key and its value matches after normalization. Key equivalence is assessed by a fixed LLM judge without access to values, followed by one-to-one matching. Overall semantic KV-F1 pools the true-positive, false-positive, and false-negative counts from Close and Open. All metrics are computed at the field level; the complete protocol is given in Appendix B.

3 TELETABLE

TeleTable addresses closed-key table KIE through table-specific supervised fine-tuning followed by TableGRPO. Given an image I, a target-table region b^{tab}, and m requested keys $K = (k_j)_{j=1}^{m}$, the model predicts their answers $\hat{V} = (\hat{v}_j)_{j=1}^{m}$. For example, the request in Figure 3 combines the age condition "≥ 60" with the column hierarchy "Female $\rightarrow$ HBcTA seroprevalence $\rightarrow$ %" to identify "33.00." Because the same value may appear in different cells, TableGRPO additionally evaluates whether each answer's structural and spatial evidence matches its annotated source.

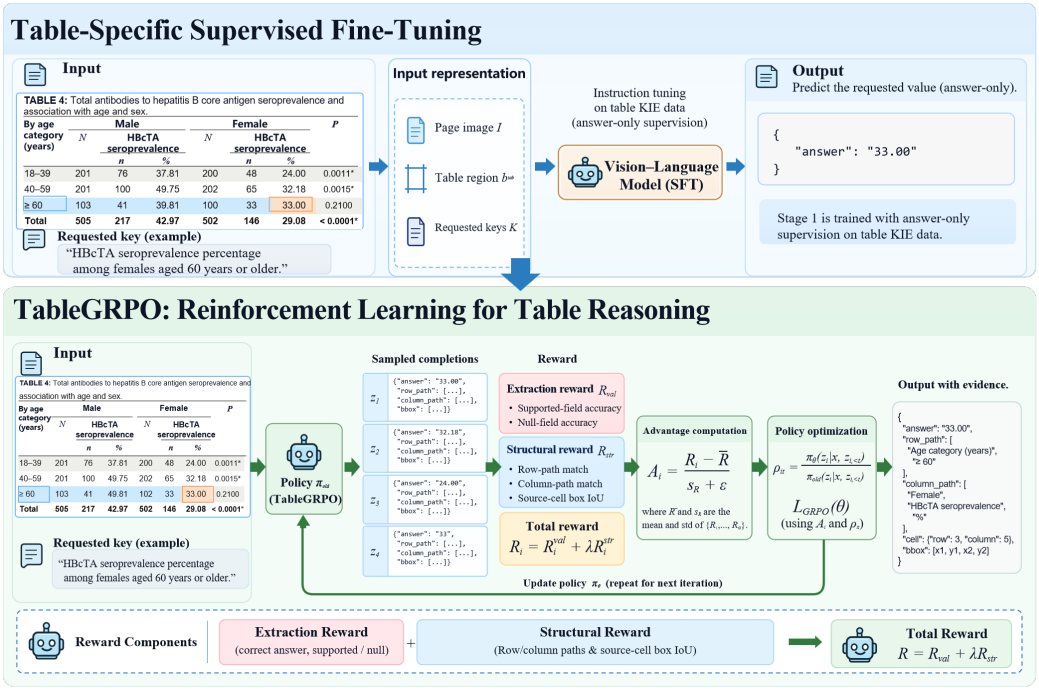


Figure 3: **Overview of TeleTable.** TeleTable first learns table-aware representations through supervised fine-tuning and is further optimized with TableGRPO, which jointly rewards value correctness, hierarchical structure paths, and source-cell localization.

3.1 Stage I: Table-Specific Supervised Fine-Tuning

We fine-tune a vision–language model on examples $(x, V^\star) \sim \mathcal{D}_{\mathrm{SFT}}$, where $x = (I, b^{\mathrm{tab}}, K)$ and $V^\star$ contains the corresponding gold answers. Training minimizes autoregressive negative log-likelihood over response tokens. This stage teaches the mapping from composite field descriptions to table values and the required KV output format. SFT targets contain only answers, without explicit path or box supervision. The objective is given in Appendix C.1.

3.2 Stage II: TableGRPO

Starting from the SFT policy, TableGRPO combines an extraction reward with an instance-level structural reward. The model generates answers together with auxiliary evidence: an ordered row path, an ordered column path, and a full source-cell box for each extracted value. Paths follow the outer-to-inner header hierarchy. Gold evidence is used only for scoring. The output specification and coordinate conventions appear in Appendix C.2.

Extraction reward. Let $q_j = \mathbf{1}[\hat{v}_j \equiv v_j^\star]$ indicate normalized exact agreement with the gold answer. List-valued answers are compared as multisets. Let $\mathcal{J}_+$ and $\mathcal{J}_0$ index fields with non-null and null gold answers, and let P_+ and P_0 denote their respective mean correctness. We define

$$R_{\mathrm{val}} = \begin{cases} 0.8P_+ + 0.2P_0, & \mathcal{J}_+ \neq \varnothing,\ \mathcal{J}_0 \neq \varnothing, \\ P_+, & \mathcal{J}_0 = \varnothing, \\ P_0, & \mathcal{J}_+ = \varnothing. \end{cases} \tag{3}$$

The weighting is applied within each input, emphasizing positive-field extraction while retaining a signal for null prediction.

Joint structural and spatial scoring. For a correctly answered positive field, we compare each predicted value and its evidence with the same gold instance. For predicted occurrence a and gold

occurrence b of field j, the pair score is

$$M_{ab}^{(j)} = \mathbf{1}[\hat{v}_{ja} \equiv v_{jb}^{\star}]\,\mathrm{IoU}(\hat{b}_{ja}, b_{jb}^{\star}) \cdot \left[\frac{1}{2} + \frac{1}{4}s_{\mathrm{path}}(\hat{r}_{ja}, r_{jb}^{\star}) + \frac{1}{4}s_{\mathrm{path}}(\hat{c}_{ja}, c_{jb}^{\star})\right], \tag{4}$$

where r, c, and b denote row paths, column paths, and cell boxes. The path score measures ordered header agreement using normalized longest common subsequence (Appendix C.3). Value agreement restricts matching to compatible content, and spatial overlap ensures that correct path labels cannot compensate for a disjoint box. The base term preserves partial localization credit when path recovery is incomplete.

For supported fields with $q_j = 1$ and matching numbers of predicted values, evidence entries, and gold instances, g_j is the mean pair score under the optimal one-to-one assignment. Otherwise, $g_j = 0$. This assignment permits different list orders without reusing gold instances. The structural reward is $R_{\mathrm{str}} = m^{-1}\sum_{j=1}^{m} g_j$, including zero contributions from null-valued fields. Assignment and validity rules are detailed in Appendix C.3.

Combined reward and optimization. The total reward is

$$R(x, \hat{Z}) = \begin{cases} R_{\mathrm{val}} + \lambda R_{\mathrm{str}}, & \text{if the response is valid,} \\ 0, & \text{otherwise,} \end{cases} \tag{5}$$

where $\hat{Z}$ contains the predicted answers and evidence. Validity requires a parseable response with exactly one answer of an allowed type per requested field. Evidence is not required for this check, so correct answers retain their extraction reward when evidence is missing or inaccurate.

We optimize this reward using GRPO (Shao et al., 2024), which compares sampled completions for the same input through group-relative advantages. The policy objective and implementation details appear in Appendix C.4.

4 Experiments

Our experiments examine three questions: (1) how existing models perform on field discovery and key-conditioned extraction from complex tables; (2) how supervised fine-tuning and TableGRPO affect extraction accuracy and evidence alignment; and (3) which reward components contribute to these changes. We additionally compare performance on general document KIE benchmarks.

4.1 Experimental Setup

Benchmark construction. We construct TableKIEBench from structured source annotations covering financial reports, public-sector documents, and scientific papers. Each record retains its source cell, row and column context, value, and spatial location. We use Qwen3.8-27B to refine hierarchical context and rewrite semantic keys while preserving the target values and boxes. Deterministic checks filter invalid outputs, incompatible evidence, and ambiguous records. For Open evaluation, all fields in each selected table are retained as reference fields. The final benchmark contains 321 requests from 268 images, covering 4,090 positive fields and 400 null candidates. Construction details appear in Appendix D.

Training data and partitioning. All data used to train TeleTable are fully disjoint from the TableKIEBench evaluation set. The retained training pool contains 19,151 fields. As summarized in Table 1, the SFT and RL subsets may share a portion of the source data, while using different training formats and supervision signals. The validation subset is image-disjoint from both SFT and RL.

Training records are converted into Close requests containing one or two fields. The RL subset emphasizes source-labeled Hard examples, which account for 96.40% of its fields, compared with 66.96% for SFT. Sample selection does not use model evaluation scores. All supported answers in the current training export are strings.

Table 1: Training and validation data. Requests contain one or two fields. Image-file hashes do not overlap across the three splits.

Split	Images	Tables	Positive	Null	Fields	Requests
SFT	436	648	11,557	1,831	13,388	6,864
RL	89	141	3,308	527	3,835	1,956
Validation	61	90	1,668	260	1,928	988
Total	586	879	16,533	2,618	19,151	9,808

Table 2: Main results on TableKIEBench. All scores are percentages; higher is better. Close accuracy is reported by difficulty and field type; candidate-null accuracy is additionally broken down by difficulty. Open semantic KV-F1 requires semantically equivalent keys and exactly matched values after normalization. Overall semantic KV-F1 pools Close and Open matching counts.

Model	**Close**									**Open**	**Overall**
	Accuracy			By field type		Null accuracy			F1	Sem. F1	Sem. F1
	Easy	Middle	Hard	Pos.	Null	Easy	Middle	Hard			
Reference models											
GPT-5.6 Sol	91.67	91.05	93.57	93.13	91.25	100.00	85.32	93.33	92.95	66.87	90.99
Qwen3.8-27B	90.28	85.85	92.31	92.17	77.75	33.33	67.89	82.46	90.77	89.35	90.65
GLM-4.6V-Flash	27.78	45.16	46.67	47.33	33.50	50.00	22.94	37.19	45.99	26.94	44.46
Kimi K3	95.83	93.33	93.37	93.23	95.08	100.00	93.42	95.42	93.41	89.99	92.44
Grok 4.7	91.67	100.00	93.41	92.50	94.60	94.60	92.70	96.14	92.70	90.14	92.49
LongCat-2.5-Preview	91.67	87.83	90.14	90.26	83.75	100.00	70.64	88.42	89.63	80.89	88.93
Compact baselines (reported sizes: 0.8B–9B)											
Qwen3.5-0.8B	52.78	40.17	23.50	30.88	0.00	0.00	0.00	0.00	29.00	3.15	27.63
MiniCPM-V-4.6 (1B)	31.94	25.08	13.19	17.95	0.75	33.33	0.00	0.35	16.28	0.00	15.48
Qwen3.5-4B	80.56	81.58	83.65	85.08	64.75	66.67	66.97	63.86	83.55	10.78	78.07
Ministral-3-8B	47.22	65.97	57.60	61.74	37.25	66.67	46.79	32.98	64.75	26.25	61.59
Qwen3.5-9B	84.72	79.92	75.57	82.24	25.50	50.00	24.77	25.26	76.87	17.18	71.50
OCR baselines											
Qianfan-OCR	83.33	61.08	47.35	55.76	8.50	83.33	12.84	5.26	52.20	29.56	50.49
GLM-OCR	63.89	54.84	44.54	52.35	0.00	0.00	0.00	0.00	47.57	0.00	45.32
TeleTable (ours)											
TeleTable-0.8B	68.06	34.34	32.87	36.17	12.00	0.00	23.85	7.72	38.25	0.00	36.04
TeleTable-4B	90.28	84.70	82.00	83.50	76.00	100.00	83.49	72.63	83.70	25.17	78.94

Models and implementation. We evaluate general-purpose vision–language models, including Qwen3.5 (0.8B, 4B, and 9B) (Qwen Team, 2026a), Qwen3.8-27B (Qwen Team, 2026b), Ministral-3-8B (Liu et al., 2026), GLM-4.6V-Flash (Z.ai, 2025), and MiniCPM-V-4.6 (OpenBMB, 2026); document-oriented models Qianfan-OCR (Dong et al., 2026) and GLM-OCR (Duan et al., 2026); and Kimi K3 (Kimi Team et al., 2026), GPT-5.6 Sol (OpenAI, 2026), Grok 4.7 (SpaceXAI, 2026), and LongCat-2.5-Preview (Meituan, 2026). TeleTable uses Qwen3.5-0.8B and Qwen3.5-4B as its backbones. SFT targets contain field identifiers and their values, while RL additionally requires row paths, column paths, and value-cell bounding boxes; the corresponding gold evidence is used to compute rewards. Within each backbone, controlled RL comparisons share the same SFT checkpoint, training data, output requirements, and development-set checkpoint selection protocol unless explicitly varied. We follow the protocol in Section 2.2, reporting Close positive-field and candidate-null accuracy, Open strict and semantic KV-F1, and overall KV micro-F1. Evidence alignment is evaluated separately and does not contribute to the benchmark's primary scores.

4.2 MAIN RESULTS ON TABLEKIEBENCH

Table 2 summarizes the results on TableKIEBench. We center our controlled comparisons on compact models of comparable scale, while larger reference models and OCR-specialized systems provide broader context.

A clear capacity bottleneck emerges at the ∼1B scale. The smallest models frequently fail on settings that require structured output, semantic field composition, or explicit rejection. Qwen3.5-0.8B and MiniCPM-V-4.6 obtain near-zero candidate-null and Open scores, indicating a pronounced extraction collapse beyond straightforward positive-field prediction. TeleTable-0.8B substantially improves Close F1 from 29.00 to 38.25 and Hard accuracy from 23.50 to 32.87, but still obtains 0.00 Open semantic F1 and only 12.00 candidate-null accuracy. Thus, additional supervision and RL can improve extraction at this scale, but cannot fully overcome the limited capacity for reliable format following, semantic composition, and abstention.

TeleTable is most effective once sufficient model capacity is available. At 4B, TeleTable improves Overall semantic F1 from 78.07 to 78.94 without increasing model size. More importantly, the gains concentrate on the more structure-sensitive dimensions: Open semantic F1 increases from 10.78 to 25.17, while candidate-null accuracy rises from 64.75 to 76.00. In contrast, positive-field accuracy changes from 85.08 to 83.50 and Close F1 remains nearly unchanged. This pattern suggests that TableGRPO does not simply increase surface-level extraction accuracy. Instead, it shifts the model toward better semantic association, structural grounding, and rejection of unsupported fields.

Large models provide a strong reference ceiling, but are not the primary comparison. Large reference models achieve roughly 89–92.5 Overall semantic F1, demonstrating substantial remaining headroom for compact models. However, these systems differ considerably in scale and deployment characteristics, and therefore do not constitute parameter-matched competitors to TeleTable. We consequently base our main conclusions on comparisons with similarly sized compact backbones. Notably, performance is not monotonic with parameter count: Qwen3.5-9B reaches 71.50 Overall semantic F1, below the 78.07 of Qwen3.5-4B. This further indicates that task-specific alignment and calibrated extraction are as important as model scale for complex table KIE.

OCR-specialized models reveal a distinct extraction bias. Qianfan-OCR performs competitively on Easy tables (83.33), but drops sharply on Hard tables (47.35) and achieves only 8.50 candidate-null accuracy. GLM-OCR similarly obtains zero candidate-null accuracy and zero Open semantic F1. These results are consistent with over-specialization toward extracting visible content: when the requested field is absent, the models tend to return a plausible page value rather than abstain. Their degradation on harder structures further suggests that strong OCR capability alone does not guarantee reliable field–value association under hierarchical context.

Taken together, the results expose three distinct regimes: small models are limited by task execution and structured generation, large models provide strong but costly reference performance, and compact models benefit most from explicit structure-aware training once sufficient capacity is available. These findings motivate TeleTable as a parameter-efficient alternative focused specifically on complex table KIE.

4.3 EFFECTIVENESS OF TELETABLE

Effectiveness of TableGRPO. Table 3 reveals a consistent pattern across both model scales. Applying GRPO with value reward alone does not improve upon SFT. Instead, Close accuracy decreases by 0.92 points for the 0.8B model and 1.91 points for the 4B model. This result suggests that policy optimization itself is insufficient: rewarding only the final value provides little guidance about *why* a prediction corresponds to the requested field and can even degrade the extraction behavior learned during SFT.

Introducing structural rewards reverses this trend. Relative to value-only GRPO, adding box supervision improves Close accuracy by 1.38 and 1.45 points for the 0.8B and 4B models, respectively, whereas adding hierarchical path supervision yields larger gains of 2.06 and 2.08 points. The re-

Table 3: **Reward ablations and policy-optimization variants on TableKIEBench.** Results for the 0.8B and 4B models are shown side by side. V, B, and P denote value, box, and path rewards, respectively. All scores are percentages; higher is better.

Training setting	Reward	TeleTable-0.8B			TeleTable-4B		
		Close Acc.	Pos. Acc.	Null Acc.	Close Acc.	Pos. Acc.	Null Acc.
SFT baseline and reward ablations							
SFT only	—	31.40	33.81	9.00	80.93	81.57	75.00
GRPO	V	30.48	32.87	8.25	79.02	79.85	71.25
GRPO	V + B	31.86	34.21	10.00	80.47	81.16	74.00
GRPO	V + P	32.54	34.88	10.75	81.10	81.94	73.25
Policy-optimization variants with full rewards							
GSPO	V + B + P	32.57	34.88	11.00	81.66	82.43	74.50
Dr. GRPO	V + B + P	33.15	35.50	11.25	82.19	82.94	75.25
TableGRPO (ours)	V + B + P	**33.83**	**36.17**	**12.00**	**82.77**	**83.50**	**76.00**

markably consistent ordering across model scales indicates that hierarchical row–column context provides a particularly informative signal for resolving field–value correspondence, while spatial localization supplies complementary evidence about the source of the extracted value.

Combining both structural signals produces the strongest results. Compared with SFT, TableGRPO improves Close accuracy by 2.43 points for TeleTable-0.8B and 1.84 points for TeleTable-4B, with simultaneous gains in both positive-field and candidate-null accuracy. The latter is important because it shows that structural supervision improves not only value extraction but also the model's ability to avoid unsupported associations. Overall, these results suggest that the benefit of TableGRPO comes from coupling answer correctness with explicit evidence about both *where* the value is located and *which hierarchical field* it belongs to.

Policy-optimization variants. With the full V+B+P reward, TableGRPO consistently outperforms GSPO and Dr. GRPO across all three metrics and both model scales. Compared with Dr. GRPO, Close accuracy further increases by 0.68 points for the 0.8B model and 0.58 points for the 4B model. The same ordering is observed for positive-field and candidate-null accuracy. This consistency indicates that the observed gains are not restricted to a particular field type or model capacity, and supports TableGRPO as an effective optimization strategy for exploiting structure-aware rewards in table KIE.

Qualitative analysis of structural rewards. Appendix E further examines representative cases with and without structural supervision. We compare predicted values together with their row–column paths and source-cell locations against the references. These examples illustrate how structural rewards encourage the extracted value to remain consistent with both its hierarchical field context and its supporting table cell, complementing the quantitative ablations above.

4.4 Generalization Beyond TableKIEBench

Evaluation setup. To assess the applicability of TeleTable beyond TableKIEBench, we evaluate both model variants on the Nanonets KIE evaluation suite, covering DocILE, Handwritten-Forms, and Nanonets-KIE. We report field-level normalized Levenshtein similarity, computed as one minus the edit distance between the predicted and reference values divided by the longer string length. Predictions are matched to ground-truth fields by their labels, and both values are converted to strings before scoring. Missing predictions are treated as empty strings, and each dataset score is averaged over all ground-truth fields. Additional predicted fields are not penalized by this metric. Besides the overall score returned by the evaluation script, we report the unweighted macro-average across the three datasets.

Table 4: **General document KIE results on the Nanonets suite.** Scores are normalized Levenshtein similarities scaled by 100; higher is better.

Model	DocILE	H-Forms	N-KIE	Overall
Qwen3.5-4B	55.89	67.11	88.26	75.19
Qwen3.5-9B	63.95	**70.40**	**88.83**	78.63
Qianfan-OCR	65.25	69.05	86.44	77.64
GLM-OCR	32.78	5.38	66.60	50.21
Ministral-3-8B	40.92	1.14	6.85	18.65
TeleTable-0.8B	79.47	58.61	82.01	79.47
TeleTable-4B	**82.46**	64.84	83.93	**82.07**

Table 5: **Training cost of TeleTable.** Time denotes measured training runtime, and GPU-hours are computed using the allocated number of GPUs.

Model	Stage	GPUs	Updates	Time (min)	GPU-h
0.8B	SFT	4	429	59.39	3.96
	RL	2	272	31.74	1.06
4B	SFT	4	429	75.18	5.01
	RL	4	340	38.49	2.57

Results and analysis. As shown in Table 4, TeleTable-4B achieves the highest overall score of 82.07, outperforming Qwen3.5-4B and Qwen3.5-9B by 6.88 and 3.44 points, respectively. TeleTable-0.8B reaches 79.47, exceeding all compared baselines under the same aggregation protocol despite its compact size. Importantly, TeleTable-4B also leads under equal dataset weighting, achieving a macro-average of 77.08 compared with 70.42 for Qwen3.5-4B and 74.39 for Qwen3.5-9B. Its overall advantage therefore remains when each dataset contributes equally to the evaluation.

The strongest gains occur on DocILE, where TeleTable-0.8B and TeleTable-4B achieve 79.47 and 82.46, respectively, exceeding the strongest baseline, Qianfan-OCR, by 14.22 and 17.21 points. However, the improvements are not uniform: TeleTable-4B trails Qwen3.5-4B by 2.27 points on Handwritten-Forms and 4.33 points on Nanonets-KIE. These results demonstrate strong aggregate performance beyond TableKIEBench while revealing dataset-dependent trade-offs, with particularly pronounced benefits on DocILE.

4.5 Computational Cost

Table 5 reports the training cost of our main experiments. Both models are trained for one SFT epoch over 6,864 requests with four GPUs. For RL, TeleTable-0.8B and TeleTable-4B use two and four GPUs, respectively. Overall, the full SFT and RL procedures require only 5.02 and 7.58 GPU-hours, demonstrating that the proposed training pipeline introduces modest computational overhead. During RL, we sample four completions per input and reuse each rollout for two optimization iterations, with a maximum completion length of 2,048 tokens. All experiments use BF16, gradient checkpointing, and LoRA for parameter-efficient training.

5 Conclusion

We introduced TableKIEBench and TeleTable for extracting information from visual tables whose field meanings are jointly defined by hierarchical row and column context. TableKIEBench supports open field discovery and closed-key extraction, evaluating semantic key generation and reliable field–value association. TeleTable combines table-specific supervised fine-tuning with Table-GRPO, whose instance-level reward couples value correctness with hierarchical and spatial evidence through one-to-one instance alignment. Our 4B experiments show that structural rewards distinguish completions with identical answer correctness and improve evidence alignment. Future work will extend evidence-aware training to open extraction and examine its effectiveness across model scales.

AI USE STATEMENT

AI tools were used to assist with language polishing.

REFERENCES

Fan Bai, Junmo Kang, Gabriel Stanovsky, Dayne Freitag, Mark Dredze, and Alan Ritter. Schema-driven information extraction from heterogeneous tables. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10252–10273. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.600. URL https://aclanthology.org/2024.findings-emnlp.600/.

Zhoujun Cheng, Haoyu Dong, Zhiruo Wang, Ran Jia, Jiaqi Guo, Yan Gao, Shi Han, Jian-Guang Lou, and Dongmei Zhang. 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)*, pp. 1094–1110. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.78. URL https://aclanthology.org/2022.acl-long.78/.

Daxiang Dong, Mingming Zheng, Dong Xu, Chunhua Luo, Bairong Zhuang, Yuxuan Li, Ruoyun He, Haoran Wang, Wenyu Zhang, Wenbo Wang, Yicheng Wang, Xue Xiong, Ayong Zheng, Xiaoying Zuo, Ziwei Ou, Jingnan Gu, Quanhao Guo, Jianmin Wu, Dawei Yin, and Dou Shen. Qianfan-OCR: A unified end-to-end model for document intelligence. *arXiv preprint arXiv:2603.13398*, 2026. doi: 10.48550/arXiv.2603.13398. URL https://arxiv.org/abs/2603.13398.

Shuaiqi Duan, Yadong Xue, Weihan Wang, Zhe Su, Huan Liu, Sheng Yang, Guobing Gan, Guo Wang, Zihan Wang, Shengdong Yan, Dexin Jin, Yuxuan Zhang, Guohong Wen, Yanfeng Wang, Yutao Zhang, Xiaohan Zhang, Wenyi Hong, Yukuo Cen, Da Yin, Bin Chen, Wenmeng Yu, Xiaotao Gu, and Jie Tang. GLM-OCR technical report. *arXiv preprint arXiv:2603.10910*, 2026. doi: 10.48550/arXiv.2603.10910. URL https://arxiv.org/abs/2603.10910.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. LayoutLMv3: Pre-training for document AI with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 4083–4091. Association for Computing Machinery, 2022. doi: 10.1145/3503161.3548112. URL https://doi.org/10.1145/3503161.3548112.

Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. FUNSD: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, pp. 1–6. IEEE, 2019. doi: 10.1109/ICDARW.2019.10029. URL https://doi.org/10.1109/ICDARW.2019.10029.

Yifan Ji, Zhipeng Xu, Zhenghao Liu, Zulong Chen, Qian Zhang, Zhibo Yang, Junyang Lin, Yu Gu, Ge Yu, and Maosong Sun. UNIKIE-BENCH: Benchmarking large multimodal models for key information extraction in visual documents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6331–6352. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.287. URL https://aclanthology.org/2026.acl-long.287/.

Xiaoqiang Kang, Shengen Wu, Zimu Wang, Yilin Liu, Xiaobo Jin, Kaizhu Huang, Wei Wang, Yutao Yue, Xiaowei Huang, and Qiufeng Wang. Can GRPO boost complex multimodal table understanding? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 12631–12644. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.637. URL https://aclanthology.org/2025.emnlp-main.637/.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. OCR-free document understanding transformer. In *Computer Vision – ECCV 2022*, pp. 498–517. Springer, 2022. doi: 10.1007/978-3-031-19815-1_29. URL https://doi.org/10.1007/978-3-031-19815-1_29.

Kimi Team et al. Kimi K3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026. doi: 10.48550/arXiv.2607.24653. URL https://arxiv.org/abs/2607.24653.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026. doi: 10.48550/arXiv.2601.08584. URL https://arxiv.org/abs/2601.08584.

Meituan. LongCat API docs: Quick start. Official documentation for LongCat-2.5-Preview, 2026. URL https://longcat.ai/platform/docs/. Accessed September 26, 2026.

Lokesh Mishra, Sohayl Dhibi, Yusik Kim, Cesar Berrospi Ramis, Shubham Gupta, Michele Dolfi, and Peter Staar. Statements: Universal information extraction from tables with large language models for ESG KPIs. In *Proceedings of the 1st Workshop on Natural Language Processing Meets Climate Change*, pp. 193–214. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.climatenlp-1.15. URL https://aclanthology.org/2024.climatenlp-1.15/.

OpenAI. GPT-5.6 system card, July 2026. URL https://deploymentsafety.openai.com/gpt-5-6.

OpenBMB. MiniCPM-V-4.6. Hugging Face model card, 2026. URL https://huggingface.co/openbmb/MiniCPM-V-4.6. Accessed September 26, 2026.

Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. CORD: A consolidated receipt dataset for post-OCR parsing. *Workshop on Document Intelligence at NeurIPS 2019*, 2019. URL https://openreview.net/forum?id=SJl3z659UH.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026b. URL https://qwen.ai/blog?id=qwen3.8.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Štěpán Šimsa, Milan Šulc, Michal Uřičář, Yash Patel, Ahmed Hamdi, Matěj Kocián, Matyáš Skalický, Jiří Matas, Antoine Doucet, Mickaël Coustaty, and Dimosthenis Karatzas. DocILE benchmark for document information localization and extraction. In *Document Analysis and Recognition – ICDAR 2023, Part II*, pp. 147–166. Springer, 2023. doi: 10.1007/978-3-031-41679-8_9. URL https://doi.org/10.1007/978-3-031-41679-8_9.

Brandon Smock, Rohith Pesala, and Robin Abraham. PubTables-1M: Towards comprehensive table extraction from unstructured documents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4634–4642. IEEE, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Smock_PubTables-1M_Towards_Comprehensive_Table_Extraction_From_Unstructured_Documents_CVPR_2022_paper.html.

SpaceXAI. Model card: Grok 4.7, September 2026. URL https://media.x.ai/v1/website/card4p7-3a96f40b.pdf. Revision September 21, 2026.

Zhiruo Wang, Haoyu Dong, Ran Jia, Jia Li, Zhiyi Fu, Shi Han, and Dongmei Zhang. TUTA: Tree-based transformers for generally structured table pre-training. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1780–1790. Association for Computing Machinery, 2021. doi: 10.1145/3447548.3467434. URL https://doi.org/10.1145/3447548.3467434.

Zhenhe Wu, Jian Yang, Zhongjiang He, Changzai Pan, Jiaheng Liu, Xianjie Wu, Yu Zhao, Shuangyong Song, Yongxiang Li, Zhoujun Li, and Xuelong Li. Table-R1: Region-based reinforcement learning for table understanding. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 27374–27391. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.1364. URL https://aclanthology.org/2026.findings-acl.1364/.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. LayoutLM: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1192–1200. Association for Computing Machinery, 2020. doi: 10.1145/3394486.3403172. URL https://doi.org/10.1145/3394486.3403172.

Zheyuan Yang, Lyuhao Chen, Arman Cohan, and Yilun Zhao. Table-R1: Inference-time scaling for table reasoning tasks. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20605–20624. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1040. URL https://aclanthology.org/2025.emnlp-main.1040/.

Zheyuan Yang, Liqiang Shang, Junjie Chen, Xun Yang, Chenglong Xu, Bo Yuan, Chenyuan Jiao, Yaoru Sun, and Yilun Zhao. TableVista: Benchmarking multimodal table reasoning under visual and structural complexity. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34967–34985. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.1745. URL https://aclanthology.org/2026.findings-acl.1745/.

Z.ai. GLM-4.6V-Flash. Hugging Face model card, 2025. URL https://huggingface.co/zai-org/GLM-4.6V-Flash. Accessed September 26, 2026.

Mingyu Zheng, Xinwei Feng, Qingyi Si, Qiaoqiao She, Zheng Lin, Wenbin Jiang, and Weiping Wang. Multimodal table understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9102–9124. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.493. URL https://aclanthology.org/2024.acl-long.493/.

A Detailed Related Work

A.1 Document Information Extraction

Document KIE aims to recover structured information from documents with diverse layouts and visual characteristics. LayoutLM and LayoutLMv3 integrate textual, spatial, and visual information into document representations (Xu et al., 2020; Huang et al., 2022), while Donut formulates document understanding through direct image-to-sequence generation (Kim et al., 2022). More recent document models, including Qianfan-OCR and GLM-OCR, support information extraction alongside recognition and document parsing (Dong et al., 2026; Duan et al., 2026). These approaches provide foundations for extracting structured outputs directly from document images.

Benchmarks cover complementary document types and extraction requirements. FUNSD and CORD study forms and receipts (Jaume et al., 2019; Park et al., 2019). DocILE evaluates information localization and extraction, including the grouping of fields into line items in business documents (Šimsa et al., 2023). UNIKIE-BENCH extends evaluation to multimodal models through constrained-category and open-category extraction (Ji et al., 2026). These settings already address spatial grounding, structured records, and variation in requested information.

TableKIEBench concentrates on how a field is defined within a complex table. Its distinguishing focus is the composition of field semantics from multiple row and column levels, together with annotations linking the resulting key to its value instance. This focus complements general document KIE evaluation by making hierarchical field composition explicit in both field discovery and key-conditioned extraction.

A.2 Table Structure and Multimodal Understanding

Table understanding research has developed representations and benchmarks for recovering and reasoning over table structure. TUTA models spatial and hierarchical relationships using tree-based representations of the two table dimensions (Wang et al., 2021). PubTables-1M provides annotations for table detection, structure recognition, and functional analysis (Smock et al., 2022). These contributions support the identification of cells, headers, and their structural relationships.

Hierarchical and multimodal benchmarks further examine how these relationships support downstream tasks. HiTab studies question answering and generation over hierarchical tables (Cheng et al., 2022). MMTab and Table-LLaVA extend table understanding to image inputs across diverse instructions and tasks (Zheng et al., 2024). TableVista evaluates multimodal reasoning under variations in visual presentation and structural complexity (Yang et al., 2026).

These studies establish the importance of table structure and visual perception. TableKIEBench examines a complementary output requirement: recovering explicit semantic key–value records whose field conditions may be distributed across several structural levels. A recovered grid identifies cell organization, whereas a semantic key must express the conditions that make a particular value meaningful. Our annotations connect these representations through row and column paths and source-cell locations.

A.3 Semantic Extraction from Tables

Semantic table extraction is closely related to our task. Schema-driven information extraction uses human-authored schemas to specify the records to be recovered from heterogeneous tables (Bai et al., 2024). It evaluates extraction across multiple domains and demonstrates how language models can map varying table organizations into a common output representation. Statements represents quantitative facts together with their contextual information and formulates extraction as translating tables into this representation (Mishra et al., 2024). Its applications to ESG reports illustrate the value of preserving the semantic context surrounding numerical entries.

These approaches share our emphasis on extracting meaningful facts rather than isolated cell strings. TableKIEBench brings together three requirements: extraction directly from table images, natural-language keys composed from hierarchical row and column conditions, and instance-level references for the structural and spatial evidence supporting each value. Open and Close probe complementary directions of this field–structure relation. Open requires generating a field description from the table, whereas Close requires resolving a supplied description to the supported value instances. This correspondence does not assume a unique wording for each field, motivating semantic key matching in Open evaluation.

A.4 Reinforcement Learning for Table Understanding

Several distinct studies use the name Table-R1. We distinguish them by their training objectives and supervision signals.

Yang et al. (2025) investigate inference-time scaling for table reasoning through two post-training strategies: distillation of reasoning traces and reinforcement learning with verifiable rewards. Their RL approach uses GRPO (Shao et al., 2024) and task-specific rewards across table question answering and fact verification. This work establishes a basis for applying verifiable reward optimization to table reasoning.

Kang et al. (2025) address multimodal table understanding with a three-stage framework. Following warm-up, Perception Alignment GRPO uses Tree-Edit-Distance Similarity (TEDS) to reward table reconstruction, and Hint-Completion GRPO trains reasoning conditioned on partial solution hints. This approach explicitly supervises table perception and structure, making it closely relevant to learning from visual table evidence.

Region-based Table-R1 (Wu et al., 2026) incorporates relevant table regions into the reasoning process. Region-Enhanced SFT teaches region identification before answer generation, and Table-Aware Group Relative Policy Optimization (TARPO) combines region accuracy with answer cor-

rectness through a changing reward balance and consistency penalties. It therefore provides explicit supervision for relevant row–column selection and its relationship to the answer.

TeleTable builds on the use of structural supervision but targets closed-key information extraction. Its reward evaluates each extracted value together with the hierarchical paths and full source-cell box attached to that occurrence. Structural credit is conditional on complete field-answer correctness; spatial overlap modulates path agreement, and one-to-one matching prevents multiple predictions from claiming the same gold instance. These components jointly assess whether the predicted value and its evidence refer to the same annotated source. We use GRPO for policy optimization and focus our task-specific design on this instance-level reward.

B Evaluation Protocol

B.1 Evaluation Units and Value Matching

A request contains one or more reference fields. Scoring is performed at the field level: an error in one field does not remove credit from other correctly predicted fields in the same parseable request.

For string values, we apply Unicode NFKC normalization, collapse consecutive whitespace into a single space, and remove leading and trailing whitespace. Let $\nu(\cdot)$ denote this normalization. Value correctness is defined as

$$\mathrm{EqVal}(a,b) = \begin{cases} 1, & a = b = \texttt{null}, \\ 1, & a, b \text{ are strings and } \nu(a) = \nu(b), \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

We do not apply numerical tolerance, case folding, unit conversion, percentage conversion, or decimal-precision adjustment to values. For example, `1.0` and `1`, `1,000` and `1000`, and `10%` and `0.1` remain different. The strings `"null"`, `""`, `"--"`, and `"N/A"` are not treated as JSON `null`.

The current benchmark represents positive answers as strings and negative candidates as JSON `null`. The scorer does not support list-valued answers or multiset comparison, and the released evaluation annotations contain no list-valued reference answers.

B.2 Close Evaluation

Close predictions are aligned with reference fields using exact request-local `field_id` values. A reference field receives credit only when its ID occurs exactly once in the prediction and the associated value satisfies EqVal.

Let $\mathcal{G}^+$ and $\mathcal{G}^0$ denote the sets of positive and candidate-null reference fields, respectively. Writing $c(g) \in \{0,1\}$ for the correctness of field g, we compute

$$\mathrm{Acc}_+ = \frac{\sum_{g\in\mathcal{G}^+} c(g)}{|\mathcal{G}^+|}, \tag{7}$$

$$\mathrm{Acc}_{\mathrm{null}} = \frac{\sum_{g\in\mathcal{G}^0} c(g)}{|\mathcal{G}^0|}. \tag{8}$$

Missing, incorrect, malformed, or duplicate-ID answers receive no accuracy credit, while their reference fields remain in the denominator. A missing answer is not interpreted as a predicted null.

For a request with a parseable field array, let N_P be the number of predicted entries and N_G the number of reference fields. The Close counts are

$$TP_C = \sum_{g\in\mathcal{G}} c(g), \qquad FP_C = N_P - TP_C, \qquad FN_C = N_G - TP_C. \tag{9}$$

Correct candidate-null predictions contribute to TP_C. An incorrect value associated with an existing reference field contributes both a false positive and a false negative. Extra predicted entries contribute false positives.

B.3 Open Evaluation

Open evaluation requires both identifying a field and extracting its value. We therefore distinguish key equivalence from value correctness.

Semantic key equivalence. We use a fixed GPT-5.6 Sol judge with low reasoning effort. The judge receives reference keys, their available aliases, predicted keys, and request-local table IDs. It receives neither extracted values nor reference values, model identities, or images.

The judge accepts paraphrases, unambiguous abbreviations, translations, and changes in word order when they identify the same field. Shared table-title wording may be omitted if the field remains unambiguous. Distinguishing information such as the row entity, measure, year, unit, or hierarchical qualifier must be preserved. Matching only the row or only the column is insufficient. The judge returns key-equivalence decisions rather than corrected keys, corrected values, or holistic answer scores.

One-to-one KV matching. We construct a bipartite graph between predicted and reference fields within each request. An edge (p, g) is allowed only when

$$\mathrm{Edge}(p,g) = \mathbf{1}\left[\begin{array}{l} \mathrm{table}(p) = \mathrm{table}(g), \\ \mathrm{KeyEq}(p,g) = 1, \\ \mathrm{EqVal}(v_p, v_g) = 1 \end{array}\right], \tag{10}$$

where KeyEq accepts either deterministic key/alias equality or a semantic-equivalence decision from the judge. Deterministic key matching applies NFKC normalization, removes whitespace, and case-folds the key.

We compute a maximum-cardinality one-to-one matching M using an augmenting-path algorithm. The Open counts are

$$TP_O = |M|, \qquad FP_O = N_P - |M|, \qquad FN_O = N_G - |M|. \tag{11}$$

Each prediction and reference field can participate in at most one matched pair. Predictions with the same table ID and normalized key are excluded from matching as duplicates, but remain in the predicted-entry count. Distinct paraphrases of the same reference field cannot receive repeated credit because matching is one-to-one.

Open precision, recall, and KV-F1 are computed from the pooled counts:

$$P_O = \frac{TP_O}{TP_O + FP_O}, \qquad R_O = \frac{TP_O}{TP_O + FN_O}, \qquad F_{1,O} = \frac{2TP_O}{2TP_O + FP_O + FN_O}. \tag{12}$$

Judge responses and field-level matching records are retained for inspection. Semantic scores rely on a single automated judge and are not human-adjudicated equivalence labels.

B.4 Overall and Grouped Metrics

We aggregate counts over all evaluated requests:

$$TP = \sum_{r \in \mathcal{R}_C} TP_r + \sum_{r \in \mathcal{R}_O} TP_r, \tag{13}$$

$$FP = \sum_{r \in \mathcal{R}_C} FP_r + \sum_{r \in \mathcal{R}_O} FP_r, \tag{14}$$

$$FN = \sum_{r \in \mathcal{R}_C} FN_r + \sum_{r \in \mathcal{R}_O} FN_r. \tag{15}$$

The overall score is

$$\text{Overall KV-F1} = \frac{2TP}{2TP + FP + FN}. \tag{16}$$

This is a pooled micro-F1, not an arithmetic average of Close and Open F1, nor a fixed task-weighted average. We report the task-specific metrics alongside the overall score because the two tasks contain different numbers of fields.

Difficulty and coverage breakdowns use the same scoring rules on the corresponding subsets. For a Close difficulty group $\mathcal{G}_d$, the reported field accuracy is

$$\mathrm{Acc}_d = \frac{\sum_{g \in \mathcal{G}_d} c(g)}{|\mathcal{G}_d|}, \tag{17}$$

including both positive and candidate-null fields. The 10%, 50%, and 100% coverage anchors define request groups; they do not modify matching rules or field weights. Actual coverage is recorded separately after integer field selection.

Full-benchmark scores require complete request coverage and completed Open semantic scoring. Scores computed from incomplete outputs are labeled as partial results rather than full-benchmark results. Assumed correctness for unavailable outputs is not part of the empirical evaluation protocol.

B.5 Output Parsing and Invalid Predictions

Before scoring, deterministic format normalization recovers supported output representations without consulting reference answers. Examples include removing a complete outer Markdown code fence and converting explicit ID–value maps into field arrays. JSON numeric value tokens may be converted to strings while preserving their original lexical form. Normalization does not infer missing answers, correct values using the references, or select among conflicting duplicates.

For parseable field arrays, scoring remains fieldwise:

- A missing reference field contributes a false negative.
- An unmatched, extra, or invalid predicted entry contributes a false positive.
- Duplicate Close IDs or duplicate Open table–key bindings receive no matching credit; unrelated fields remain scoreable.
- An incorrect prediction for a reference field contributes both a false positive and a false negative.

If normalization cannot recover a valid field array, the entire request receives

$$TP = 0, \qquad FP = N_G, \qquad FN = N_G. \tag{18}$$

Duplicate property names within a JSON object are rejected rather than resolved by retaining one occurrence. Whole-request format validity is recorded separately from field-level correctness.

B.6 Candidate-Null Annotations

The evaluation set contains 400 null annotations marked as `unverified_null_candidate`, with `human_verified=false`. We therefore report candidate-null accuracy separately from positive-field accuracy and do not describe these annotations as human-verified negatives. Candidate-null fields are nevertheless included in Close and overall KV-F1 under the stated reference-label convention.

C TeleTable Objectives and Scoring Details

C.1 Supervised Fine-Tuning Objective

For each training example $(x, V^\star) \sim \mathcal{D}_{\mathrm{SFT}}$, let $s^\star = \mathrm{serialize}(V^\star)$ be the target response. The supervised objective is

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{(x,V^\star)\sim\mathcal{D}_{\mathrm{SFT}}} \left[\sum_{t=1}^{|s^\star|} \log \pi_\theta(s_t^\star \mid x, s_{<t}^\star) \right]. \tag{19}$$

Only response tokens contribute to the loss. The targets contain KV answers and do not explicitly supervise row paths, column paths, or cell boxes.

C.2 Evidence Representation

The expanded model output is

$$\hat{Z} = \big((\hat{v}_j, \hat{E}_j)\big)_{j=1}^{m}, \qquad \hat{E}_j = \big((\hat{r}_{ja}, \hat{c}_{ja}, \hat{b}_{ja})\big)_{a=1}^{\ell_j}, \tag{20}$$

where ℓ_j is the number of predicted evidence entries. The a-th entry is associated with the a-th extracted value and contains its row path, column path, and source-cell box. For `null`, the evidence list is empty.

Paths contain header texts in outer-to-inner order. Each box covers the full source cell, including internal whitespace and the entire merged region where applicable. Coordinates use $[x_{\min}, y_{\min}, x_{\max}, y_{\max}]$ normalized to $[0, 1000]$ by the width and height of the supplied image. The image may be a document page or a table crop.

The model generates each answer before its evidence. Gold evidence is supplied to the reward function rather than substituted for model predictions. Primary KV evaluation uses the predicted answers without requiring auxiliary evidence.

C.3 Matching and Structural Scoring

Answer matching. Field correctness is $q_j = \mathbf{1}[\hat{v}_j \equiv v_j^\star]$, where $\equiv$ denotes normalized exact matching. Lists are compared as multisets, so order is ignored but repeated occurrences are preserved. For each nonempty field set,

$$P_+ = \frac{1}{|\mathcal{J}_+|} \sum_{j \in \mathcal{J}_+} q_j, \qquad P_0 = \frac{1}{|\mathcal{J}_0|} \sum_{j \in \mathcal{J}_0} q_j. \tag{21}$$

Path agreement. For predicted and gold paths p and $p^\star$,

$$s_{\text{path}}(p, p^\star) = \frac{2|\,\text{LCS}(p, p^\star)|}{|p| + |p^\star|}, \tag{22}$$

where $|p|$ counts header elements and LCS is the longest common subsequence of normalized header texts. This score preserves relative order while allowing partial agreement. Two empty paths score one; a missing or malformed predicted path scores zero.

Instance assignment. Let $n_j^\star$ be the number of gold value occurrences. For a supported field with $q_j = 1$ and equal numbers of predicted values, evidence entries, and gold occurrences, we compute

$$g_j = \frac{1}{n_j^\star} \max_{\sigma \in \mathfrak{S}_{n_j^\star}} \sum_{a=1}^{n_j^\star} M_{a,\sigma(a)}^{(j)}, \qquad R_{\text{str}} = \frac{1}{m} \sum_{j=1}^{m} g_j, \tag{23}$$

where $M_{ab}^{(j)}$ is defined in Equation 4. The Hungarian algorithm finds the optimal one-to-one assignment. Each predicted value remains attached to its own evidence throughout matching.

The formulation supports multiple values. In the current training export, supported answers are strings, so the assignment reduces to a single predicted–gold instance pair.

Invalid and missing evidence. We set $g_j = 0$ for null or incorrect answers, mismatched instance counts, or evidence that is missing, non-list, or contains non-object entries. Within an otherwise admissible evidence list, an invalid box receives zero IoU, while an invalid path receives zero path similarity. Thus, an invalid component does not automatically invalidate the entire response.

In Equation 4, the path-dependent factor ranges from 0.5 to 1. A pair receives zero credit when its values differ or its boxes do not overlap. Partial path agreement modulates credit only when the predicted box overlaps the corresponding gold cell. Null-valued fields receive no structural credit but remain in the denominator m.

Table 6: Source-pool composition. Easy, Middle, Hard, and Open are inherited source groups. Difficulty labels are not newly assigned using a structural-complexity classifier.

Source group	Positive fields	Null candidates	Total
Easy	540	60	600
Middle	5,412	601	6,013
Hard	19,722	2,194	21,916
Open	1,470	163	1,633
Total	27,144	3,018	30,162

C.4 Group-Relative Policy Optimization

We initialize from the SFT policy. For each input x, the sampling policy π_{old} generates $G = 4$ completions. For completion i with reward R_i and tokens $(z_{it})_t$, the group-relative advantage and token-level importance ratio are

$$A_i = \frac{R_i - \bar{R}}{s_R + \epsilon}, \qquad \rho_{it} = \frac{\pi_\theta(z_{it} \mid x, z_{i,<t})}{\pi_{\text{old}}(z_{it} \mid x, z_{i,<t})}, \tag{24}$$

where $\bar{R}$ and s_R are the mean and sample standard deviation of rewards for the same input, and $\epsilon = 10^{-4}$.

For a batch $\mathcal{B}$ of completions, the loss is

$$\mathcal{L}_{\text{TableGRPO}} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \frac{\sum_t m_{it} \min(\rho_{it} A_i,\ \text{clip}(\rho_{it}, 1-\eta, 1+\eta) A_i)}{\max(1, \sum_t m_{it})}, \tag{25}$$

where $\eta = 0.2$ and m_{it} indicates whether token t contributes to optimization. The objective averages over included tokens within each completion before averaging across completions.

Each rollout batch is reused for two iterations. Updates use LoRA in the language backbone without a KL penalty. Truncated completions contribute zero policy loss but remain in the group reward statistics.

D Data Construction and Training Preparation

D.1 Source Records

After applying existing annotation edits and deletion records, the source pool contains 30,162 field records: 27,144 positive fields and 3,018 null candidates. These records reference 906 target-table identifiers and 602 image paths, corresponding to 598 distinct image-file hashes. The difference between path and hash counts reflects file aliases and does not establish document-level deduplication.

The sources include Chinese financial reports, government and public-sector materials, and Chinese- and English-language scientific papers. Each record retains its table identifier, source-cell identity, row and column indices, hierarchical paths, value, cell box, image dimensions, and annotation status. This provenance allows refinement and filtering to operate on source-cell associations rather than key strings alone.

D.2 Request Sampling

Sampling targets an evaluation budget of approximately 4,000–5,000 fields. A fixed random seed and category-weighted table ordering favor Hard tables while retaining Easy and Middle examples. Selection uses source metadata and integer-feasible field counts, without consulting model evaluation scores or enforcing exact field quotas for each difficulty group.

For Close, each selected table is assigned one request scope, with 10%, 50%, and 100% serving as reference coverage anchors. Selected field counts are adapted to the available positive annotations, and realized coverage is recorded. Null candidates are sampled separately and excluded from positive-field coverage calculations. For Open, each selected table contributes all intended positive reference fields.

Table 7: Deterministic checks used after key refinement. These checks detect defined inconsistencies but do not guarantee semantic correctness.

Category	Checks
Output validity	JSON schema, target-ID coverage, duplicate identifiers, field types, and output truncation.
Evidence membership	Evidence identifiers must belong to the permitted candidate set; target answer cells cannot be reused as header evidence.
Spatial compatibility	Column evidence must lie above and cover the target column; row evidence must satisfy left-side or group-range conditions.
Key expression	Nonempty text, length bounds, and filtering of mechanical indices, generic descriptions, and path-concatenation patterns.
Consistency	Basic language checks, heuristic answer-leakage checks, and normalized key conflicts within a table.
Uncertainty	Explicit review flags and unresolved source-cell or multi-value associations.

The initial evaluation sample contains 3,603 positive fields and 400 null candidates across 225 requests. A supplemental candidate batch contains 2,005 positive fields and 220 null candidates. Records retained for final evaluation are excluded from the training exports at the field-record level.

D.3 Structure-Conditioned Key Refinement

We use Qwen3.8-27B to refine the structural context and semantic key associated with each fixed target cell. The model receives the source image, target-table location, full source-cell context, target-cell identity, and candidate row and column evidence. Its output specifies the applicable header hierarchy, the rewritten key, evidence-cell identifiers, and row-group ranges where required.

The refinement has two objectives. First, it identifies the row and column conditions governing the target cell, including relevant groups, periods, units, and experimental conditions. Second, it expresses these conditions as a coherent natural-language field description. Keys follow the language of the source table where feasible and avoid mechanical concatenation of header strings.

For example, the source key "This field represents the effect on A entry for equalize library" is refined to "the effect on A for the equalize library intervention with K=10." The corresponding row path is "equalize library $\rightarrow$ K=10," and the column path is "effect on A."

The source value, value-cell box, image, and record identifier are preserved. Only the key, paths, and associated structural metadata are revised. The annotation model can observe the target value through the image and cell context; this procedure is annotation refinement rather than answer-blind verification.

D.4 Consistency Filtering

The checks cover output validity, admissible evidence, spatial compatibility, and key consistency, as summarized in Table 7.

Records failing these checks are excluded from the active export while preserving their original content and rejection reasons. Fields with unresolved associations to multiple source cells are excluded rather than automatically merged into list-valued answers. Multiline text within a single cell remains one string.

For Open, field-level filtering is followed by a table-level coverage check. If an intended reference field is removed, the table is excluded from Open evaluation, including any of its fields that passed individual filtering. This preserves completeness relative to the intended source annotations.

D.5 Training Conversion and Coordinates

Training and validation subsets are grouped by image-file hash before conversion into requests. Each request contains one or two fields from its target table. Source records assigned to Open are also converted into closed-key training examples.

SFT target responses contain field identifiers and answers, without explicit path or box targets. RL prompts request the answer followed by its evidence, and reward annotations retain the row path, column path, and source-cell box. Conversion checks cover image readability, tokenizer compatibility, target-format round-trip parsing, and sequence-length budgets.

Boxes cover complete source cells, including internal whitespace and the full merged region where applicable. For an input image of width W and height H, a pixel box is converted as

$$b_{\text{norm}} = 1000 \left[\frac{x_{\min}}{W}, \frac{y_{\min}}{H}, \frac{x_{\max}}{W}, \frac{y_{\max}}{H} \right]. \tag{26}$$

The coordinate frame is the supplied image, which may be a full document page or a table crop. It is not defined relative to the target-table region unless that region is itself the supplied image.

The scoring formulation supports list-valued answers through one-to-one instance matching. Supported answers in the current training export are strings, so this assignment reduces to one predicted–gold instance pair. The current training experiments therefore do not isolate the contribution of multi-instance assignment.

D.6 Annotation Inference Configuration

Annotation inference uses four A100 40GB GPUs per machine with tensor parallelism of four and up to 16 concurrent requests. The context limit is 49,152 tokens and the maximum output length is 8,192 tokens. Temperature is set to zero, thinking mode is disabled, and outputs are constrained by a JSON schema. Each batch contains at most 12 target fields together with the complete table context.

A shared task manifest and file-based locking coordinate work across machines. Inputs, raw outputs, model configurations, source hashes, and rejection reasons are retained. Completed batches are persisted to support resumption. Failed or overlength tasks are explicitly excluded rather than accepting silently truncated outputs.

E Qualitative Analysis of Table-Specific SFT

We examine four selected Close examples from Chinese and English tables to illustrate successful extraction by the TeleTable-4B. Differences caused only by JSON wrappers or punctuation are not used to select these examples. Correct comparison-model outputs are also shown. These deliberately selected cases illustrate specific output differences, rather than establish an aggregate performance advantage or an effect of RL.

Row and column context. Figure 4 illustrates two forms of context selection. In the database table, the requested attribute must be combined with the two-level column path *Graph model* $\rightarrow$ *HyperGraphDB*. In the seroprevalence table, the age group and the nested *Male* $\rightarrow$ *HBcTA seroprevalence* $\rightarrow$ % header jointly identify the target. Some incorrect answers equal values in other rows, such as the male total or the older age group. This is an observable correspondence between outputs and table entries, not evidence of the model's internal reading or localization process.

Fine-grained experimental conditions. Figure 5 shows why closely related settings must remain distinct. The image-generation example distinguishes `VAR-d16` from `VAR-d20` under the same tokenizer; the plant-science example distinguishes a treatment from its control. The SFT answers preserve these conditions in the selected fields, while GLM-4.6V-Flash returns the other configuration's or control group's value. Other models also answer these fields correctly.

Visualization protocol. The panels use crops of the original input images and reference structure annotations. Predictions and reference values are unchanged. For legibility, the row and value overlays in Figure 4(a) are shifted downward by ten original-image pixels relative to the stored annotations; both original and display coordinates are retained in the accompanying metadata. All other displayed overlays use the stored coordinates directly. The figures do not provide new human-adjudicated ground truth or evaluate predicted evidence alignment.

(a) Hierarchical column context: database persistence

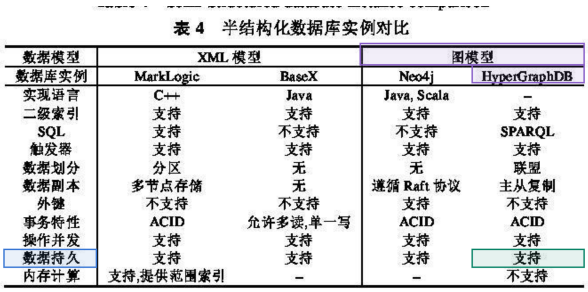


表 4 半结构化数据库实例对比

数据模型	XML 模型		图模型	
数据库实例	MarkLogic	BaseX	Neo4j	HyperGraphDB
实现语言	C++	Java	Java, Scala	–
二级索引	支持	支持	支持	支持
SQL	支持	不支持	不支持	SPARQL
触发器	支持	支持	支持	支持
数据划分	分区	无	无	联盟
数据副本	多节点存储	无	遵循 Raft 协议	主从复制
外键	不支持	不支持	支持	不支持
事务特性	ACID	允许多读,单一写	ACID	ACID
操作并发	支持	支持	支持	支持
数据持久	支持	支持	支持	支持
内存计算	支持,提供范围索引	–	–	不支持

Key (English gloss). Data persistence support of HyperGraphDB under the graph model.

Reference. Supported.
Row. Data persistence.
Columns. Graph model → HyperGraphDB.
Cell. (r_{11}, c_4).

Model	Answer
TeleTable-4B	**Supported**
Qwen3.5-4B	Not supported
Qwen3.5-9B	ACID
Qwen3.8-27B	Supported

(b) Age-specific statistics versus a population total

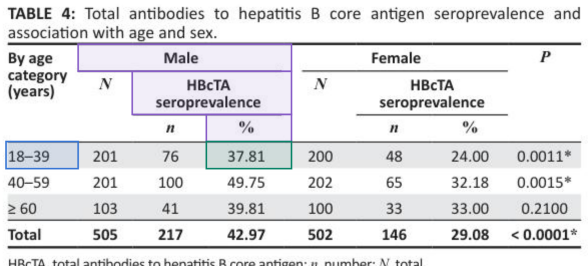


TABLE 4: Total antibodies to hepatitis B core antigen seroprevalence and association with age and sex.

By age category (years)	Male			Female			*P*
	N	HBcTA seroprevalence		*N*	HBcTA seroprevalence		
		n	%		*n*	%	
18–39	201	76	37.81	200	48	24.00	0.0011*
40–59	201	100	49.75	202	65	32.18	0.0015*
≥ 60	103	41	39.81	100	33	33.00	0.2100
Total	**505**	**217**	**42.97**	**502**	**146**	**29.08**	**< 0.0001***

HBcTA, total antibodies to hepatitis B core antigen; *n*, number; *N*, total

Key. HBcTA seroprevalence percentage for males in the 18–39 age category.

Reference. 37.81.
Row. 18–39.
Columns. Male → HBcTA seroprevalence → %.
Cell. (r_3, c_3).

Model	Answer
TeleTable-4B	**37.81**
Qwen3.5-4B	42.97
GLM-4.6V-Flash	49.75
Qwen3.8-27B	37.81

Figure 4: **Selected successes of the TeleTable-4B model in composing row and column context.** Blue, purple, and green boxes indicate reference row context, column context, and target values, respectively; they are not model-predicted evidence. Red answers differ substantively from the reference. In (a), query and answer text are translated from Chinese for readability. In (b), 42.97 is the male total, whereas 49.75 belongs to males aged 40–59. Cell indices are zero-based and include header rows.

(c) Same tokenizer, different generator configuration

Table 7: Class-conditional image generation results on ImageNet 256 × 256.

Type	Tokenizer	Generator	Param.	FID↓	IS↑	Pre.↑	Rec.↑
Diff.	SD-VAE [23]	DiT-L/2 [20]	458M	5.02	167.2	0.75	0.57
Diff.	SD-VAE [23]	DiT-XL/2 [20]	675M	2.27	278.2	0.83	0.57
Mask.	VQGAN [6]	MaskGIT [1]	227M	6.18	182.1	0.80	0.51
VAR	VAR [28]	VAR-d16 [28]	310M	3.30	274.4	0.84	0.51
VAR	VAR [28]	VAR-d20 [28]	600M	2.57	302.6	0.83	0.56
AR	LlamaGen [27]	LlamaGen-L [27]	343M	3.80	248.3	0.83	0.51
AR	LlamaGen [27]	LlamaGen-XL [27]	775M	3.39	227.1	0.81	0.54
AR	FVQ [2]	LlamaGen-L [27]	343M	2.39	276.6	0.84	0.56
AR	FVQ [2]	LlamaGen-XL [27]	775M	2.07	287.0	0.83	0.58
AR	IBQ [25]	IBQ-B [25]	342M	2.88	254.7	0.84	0.51
AR	IBQ [25]	IBQ-L [25]	649M	2.45	267.5	0.83	0.52
AR	StableVQ	IBQ-B [25]	342M	2.35	256.0	0.82	0.58
AR	StableVQ	IBQ-L [25]	649M	2.18	250.4	0.82	0.59

Key. IS score for the VAR-d16 [28] generator using the VAR [28] tokenizer in the class-conditional image generation results on ImageNet 256 × 256.

Reference. 274.4.
Row context. VAR; VAR [28]; VAR-d16 [28].
Column. IS↑.
Cell. (r_4, c_5).

Model	Answer
TeleTable-4B	**274.4**
GLM-4.6V-Flash	302.6
Qwen3.8-27B	274.4
GPT-5.6 Sol	274.4

(d) Treatment-specific measurement versus control

Table 3. Differences of leaf physiological indexes under 5 different biofertilizer treatments of spinach.

Treatments	Leaf physiological indexes							
	SP (mg/g FW)	SC (mg/g FW)	MDA (mg/gFW)	Chl a (mg/g FW)	Chl b (mg/g FW)	Chl a+b (mg/g FW)	Chl a/b (mg/g FW)	WP(Ψ, Mpa)
None, CK	1.83±0.26c	0.73±0.05d	5.49±0.51a	1.16±0.03e	0.36±0.03c	1.52±0.03e	3.25±0.30b	-6.57±0.25d
Bs, T1	3.31±0.20b	2.26±0.06b	1.80±0.13b	2.73±0.22b	0.58±0.06ab	3.30±0.15b	4.91±0.83a	-4.23±0.12b
Bs+Bm, T2	4.14±0.04a	3.59±0.13a	1.25±0.14b	3.23±0.04a	0.60±0.03a	3.83±0.07a	5.36±0.25a	-2.79±0.22a
Bm+Ba, T3	2.37±0.08c	1.59±0.11b	4.82±0.31a	1.70±0.07d	0.40±0.01c	2.11±0.07d	4.21±0.09ab	-5.53±0.26c
Bs+Ba, T4	3.28±0.22b	2.23±0.11c	2.09±0.15b	2.29±0.11c	0.47±0.02bc	2.75±0.09c	4.91±0.43a	-4.79±0.12b
F value	24.48**	118.35**	44.95**	50.63**	8.80**	118.34**	203.88**	48.72**

Key. The MDA (mg/g FW) value for the Bs, T1 treatment in the spinach leaf physiological indexes table.

Reference. 1.80 ± 0.13b.
Row. Bs, T1.
Columns. Leaf physiological indexes → MDA → (mg/gFW).
Cell. (r_4, c_3).

Model	Answer
TeleTable-4B	$\mathbf{1.80 \pm 0.13b}$
GLM-4.6V-Flash	5.49 ± 0.51a
Qwen3.8-27B	1.80 ± 0.13b
Qwen3.5-9B	1.80 ± 0.13b

Figure 5: **Selected successes under fine-grained experimental conditions.** In (c), the incorrect answer 302.6 belongs to VAR-d20, not the requested VAR-d16 configuration. The three row labels are parallel identifying attributes, not necessarily a semantic hierarchy. In (d), 5.49 ± 0.51a belongs to the None, CK control, not the requested Bs, T1 treatment. The final letters in (d) are part of the reference value and are retained in the displayed answers. Reference-box colors and cell-index conventions follow Figure 4.