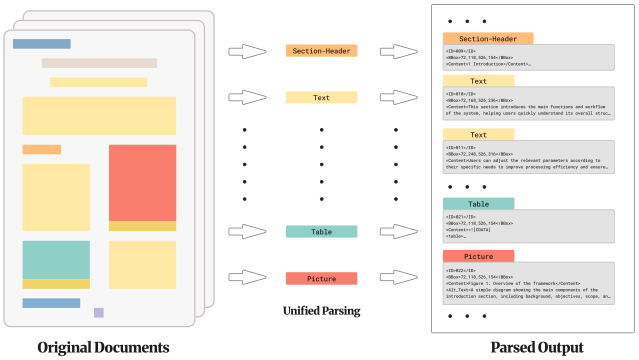
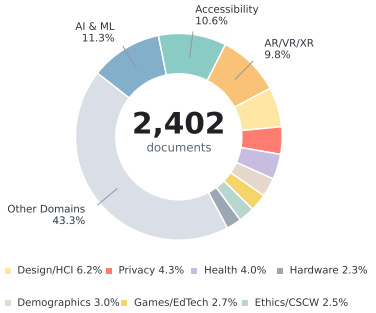
Read, Parse, Describe: Unified Document Parsing with Visual Element Description Generation

Yuheng Chen[1], Yufan Chen[1,*], Zhuojun Cai[1], Ruiping Liu[1], Junwei Zheng[1], Jiale Wei[1]
Di Wen[1], Weijia Fan[1,3], Kunyu Peng[1], Jiaming Zhang[2], Rainer Stiefelhagen[1]
[1]Karlsruhe Institute of Technology [2]Hunan University [3]University of Alberta
chenc6417@gmail.com, yufan.chen@kit.edu



(a) Unified document parsing with different document elements.



(b) Domain distribution.

Figure 1: **Read-Parse-Describe Dataset Overview**. Fig. 1a shows that the unified parsing contract converts documents into structured outputs with localized elements, reading order, textual content, and visual descriptions. Fig. 1b shows the domain distribution of the 2,402 source documents.

Abstract

Document understanding systems are typically evaluated as separate components via layout analysis, optical text recognition, reading-order prediction, and image captioning. This separation leaves scientific figures, captions, and visual semantics weakly connected, and fails to measure whether a model can produce a single page-level parse that is both structurally faithful and semantically accessible. We introduce Read-Parse-Describe, a unified dataset for document parsing with visual element description generation. Given a page image, models are required to output an ordered list of layout elements containing bounding boxes, semantic labels, reading-order indices, recognized textual content, and alternative text descriptions for visual elements. The benchmark is built from 2,402 documents, comprising 18,163 continuous pages, 125,218 layout elements, and 11,338 visual instances. Besides, we further propose SARA, a plug-and-play space-awareness enhancer that reuses multi-layer visual features and injects grid-aware spatial guidance into vision-language tokens for layout-aware generation. Experiments across end-to-end VLMs and pipeline parsers show that existing systems still struggle to jointly recover reading order and visual semantics, while our enhanced model achieves strong joint performance. These results highlight the need to move beyond isolated document parsing modules toward unified models that jointly read, parse, and describe complex documents. The new dataset and code will be made publicly available at RPD Project.

*Corresponding author: Yufan Chen (yufan.chen@kit.edu).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

Scientific documents are not just sequences of recognized words. A single page may contain two-column prose, section headers, equations, tables, figures, captions, footnotes, author metadata, and visual evidence whose meaning is not recoverable from optical character recognition (OCR) alone. Document AI has therefore progressed through layout analysis [1, 2, 3, 4], table recognition [5], document VQA [6], OCR-free parsing [7, 8], and multimodal document models [9]. In accessibility and machine-readable paper ingestion, the useful output is not an isolated layout map or a free-form page summary, but a structured page representation that can be read, reflowed, indexed, and audited.

Existing benchmarks usually evaluate pieces of this representation in isolation. Layout datasets measure region localization [10, 11, 12], OCR and table benchmarks [13] measure textual structure, reading-order datasets [14] focus on sequencing, captioning, and alt-text datasets [15, 16] describe isolated or context-conditioned images. These tasks are necessary, but they do not test whether a system can produce a single consistent parse in which boxes, labels, reading order, text, tables, captions, and visual descriptions agree with each other. This gap is particularly visible in scientific PDFs, where captions frequently refer to nearby figures or tables, equations interrupt prose, and visual elements carry technical information that should be exposed to downstream readers.

We introduce **Read-Parse-Describe**, a unified task and dataset for unified document parsing with visual element description generation as shown in Fig. 1. Given a page image, a model must output an ordered list of elements, which includes a bounding box, semantic category, reading-order index, recognized text content, an explicit caption link when applicable, and an alternative-text description for visual elements. The key novelty is not any single field, but enforcing their agreement in one auditable page parse. The unified output contract can be implemented by either end-to-end models or modular systems. The benchmark contains 2,402 open-access scientific documents, 18,163 continuous pages, 125,218 layout elements, and 11,338 visual instances.

Modern VLMs and document-specialized models [17, 18, 19, 20, 21, 22, 23, 24] increasingly use frozen or partially frozen visual encoders. When such models merge visual tokens before generation, local spatial evidence can be weakened exactly where element extents, reading order, and caption-object alignment are required. We therefore introduce **SARA**, a lightweight Spatial Alignment Residual Adapter for end-to-end VLM parsing. SARA freezes the base VLM, reuses intermediate visual encoder layers, projects their concatenated features into the token space, aligns with crop-aware spatial grids, and injects into the native merged tokens. The task makes spatial-alignment failures measurable, while SARA tests a direct architectural response without changing labels or evaluation. To summarize, our contributions are as follows:

- We introduce Read-Parse-Describe, a novel task and dataset for unified document parsing with alternative-text generation into a shared structured schema.
- We propose SARA, a Spatial Alignment Residual Adapter that injects crop-aware spatial cues into VLM tokens to improve layout-sensitive document understanding.
- We establish a unified evaluation protocol covering both document parsing and alternative-text generation, enabling systematic assessment of parsing fidelity and generation quality.
- We conduct extensive experiments and ablation studies, showing that crop-aware spatial alignment consistently benefits unified document parsing and alternative-text generation.

2 Related Work

2.1 Document Layout and Parsing Datasets

Document-layout benchmarks have made document understanding measurable by isolating the output. Early resources, such as RVL-CDIP [25], treat documents as classes, whereas FUNSD [26] focuses on form. CORD [27] and SROIE [28] further show that domain-specific schemas enhance extraction but are specialized. Similar to the table, SciTSR [29] targets scientific table structure, and PubTables-1M [13] scales table recognition, but neither requires the table, caption, reading position, and visual description to agree in one element record. PubLayNet [10] made large-scale layout detection practical, and DocBank [11] expanded token-level layout supervision. DocLayNet [12] improved further with richer human annotations, while M6Doc [30] expanded layout analysis across heterogeneous

documents. For robustness, RoDLA [4] stresses layout robustness, whereas ReadingBank [14] isolates reading-order prediction. Broader evaluations such as DocVQA [31] and DUDE [32] test document understanding through answers rather than page parses. OmniDocBench [33] moves closer to document parsing, while MMLongBench-Doc [34] stresses long-context document understanding.

2.2 Document VLMs and Pipeline Systems

Structure-aware representations are often precise, while generative systems are flexible but low-performing. LayoutLM [1] and LayoutLMv2 [2] made layout-aware pretraining central for document understanding. LayoutLMv3 [3] further unifies text and image modeling through task-specific heads rather than a complete generator. DocFormer [35] and VGT [36] emphasize structure-aware encoding, reinforcing the explicit geometry. OCR-free generation such as Donut [6] can emit structured text directly from an image, while Pix2Struct [37] supports screenshot-style pretraining. Nougat [38] specializes this generative view for academic documents, and GOT-OCR [7] extends OCR-style generation toward broader text recognition. Recent document parsers push toward full-page extraction, *e.g.*, DeepSeek-OCR [8] uses visual compression for OCR, while Dolphin-v2 [39] illustrates universal document parsing. MonkeyOCR [40] emphasizes structure-recognition-relation decomposition, and PaddleOCR-VL [41] targets compact multilingual document parsing. MinerU [42] takes a more inspectable pipeline route. General VLMs provide another axis of flexibility, *e.g.*, BLIP-2 [18] and InstructBLIP [19] align visual features with language models, while LLaVA [20] and Qwen2.5-VL [23] show strong instruction-following multimodal generation. Agentic parsing is also studied in ParseFixer [43], which applies selective correction to backbone Markdown outputs, and ParseBench [44], which evaluates agent-relevant parsing capabilities on enterprise documents.

2.3 Visual Descriptions and Accessibility

Accessibility work shows why visual description cannot be reduced to generic captioning. Conceptual Captions [15] demonstrates the scale available in web alt text, but weakly tied to accessibility intent. VizWiz [45] and VizWiz-Captions [46] foreground blind-user scenarios, where the useful answer depends on context. TextCaps [47] shows that descriptions may depend on visible text, and OCR-VQA [48] makes text reading part of visual question answering. Screen2Words [49] shifts the target to UI screens, while Alt-Text with Context [16] conditions descriptions on surrounding social text. Furthermore, ChartQA [50] requires visual and logical reasoning, and DePlot [51] recovers chart data before reasoning. Chart-to-Text [52] directly studies chart description, while PlotQA [53] emphasizes quantitative plot understanding. A useful description must be tied to the correct object, caption, page position, and surrounding reading flow. Read-Parse-Describe therefore evaluates alternative text as a field in the parse rather than as an isolated captioning output.

3 Read-Parse-Describe Dataset

3.1 Task Definition

We introduce Read-Parse-Describe, a dataset for document parsing with grounded visual description. This task is motivated by two limitations of existing document understanding benchmarks, as shown in Tab. 1. First, many layout datasets primarily focus on the physical localization of page regions but do not require models to recover a complete logic-aware document representation. This limitation is especially pronounced in scientific papers, where simple top-to-bottom ordering rules are frequently broken. Second, visual elements are often treated as layout categories rather than semantic objects. Figures, charts, diagrams, and screenshots may be localized, but the information they convey is rarely required to be presented in a grounded, accessible form.

3.2 Output Schema

Read-Parse-Describe therefore evaluates document understanding as a unified multi-modal parsing problem. The target output is an ordered list of document elements. Each element contains its bounding box, semantic category, reading-order index, recognized content when applicable, caption linkage, and visual description when applicable:

$$Y = [e_1, e_2, \ldots, e_n], \tag{1}$$

Table 1: Comparison of dataset target contracts. A checkmark indicates that the benchmark explicitly requires the corresponding field as part of its evaluation target. **Caption Links** refer to explicit associations between visual objects and their captions. **Structured Output** denotes a single page-level prediction that jointly contains layout, order, textual content, and caption links.

Dataset	Docs	Layout	Reading Order	Textual Elements	Visual Elements	Caption Links	Alternative Text	Structured Output
PubLayNet [10]	PDF	✓	–	–	✓	–	–	–
DocLayNet [12]	Mixed	✓	–	–	✓	–	–	–
DocBank [11]	PDF	✓	–	–	✓	–	–	–
M6Doc [30]	Mixed	✓	–	–	✓	–	–	–
D4LA [36]	Mixed	✓	–	–	✓	–	–	–
SciTSR [29]	Images	–	–	✓	–	–	–	–
ReadingBank [14]	Mixed	✓	✓	✓	–	–	–	–
DocVQA [31]	Images	–	–	✓	✓	–	–	–
ChartQA [50]	Images	–	–	–	✓	–	–	–
InfographicVQA [54]	Images	–	–	–	✓	–	–	–
DUDE [32]	Mixed	–	–	✓	✓	–	–	–
OmniDocBench [33]	Mixed	✓	✓	✓	✓	✓	–	✓
Read-Parse-Describe (Ours)	**PDF**	✓	✓	✓	✓	✓	✓	✓

where each element is represented as

$$e_i = \{\texttt{id}, \texttt{bbox}, \texttt{category}, \texttt{order}, \texttt{text}, \texttt{html}, \texttt{caption_id}, \texttt{alt_text}\}, \tag{2}$$

where `id` is the element identifier, `bbox` is a page-coordinate box $[x_1, y_1, x_2, y_2]$ in the rendered image coordinate system. `category` is the semantic layout label, and `order` is the canonical human reading-order index. The submitted list is expected to be sorted by `order`, *i.e.*, if list position and `order` disagree, evaluation first normalizes by the `order` field and leaves duplicate or missing indices to be penalized by matching and sequence metrics. `text` stores recognized text for textual regions, `html` stores table structure, `caption_id` links a visual object to caption, and `alt_text` stores a factual visual description. Non-applicable fields are set to `null`.

Terminology. An *ordered schema* is the full element list Y after sorting by the reading-order index. The central design choice is to make fields agree inside the same element object, not to score familiar subtasks in isolation. The contract has four coupled decisions:

Coupling	Design reason	Failure exposed
Element identity	Bind `id`, `category`, textual content, caption links, and descriptions to the same object.	Free-floating OCR, HTML, captions, or descriptions cannot be credited without the element they claim to describe.
Geometry as evidence	Treat `bbox` as the audit handle for text, tables, and visual descriptions.	A plausible description or table string attached to the wrong crop becomes measurable rather than silently accepted.
Canonical reading flow	Use `order` as human reading order rather than visual-nearest-neighbor order.	Two-column text, spanning figures, and separated captions expose sequence errors even when OCR is correct.
Caption semantics	Require `caption_id` and `alt_text` for visual instances when applicable.	A model must bind the object, caption, and description instead of generating generic image captions.

The couplings above make the task a test of *spatially grounded binding*. Region proposals decide the visual evidence available to OCR, table serialization, caption linkage, and alt-text generation. Reading-order decisions decide where captions and objects sit in the page narrative; and visual descriptions are auditable only when their evidence is localized.

3.3 Dataset Creation Pipeline

We collect open-access PDF papers from the ACM Digital Library and retain document metadata. Since the main contributions, core figures, methods, and experimental results of scientific papers are usually concentrated in the main body, we extract the first eight pages of each paper and keep shorter papers in their entirety. This page-level curation strategy preserves structurally and semantically informative regions of papers while reducing the dominance of long reference sections and appendices. The dataset presents a more balanced distribution across diverse scientific disciplines without diluting evaluation on repetitive textual lists. Additional detailed dataset statistics would be included in the technical appendix.

The visual elements include tables, diagrams, screenshots, photographs, charts, and long-tail illustrations. This diversity matters because visual-description generation is not the same problem for each object type, *e.g.*, a UI screenshot requires interface semantics. The layout labels are similarly heterogeneous, covering structural headers, scholarly objects, captions, references, lists, and paragraphs. We keep these categories explicit to avoid hiding errors that make scientific PDFs hard to parse. We split at the paper level with a 90:5:5 train/validation/test ratio after deduplication. Exact duplicate PDFs are removed using SHA-256 hashes, DOI metadata is used when available, normalized titles are matched otherwise, and near-duplicate title/abstract pairs above the similarity threshold are manually inspected. Paper-level splitting is important because adjacent pages share templates, section styles, figure numbering, author metadata, and sometimes repeated visual objects. A page-level split would overstate generalization for reading order and caption linkage.

3.4 Annotation Pipeline

Annotations are produced with a bootstrap-and-correct pipeline designed around cross-field consistency. A document parser first initializes layout boxes and labels. The generated boxes, semantic categories, caption links, and reading-order indices are then manually inspected and corrected together, rather than as independent layers. Corrections target common scientific-PDF failure modes: two-column reading order, spanning figures, captions separated from visual objects, footnotes, equations, author metadata, and fine-grained heading levels. This joint correction step is important: moving a box can change its text crop, its caption candidate, and its reading-order position.

Visual instances then receive alternative-text annotations through a triage loop: generate an initial description, score it against the accessibility rubric, refine low-score cases by human review, and audit the revised labels. Gemini 3 Pro [55] provides the initial descriptions and Gemini 3 Flash [56] scores factual accuracy, completeness, and prompt adherence on a 1–5 scale. The 1,408 descriptions scoring below 4 are routed to human refinement, then re-scored:

Score	Generated	Refined	Score	Generated	Refined
1	0	0	4	2,419	226
2	146	0	5	7,511	11,112
3	1,262	0	**Score-5 rate**	**66.2%**	**98.1%**

After refinement, the score-5 rate increases from 66.2% to 98.1%. A manual audit of 200 revised samples with previous scores below 5 gives 194/200 exact agreement between human and LLM scores. The technical appendix includes the prompt, scoring rubric, full score distribution, and audit matrix for this annotation stage.

The alt-text process is designed for benchmark consistency, not as an automatic accessibility deployment pipeline. The prompt asks for concise factual descriptions and discourages speculation beyond visible evidence. The scoring step checks whether the description is faithful, complete enough for the object type, and aligned with the prompt; human refinement is concentrated on low-score cases because these are most likely to contain missing relationships, hallucinated labels, or vague descriptions. We expose this procedure because visual-description labels are harder to audit than boxes or OCR strings, and the reader should be able to separate measured model behavior from annotation assumptions.

4 SARA: Spatial Alignment Residual Adapter

Read-Parse-Describe turns parsing into a binding problem, *i.e.*, each generated element must keep the attribute attached to the same page evidence. This creates a different pressure from ordinary VLM generation. A visual concatenation is useful due to the compression of image tokens into a sequence the decoder can consume, but that compression can blur the local spatial cues needed to decide where an element starts, how two-column regions should be ordered, and which caption belongs to which object. The design goal of SARA is therefore not to replace the base VLM, but to restore crop-level spatial evidence at the point where the decoder receives merged visual tokens. The key design choice is to preserve correspondence between each crop's pre-merger grid and its native post-merger tokens, enabling aligned feature re-injection without ground-truth input boxes.

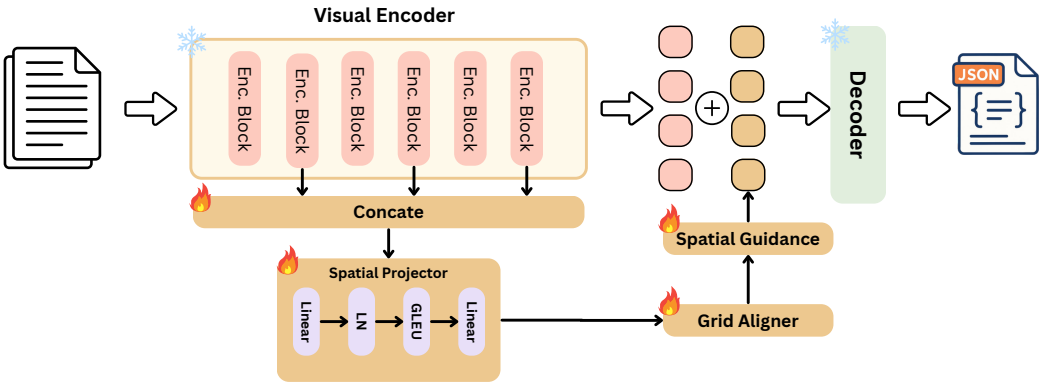


Figure 2: Document parsing model with SARA adapter, which is a plug-and-play spatial enhancer that leverages multi-layer visual features. Features from encoder layers are concatenated, followed by a grid-aware multi-crop alignment mechanism that preserves spatial structure. The aligned features serve as spatial guidance and are injected into the original tokens via a gated residual connection.

5 Method

SARA follow threefold design as shown in the Fig. 2: (1) Reuse intermediate visual states rather than only the final merged tokens, because earlier layers retain finer layout and boundary information. (2) Align the reused states to the same crop grid as the native visual merger, so the residual signal has the same token-level support as the decoder's original visual stream. (3) Inject the aligned guidance through a learnable residual gate, preserving the pretrained visual-language pathway while letting training decide how much spatial correction is useful.

Multi-layer spatial features. Let $\mathcal{S} = \{2, 5, 8, 11\}$ be the selected visual encoder layers. During the forward pass, hooks collect hidden states

$$\boldsymbol{X}^{(l)} \in \mathbb{R}^{L_v \times d_v}, \quad l \in \mathcal{S}, \tag{3}$$

where L_v is the pre-merger visual-token length and d_v is the visual feature dimension. We concatenate these states along the channel dimension:

$$\boldsymbol{X}_{\text{cat}} = \text{Concat}\left(\boldsymbol{X}^{(2)}, \boldsymbol{X}^{(5)}, \boldsymbol{X}^{(8)}, \boldsymbol{X}^{(11)}\right). \tag{4}$$

The concatenated feature is mapped to the merger dimension d_m by a small projector,

$$\boldsymbol{G} = \boldsymbol{W}_2\, \sigma\left(\text{LN}(\boldsymbol{W}_1 \boldsymbol{X}_{\text{cat}})\right), \tag{5}$$

where LN is LayerNorm and σ is GELU.

Crop-aware alignment. The visual merger reduces spatial resolution, so the guidance $\boldsymbol{G}$ must be aligned with the native merged tokens. For each crop with grid metadata (t_i, h_i, w_i), we slice the corresponding segment from $\boldsymbol{G}$, reshape it into its two-dimensional grid, and average-pool to the merged-grid resolution:

$$\boldsymbol{G}'_i = \text{Pool}_{2D}\left(\text{Reshape}(\boldsymbol{G}_i), (h_i/2, w_i/2)\right). \tag{6}$$

All crop-level features are flattened and concatenated into $\boldsymbol{G}' \in \mathbb{R}^{L_m \times d_m}$, matching the length and dimension of the original merged visual tokens T.

Residual injection. The enhanced visual tokens are

$$\hat{\boldsymbol{T}} = \boldsymbol{T} + \lambda \boldsymbol{G}', \tag{7}$$

where λ is a learnable scalar gate initialized to 1.0. The language decoder then generates the same ordered schema used by all baselines. SARA changes only the visual token pathway; it does not alter the benchmark labels, target schema, or evaluation metrics.

Shape summary and scope. The data flow is $\boldsymbol{X}^{(l)} \in \mathbb{R}^{L_v \times d_v}$, $\boldsymbol{X}_{\text{cat}} \in \mathbb{R}^{L_v \times |\mathcal{S}| d_v}$, $\boldsymbol{G} \in \mathbb{R}^{L_v \times d_m}$, and $\boldsymbol{G}', \boldsymbol{T}, \hat{\boldsymbol{T}} \in \mathbb{R}^{L_m \times d_m}$ after crop-aware pooling. SARA is not intended to replace document-specific supervision or table-aware decoding. The residual form keeps the original visual-token stream intact and lets training learn the aligned guidance.

6 Experiments

6.1 Settings and Metrics

We evaluate three categories of systems: general-purpose VLMs, specialized document models, and pipeline parsers. General-purpose VLMs include Qwen3.5 [57], Gemma4 [58], ChatGPT-5.4 [59], ChatGPT-5.6 [60], Gemini 3.6 Flash [61] and Gemini 3.1 Pro [62]. Specialized document models include DeepSeek-OCR [8], DeepSeek-OCR2 [63], Dolphin-v2 [39], MonkeyOCR [40], VGT [36], GOT-OCR 2.0 [7], Nougat [38], and PaddleOCR-VL [41]. Pipeline parsers include PP-StructureV3 [64] and MinerU [42].

All systems are evaluated under the same page-level output schema. Zero-shot systems use released checkpoints or official APIs without in-context examples. Full fine-tuning baselines serialize each target element list into text and optimize the standard autoregressive next-token objective on the training split. The validation split is used for model selection, and final results are reported on the held-out test split. For SARA runs, the base VLM is frozen and only the adapter pathway is trained. For generative VLMs, we use deterministic decoding whenever supported. Outputs from pipeline systems are converted into the unified schema.

Metrics. All metrics are computed after schema conversion. Minor syntax errors are repaired deterministically; if a prediction remains unparsable, the affected fields are treated as empty. F1 follows the IoU-matching convention used by PubLayNet [10], DocLayNet [12], and RoDLA [4]: a match requires the same category label and $\text{IoU} \geq 0.5$. mIoU is computed over matched true positives. Reading-order (RO) score uses normalized edit-distance similarity over matched element identifiers [14]. OCR accuracy is computed as normalized character-level edit-distance similarity after text normalization. For visual descriptions, we report METEOR [65] and BERTScore [66].

6.2 Main Results

Table 2 should be read as a contract diagnostic, not a leaderboard. The unified schema requires models to jointly recover localized elements, reading order, textual content, and visual descriptions. Pipeline parsers often retain strong OCR or localization performance, but they usually leave caption links and alt-text fields incomplete. General-purpose VLMs can produce richer schema fields, but they are less reliable in spatial grounding without adaptation. Document-specialized VLMs and SARA are complementary: the former excel at specialized parsing, while SARA improves the structural fields of a frozen general-purpose VLM and maintains competitive visual-description quality.

Table 2 reveals different strengths across the evaluated systems and reported metrics. Fine-tuned Dolphin-v2 achieves the strongest localization performance, reaching 0.9826 F1 and 0.9588 mIoU. It also obtains the highest RO score, 0.9857. Fine-tuned DeepSeek-OCR achieves the highest METEOR, 0.3998, while GPT-5.6 achieves the highest BERTScore, 0.9004. DeepSeek-OCR also maintains a high RO score of 0.9825. Among the evaluated proprietary zero-shot models, GPT-5.6 obtains the highest F1, RO, and description scores, while Gemini 3.6 Flash obtains the highest mIoU. Including Gemini 3.1 Pro extends the comparison to the requested Pro-series model under the same schema. These results suggest that spatially precise parsing and open-ended visual description remain partially distinct capabilities.

Pipeline parsers remain competitive on some low-level fields but are less complete under the unified output schema. For example, MinerU reaches 0.9671 OCR accuracy and 0.9151 mIoU, but its RO score is 0.6757, and it does not produce visual alternative text under the evaluated schema. This does not mean that pipeline systems are ineffective. Instead, it identifies the fields that require additional modules or tighter integration when the target is a complete accessible page parse.

Compared with the Qwen3.5-2B full fine-tuning baseline, SARA substantially improves structural performance while keeping the base VLM frozen and training with 83.99M parameters. F1 increases

Table 2: Main test-set results under the unified Read-Parse-Describe output schema. Rows are grouped by evaluation setting to separate **Zero-shot Models**, **Full Fine-Tuning**, which means fine-tuning the whole model, and **Adapter Tuning**, which means only fine-tuning the adapter part. -- indicates that the corresponding field is not produced by the evaluated system. **bold** and underlined scores denote the best and second-best results for each metric, respectively.

Model	Trainable Params	DLA		OCR Acc.↑	RO Score↑	Alt. Text	
		F1↑	mIoU↑			METEOR↑	BERT ↑
Zero-shot Models							
GOT-OCR 2.0	--	--	--	0.8652	0.5330	--	--
Nougat	--	--	--	0.7327	0.4640	--	--
Dolphin-v2	--	0.9124	0.9265	0.9307	0.8221	--	--
MonkeyOCR	--	0.6030	0.8949	0.9684	0.6254	--	--
VGT	--	0.6795	0.8540	0.6304	0.3833	--	--
PaddleOCR-VL	--	0.7564	0.9458	0.9043	0.6619	--	--
PP-StructureV3	--	0.6375	0.8870	0.9558	0.5054	--	--
MinerU	--	0.7171	0.9151	0.9671	0.6757	--	--
ChatGPT-5.4	--	0.6978	0.6454	0.9796	0.7253	0.3506	0.8751
ChatGPT-5.6	--	0.7489	0.7781	0.9448	0.8967	0.3793	**0.9004**
Gemini 3.6 Flash	--	0.6909	0.8658	0.9544	0.8183	0.3744	0.8841
Gemini 3.1 Pro	--	0.7421	0.8551	0.9633	0.8337	0.3785	0.8961
Gemma4-E4B	--	0.5473	0.7674	0.8977	0.5256	0.1679	0.8455
DeepSeek-OCR	--	0.5987	0.6074	0.9064	0.5003	0.2562	0.8280
DeepSeek-OCR2	--	0.5806	0.5963	0.8688	0.4830	0.2604	0.8332
Full fine-tuning							
Qwen3.5-0.8B	752.39M	0.7177	0.8345	0.9475	0.9605	0.3666	0.8904
Qwen3.5-2B	1881.83M	0.7336	0.8350	0.9408	0.9628	0.3741	0.8928
Gemma4-E4B	7463.01M	0.9358	0.8836	0.9452	0.9628	0.3047	0.8821
DeepSeek-OCR	2937.36M	0.9644	0.9492	0.9679	0.9825	**0.3998**	0.8968
DeepSeek-OCR2	2935.88M	0.8783	0.9189	0.9416	0.9231	0.3229	0.8094
Dolphin-v2	3085.93M	**0.9826**	**0.9588**	**0.9751**	**0.9857**	0.3396	0.8838
Adapter Tuning							
Gemma4-E4B + LoRA	103.82M	0.7687	0.7238	0.9164	0.9451	0.2871	0.8753
Gemma4-E4B + SARA	158.82 M	0.9437	0.9136	0.9474	0.9288	0.3009	0.8746
Qwen3.5-2B + LoRA	48.37M	0.7004	0.6845	0.9673	0.9304	0.3586	0.8652
Qwen3.5-2B + SARA	83.99 M	0.9189	0.8919	0.9636	0.9774	0.3834	0.8946

to 0.9189, mIoU to 0.8919, OCR accuracy to 0.9636, and RO score to 0.9774. Its alt-text scores, 0.3834 METEOR and 0.8946 BERTScore, are also competitive with the strongest fine-tuned document VLMs. On Gemma4-E4B, SARA also improves F1 and mIoU relative to full fine-tuning, but RO and both description metrics decrease slightly. Compared with LoRA-only adaptation, SARA improves F1 and mIoU by 0.2185 and 0.2074 on Qwen and 0.1750 and 0.1898 on Gemma, respectively. Other metrics show trade-offs: Qwen LoRA has higher OCR accuracy, while Gemma LoRA has higher RO and slightly higher BERTScore.

Interpreting the joint table. The main result is not that one model is uniformly best, but that single-axis strengths do not necessarily imply a coherent page parse. A model with high OCR but low RO may still fail to recover columns or reading sequences. A model with strong F1 but no alt-text may be useful as a front end, but it is incomplete as an accessible parser. Conversely, a model with strong alt-text but weak mIoU may describe objects that are poorly localized. Reporting these fields side by side makes such tradeoffs visible without imposing a deployment-specific ranking.

6.3 Ablations and Analysis

Table 3 isolates the contribution of each SARA component on Qwen3.5-2B. The full fine-tuning row is included as a reference, while all adapter rows keep the backbone frozen. Grid alignment alone provides a modest improvement over full fine-tuning, especially for RO and OCR. In contrast, projector-only training underperforms full fine-tuning, indicating that the gains do not come merely

Table 3: Component ablation with Qwen3.5-2B.

Configuration	Trainable Params	F1↑	mIoU↑	OCR Acc.↑	RO Score↑
Full FT	1881.83M	0.7336	0.8350	0.9408	0.9628
Grid align only	81.67M	0.7541	0.8440	0.9566	0.9762
Projector only	35.96M	0.6830	0.8139	0.9134	0.9262
Grid align + projector	83.99M	**0.9189**	**0.8919**	**0.9636**	**0.9774**

Table 4: Feature-layer selection ablation.

Layers	F1↑	mIoU↑	OCR Acc.↑	RO Score↑
{2, 5}	0.6855	0.7682	0.9076	0.9372
{16, 18, 20, 22}	0.7221	0.8175	0.9256	0.9412
{2, 5, 8, 11}	**0.9192**	**0.8968**	**0.9602**	**0.9736**

from adding trainable parameters. The full configuration, which combines grid-aware alignment with projector training, produces the largest improvement across all four structural metrics.

Table 4 studies which visual feature layers are most useful for the adapter. Early-to-mid layers, {2,5,8,11}, outperform both shallow-only layers and late-layer selections. This pattern is consistent with the design goal of preserving local spatial cues while retaining sufficient semantic abstraction for document elements. On both tested backbones, SARA improves localization relative to full fine-tuning, while its effects on reading order and description similarity vary. Overall, these ablations suggest that SARA primarily acts as a spatial-structure adapter rather than as a general-purpose language-generation module. This also explains why alt-text improvements are more mixed than the localization and reading-order gains.

Training-data scale. On Qwen3.5-2B, SARA trails full fine-tuning on all four structural metrics at 10% training data and yields mixed results at 30%, while its structural advantage emerges at 50% and 100%. The complete comparison is reported in Appendix E.

6.4 Qualitative Results

Figure 3 compares annotation mapping results from the ground truth, the base Qwen3.5-2B model, full fine-tuning, and SARA. The base model often produces incomplete or poorly grounded mappings. Full fine-tuning improves schema compliance, but spatial alignment errors can remain. In comparison, SARA produces element mappings that are more consistent with the ground-truth page structure, especially for localized regions and their associated schema fields.

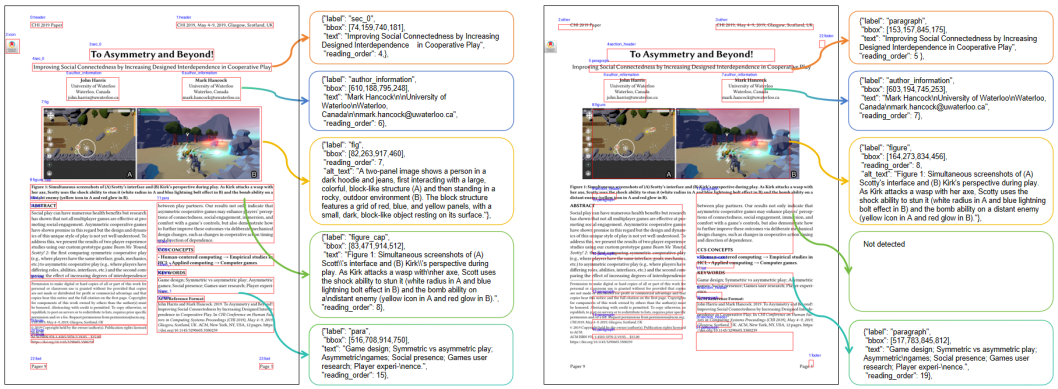
7 Conclusion

We introduced Read-Parse-Describe, a unified benchmark for scientific-document parsing, and SARA, a spatial adapter tailored to the same page-level output contract. The task jointly evaluates localized elements, labels, reading order, textual and table content, caption links, and alternative text within a single ordered parse. Experiments show that existing systems tend to fail different parts in evaluation, while frozen-backbone SARA tuning substantially improves structural parsing for a general VLM.

Limitations. Read-Parse-Describe is limited to open-access ACM scientific PDFs and primarily covers the first eight pages, leaving late appendices and references underrepresented. Alt-text annotations rely on LLM assistance with human refinement, and closed-source pre-training exposure cannot be fully excluded. The benchmark is intended to support accessible document understanding, search, and document mining, but generated accessibility metadata should not be treated as deployment-ready without further validation.

Acknowledgments and Disclosure of Funding

This work was supported in part by Helmholtz Association of German Research Centers, in part by the Ministry of Science, Research and the Arts of Baden-Württemberg (MWK) through the Cooperative Graduate School Accessibility through AI-based Assistive Technology (KATE) under Grant BW6-03, and in part by the National Natural Science Foundation of China under Grant No. 62503166, in part by the Hunan Provincial Research and Development Project under Grant number 2026QK3018, and in part by the Yuelushan Industrial Innovation Center. This work was partially performed on the HoreKa supercomputer funded by the MWK and by the Federal Ministry of Education and Research, partially on the HAICORE@KIT partition supported by the Helmholtz Association Initiative and Networking Fund, and partially on bwForCluster Helix supported by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1597-1 FUGG.



(a) Ground truth (b) Qwen3.5-2B

(c) Qwen3.5-2B + full fine-tuning (d) Qwen3.5-2B + SARA

Figure 3: Qualitative comparison of annotation mapping results. All model-based methods use Qwen3.5-2B as the backbone. We compare the ground-truth annotations with outputs from the base model, full fine-tuning, and the proposed SARA adapter.

References

[1] 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 and Data Mining*, pages 1192–1200, 2020.

[2] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. LayoutLMv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 2579–2591, 2021.

[3] 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*, pages 4083–4091, 2022.

[4] Yufan Chen, Jiaming Zhang, Kunyu Peng, Junwei Zheng, Ruiping Liu, Philip Torr, and Rainer Stiefelhagen. Rodla: Benchmarking the robustness of document layout analysis models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15556–15566, 2024.

[5] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. Image-based table recognition: Data, model, and evaluation. In *European Conference on Computer Vision*, pages 564–580, 2020.

[6] 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 *European Conference on Computer Vision*, pages 498–517, 2022.

[7] Haoran Wei, Chenglong Liu, Jinyue Chen, Jia Wang, Lingyu Kong, Yanming Xu, Zheng Ge, Liang Zhao, Jianjian Sun, Yuang Peng, et al. General ocr theory: Towards ocr-2.0 via a unified end-to-end model. *arXiv preprint arXiv:2409.01704*, 2024.

[8] Haoran Wei, Yaofeng Sun, and Yukun Li. Deepseek-ocr: Contexts optical compression. *arXiv preprint arXiv:2510.18234*, 2025.

[9] Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Yuhao Dan, Chenlin Zhao, Guohai Xu, Chenliang Li, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. mPLUG-DocOwl: Modularized multimodal large language model for document understanding. *arXiv preprint arXiv:2307.02499*, 2023.

[10] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. Publaynet: Largest dataset ever for document layout analysis. In *International Conference on Document Analysis and Recognition*, pages 1015–1022, 2019.

[11] Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. Docbank: A benchmark dataset for document layout analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 949–960, 2020.

[12] Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S. Nassar, and Peter W. J. Staar. Doclaynet: A large human-annotated dataset for document-layout segmentation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3743–3751, 2022.

[13] 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*, pages 4634–4642, 2022.

[14] Zilong Wang, Yiheng Xu, Lei Cui, Jingbo Shang, and Furu Wei. Layoutreader: Pre-training of text and layout for reading order detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4735–4744, 2021.

[15] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2556–2565, 2018.

[16] Nikita Srivatsan, Sofia Samaniego, Omar Florez, and Taylor Berg-Kirkpatrick. Alt-text with context: Improving accessibility for images on twitter. In *International Conference on Learning Representations*, 2024.

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[18] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, 2023.

[19] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N. Fung, and Steven C. H. Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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

[22] Tengchao Lv, Yupan Huang, Jingye Chen, Lei Cui, Shuming Ma, Yaoyao Chang, Shaohan Huang, Wenhui Wang, Li Dong, Weiyao Luo, Shaoxiang Wu, Guoxin Wang, Cha Zhang, and Furu Wei. Kosmos-2.5: A multimodal literate model. *arXiv preprint arXiv:2309.11419*, 2023.

[23] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[24] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

[25] Adam W. Harley, Alex Ufkes, and Konstantinos G. Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *International Conference on Document Analysis and Recognition*, pages 991–995, 2015.

[26] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. FUNSD: A dataset for form understanding in noisy scanned documents. In *International Conference on Document Analysis and Recognition Workshops*, pages 1–6, 2019.

[27] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. CORD: A consolidated receipt dataset for post-OCR parsing. In *Document Intelligence Workshop at Neural Information Processing Systems*, 2019.

[28] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. ICDAR2019 competition on scanned receipt OCR and information extraction. *arXiv preprint arXiv:2103.10213*, 2021.

[29] Zewen Chi, Heyan Huang, Heng-Da Xu, Houjin Yu, Wanxuan Yin, and Xian-Ling Mao. Complicated table structure recognition. *arXiv preprint arXiv:1908.04729*, 2019.

[30] Hiuyi Cheng, Peirong Zhang, Sihang Wu, Jiaxin Zhang, Qiyuan Zhu, et al. M6doc: A large-scale multi-format, multi-type, multi-layout, multi-language, multi-annotation category dataset for modern document layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15138–15147, 2023.

[31] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2199–2208, 2021.

[32] Jordy Van Landeghem, Rubèn Tito, Łukasz Borchmann, Michał Pietruszka, Pawel Joziak, Rafal Powalski, et al. Document understanding dataset and evaluation (DUDE). In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19528–19540, 2023.

[33] Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, et al. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24838–24848, 2025.

[34] Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, et al. MMLongBench-Doc: Benchmarking long-context document understanding with visualizations. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[35] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R. Manmatha. DocFormer: End-to-end transformer for document understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 993–1003, 2021.

[36] Cheng Da, Chuwei Luo, Qi Zheng, and Cong Yao. Vision grid transformer for document layout analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19462–19472, 2023.

[37] Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2Struct: Screenshot parsing as pretraining for visual language understanding. In *International Conference on Machine Learning*, 2023.

[38] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents. *arXiv preprint arXiv:2308.13418*, 2023.

[39] Hao Feng, Wei Shi, Ke Zhang, Xiang Fei, Lei Liao, Dingkang Yang, Yongkun Du, Xuecheng Wu, Jingqun Tang, Yang Liu, Hong Chen, and Can Huang. Dolphin-v2: Universal document parsing via scalable anchor prompting. *arXiv preprint arXiv:2602.05384*, 2026.

[40] Zhang Li, Yuliang Liu, Qiang Liu, Zhiyin Ma, Ziyang Zhang, Shuo Zhang, Zidun Guo, Jiarui Zhang, Xinyu Wang, and Xiang Bai. Monkeyocr: Document parsing with a structure-recognition-relation triplet paradigm. *arXiv preprint arXiv:2506.05218*, 2025.

[41] Cheng Cui, Ting Sun, Suyin Liang, Tingquan Gao, Zelun Zhang, Jiaxuan Liu, Xueqing Wang, Changda Zhou, Hongen Liu, Manhui Lin, et al. Paddleocr-vl: Boosting multilingual document parsing via a 0.9b ultra-compact vision-language model. *arXiv preprint arXiv:2510.14528*, 2025.

[42] Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, et al. Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*, 2024.

[43] LeKai Yu, Hao Liu, Kun Wang, Zhiran Li, Ruping Cao, Fan Liu, and Yupeng Hu. Parsefixer: An agentic framework for document parsing via selective multimodal correction. *arXiv preprint arXiv:2606.11977*, 2026.

[44] Boyang Zhang, Sebastián G. Acosta, Preston Carlson, Sacha Bron, Pierre-Loïc Doulcet, Daniel B. Ospina, and Simon Suo. Parsebench: A document parsing benchmark for ai agents, 2026.

[45] Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3617, 2018.

[46] Danna Gurari, Yinan Zhao, Meng Zhang, and Nilavra Bhattacharya. Captioning images taken by people who are blind. In *European Conference on Computer Vision*, pages 417–434, 2020.

[47] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. TextCaps: A dataset for image captioning with reading comprehension. In *European Conference on Computer Vision*, pages 742–758, 2020.

[48] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. OCR-VQA: Visual question answering by reading text in images. In *International Conference on Document Analysis and Recognition*, pages 947–952, 2019.

[49] Bryan Wang, Gang Li, Xin Zhou, Zhourong Chen, Tovi Grossman, and Yang Li. Screen2Words: Automatic mobile UI summarization with multimodal learning. In *Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology*, pages 498–510, 2021.

[50] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*, 2022.

[51] Fangyu Liu, Julian Eisenschlos, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Wenhu Chen, Nigel Collier, and Yasemin Altun. DePlot: One-shot visual language reasoning by plot-to-table translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10381–10399, 2023.

[52] Shankar Kantharaj, Rixie Tiffany Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. Chart-to-text: A large-scale benchmark for chart summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 4005–4023, 2022.

[53] Nitesh Methani, Pritha Ganguly, Mitesh M. Khapra, and Pratyush Kumar. PlotQA: Reasoning over scientific plots. In *IEEE Winter Conference on Applications of Computer Vision*, pages 1516–1525, 2020.

[54] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V. Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022.

[55] Google DeepMind. Gemini 3 Pro Model Card. https://deepmind.google/models/model-cards/gemini-3-pro/, December 2025.

[56] Google DeepMind. Gemini 3 Flash Model Card. https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-Flash-Model-Card.pdf, December 2025. Model card dated December 17, 2025. Accessed: 2026-05-07.

[57] Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026.

[58] Google DeepMind. Gemma 4 Model Card. https://ai.google.dev/gemma/docs/core/model_card_4, April 2026.

[59] OpenAI. Introducing GPT-5.4. https://openai.com/index/introducing-gpt-5-4/, March 2026.

[60] OpenAI. Gpt-5.6: Frontier intelligence that scales with your ambition. https://openai.com/index/gpt-5-6/, July 2026.

[61] Google DeepMind. Gemini 3.6 flash: Model card. https://deepmind.google/models/model-cards/gemini-3-6-flash/, 2026.

[62] Google DeepMind. Gemini 3.1 pro: Model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, 2026.

[63] Haoran Wei, Yaofeng Sun, and Yukun Li. DeepSeek-OCR 2: Visual causal flow, 2026.

[64] Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, Yue Zhang, Wenyu Lv, Kui Huang, Yichao Zhang, Jing Zhang, Jun Zhang, Yi Liu, Dianhai Yu, and Yanjun Ma. PaddleOCR 3.0 Technical Report, 2025.

[65] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, 2005.

[66] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*, 2020.

A Dataset Details

This appendix is supplementary reading. The main text defines the task contract and reports the primary experimental conclusions; the material below gives the distributional and procedural detail needed to inspect the benchmark more closely.

A.1 Label Taxonomy and Corpus Profile

Read-Parse-Describe uses a 23-label taxonomy rather than a small generic layout set. The goal is to keep elements that matter for reading order, reflow, caption linkage, and auditability visible as distinct targets. Headings, paragraph fragments, captions, equations, code blocks, metadata, references, lists, and visual objects therefore remain separately labeled.

Table 5: Semantic label taxonomy used in Read-Parse-Describe.

ID	Description	Label	ID	Description	Label
1	Paper Title	`sec_0`	13	Table	`tab`
2	Level-1 Heading	`sec_1`	14	Figure-Caption	`figure-cap`
3	Level-2 Heading	`sec_2`	15	Table-Caption	`table-cap`
4	Level-3 Heading	`sec_3`	16	Annotation	`anno`
5	Level-4 Heading	`sec_4`	17	Formula	`equ`
6	Paragraph	`para`	18	Code Block	`code`
7	Spanning Paragraph	`half_para`	19	Catalog	`catalogue`
8	Header	`header`	20	Reference	`reference`
9	Footer	`foot`	21	Author-information	`author-info`
10	Footnote	`fnote`	22	Conference logo	`icon`
11	Watermark	`watermark`	23	List	`list`
12	Figure	`fig`			

The benchmark itself contains 2,402 curated scientific documents and 18,163 continuous pages. The domain chart below summarizes the broader analyzed source-corpus pool of 2,151 ACM articles from 2020–2024; the residual slice groups domains that were not individually named in the figure.

Table 6: Primary Domain Distribution of Analyzed Articles from 2020 - 2024

Category	Count	Percentage
AI & ML	271	12.60%
Accessibility	255	11.85%
AR/VR/XR	235	10.93%
Design & HCI Methods	149	6.93%
Privacy & Social	103	4.79%
Health & Wellbeing	95	4.42%
Demographics	71	3.30%
Games, EdTech & Creativity	65	3.02%
Ethics, Fairness & CSCW	61	2.84%
Hardware & Fabrication	56	2.60%
Others / Long-tail	790	36.72%
Total	**2,151**	**100.00%**

A.2 Page Curation and Element Distributions

The first-eight-page rule is intended to increase the density of multimodal scientific content while keeping each paper as a continuous page sequence. It preserves the early-paper regions where abstracts, methods, figures, tables, captions, equations, and author metadata interact most strongly. Later pages are not treated as unimportant; they are underrepresented because they are disproportionately reference-heavy in the analyzed full-text subset.

The following tables decompose the visual and layout population from two complementary angles. Subject categories describe the broad disciplinary context of visual instances. Image types describe the visual form that the alt-text and caption-linking steps must handle. Aggregated layout groups

Table 7: Impact of the 8-Page Truncation Strategy on Layout Element Distribution. By comparing the element proportions in Pages 1–8 versus Pages ≥ 9 (estimated on a full-text subset), we observe that the truncation effectively preserves high-value multimodal objects while filtering out visually monotonous reference lists.

Element Category	**Pages 1–8 (%)**	**Pages ≥ 9 (%)**	**Density (1–8)**	**Density (≥ 9)**
Title, Abstract & Metadata	100.00%	0.00%	1.2 / page	0.0 / page
Equations & Math Blocks	78.42%	21.58%	4.5 / page	1.2 / page
Figures & Captions	69.15%	30.85%	2.1 / page	0.9 / page
Body Paragraphs	62.30%	37.70%	12.4 / page	7.5 / page
Tables & Algorithms	54.80%	45.20%	0.8 / page	0.7 / page
References	8.25%	91.75%	0.5 / page	15.6 / page
Overall Multimodal Complexity	**High**	**Low**	**Diverse**	**Monotonous**

show how much of the annotation budget is spent on document structure, metadata, scholarly objects, captions, references, and fragments.

Table 8: Topic Type Distribution

Category	**Count**	**Percentage**
Computer Science & AI	6,443	56.8%
Social Science	2,181	19.2%
Engineering	1,359	12.0%
Medicine & Healthcare	652	5.8%
Other	245	2.2%
Biology	110	1.0%
Physics	106	0.9%
Mathematics	98	0.9%
Economics & Finance	72	0.6%
Environmental Science	50	0.4%
Chemistry	22	0.2%
Total	**11,338**	**100.0%**

Table 9: Image Type Distribution

Image Type	**Count**	**Percentage**
Table	2758	24.3%
Diagrammatic Instruction / Concept	2086	18.4%
User Interface / Screenshot	1743	15.4%
Photographic Image	1529	13.5%
Flowchart / Block Diagram	1000	8.8%
Bar Chart / Histogram	942	8.3%
Line Graph	466	4.1%
Other	371	3.3%
Scatter Plot	205	1.8%
Heatmap	147	1.3%
Pie Chart	27	0.2%
Anatomical Illustration	27	0.2%
Circuit Diagram	21	0.2%
Radar Chart	12	0.1%
Molecular Structure	4	0.0%
Total	**11338**	**100.0%**

B Alt-text Annotation Details

Visual descriptions are harder to audit than boxes or OCR strings because a fluent sentence can still be vague, hallucinated, or attached to the wrong object. We therefore separate the alt-text process

Table 10: Distribution of high-value layout elements.

Category	Label	Count	Percentage
Structural Headers	`sec_1`, `sec_2`, `sec_0`	44,617	35.64%
Academic Metadata	`author-info`, `fnote`	28,381	22.66%
Scholarly Objects	`fig`, `tab`, `equ`, `code`	12,336	9.85%
Captions and References	`figure-cap`, `table-cap`, `reference`	21,381	17.08%
Layout Fragments	`half_para`, `list`, `icon`	17,564	14.03%
Others	`anno`, `sec_3`, `sec_4`, `watermark`, etc.	939	0.74%
Total	–	**125,218**	**100.00%**

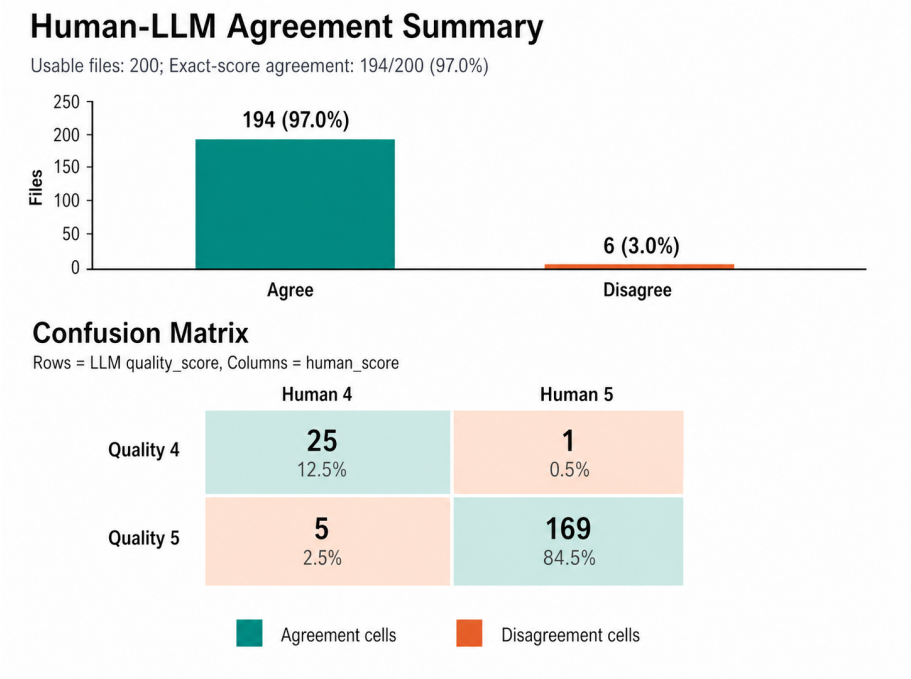


Figure 4: Human-LLM agreement audit over 200 revised samples. Agreement is measured after low-score descriptions are refined and re-scored.

into generation, rubric-based scoring, targeted human refinement, and a manual audit. The detailed prompt and rubric below are provided to make the label assumptions inspectable.

The score distribution reports the same 11,338 visual instances before and after human refinement. Low-scoring generated descriptions are revised, then re-scored under the same rubric.

C Evaluation Metric

To quantitatively assess the performance of the fine-tuned models in handling complex scholarly layouts, we employ a multi-dimensional evaluation suite. This suite covers three complementary aspects: (1) *physical localization* of layout elements, (2) *logical sequencing* of the reading flow, and (3) *textual and tabular accuracy* of the extracted content.(4) *Accessibility Metadata Evaluation*.Such a decomposition is necessary because a model can, for example, detect elements correctly but still fail to order them properly, or produce fluent text while misplacing bounding boxes.

1. Physical Layout Analysis: F1-score and mIoU

For Document Layout Analysis (DLA), we evaluate the model's ability to correctly identify and localize structural entities (e.g., figures, tables, and section blocks).

Table 11: Detailed Prompt Configurations and Expanded Evaluation Rubric for Alt-text Generation

Configuration	Detailed Content / Implementation
Generation Prompt	"Describe this image in 1–3 concise sentences for accessibility. If the image contains multiple panels or subfigures, summarize them as a single combined figure rather than describing each panel separately. Focus only on the key visual information, such as the type of chart or image, the main subjects, and clearly visible labels or visual patterns. Do not use phrases like 'image of' or 'picture of'. Provide a factual description based only on what is visible, without interpreting or explaining the underlying data."
Generation Parameters	**Temperature:** 0.2 **Top-p:** 0.9 **Max New Tokens:** 256 **Repetition Penalty:** 1.05
Evaluation Role	**System Prompt:** You are a rigorous Image Accessibility Auditor specializing in scientific and technical content. Your goal is to determine if the provided 'Alt-text' accurately represents the visual information for a visually impaired user. Evaluate strictly based on Accuracy, Completeness, and Constraint Adherence.
Evaluation Rubric	**Score 1 (Failure - Misleading/Hallucination):** Completely irrelevant, contains gross data hallucinations, or generates pure technical noise. Fails to identify the basic type of the image. Actively misleads the visually impaired user. **Score 2 (Poor - Severe Omissions/Violations):** Identifies the general chart/image type but suffers from severe omissions (e.g., missing critical axes, legends, or main subjects). Alternatively, it severely violates negative constraints (e.g., provides subjective interpretations of data instead of visual facts). **Score 3 (Acceptable - Basic but Flawed):** Correctly identifies the main subjects and chart type. However, it lacks specific critical details (e.g., distinct visual trends or key labels). May fail minor constraints (e.g., exceeds the 3-sentence limit, or includes filler phrases like "This is a picture of"). **Score 4 (Good - Professional & Factual):** Clear, objective, and accurate. Successfully captures the chart type, major variables, key labels, and significant visual patterns. Strictly adheres to negative constraints. May only miss minor, non-essential visual nuances or slightly lack extreme conciseness. **Score 5 (Excellent - Perfect Accessibility):** Highly concise (strictly 1-3 sentences), factual, and deeply informative. Seamlessly summarizes complex or multi-panel figures into a cohesive whole. Provides a visually impaired user with the exact informational equivalent of a quick visual scan, devoid of any over-interpretation or fluff.

Table 12: Alt text generation: LLM-generated vs LLM+human_modified

category	LLM generated (Count)	LLM + human_modified (Count)
Score 1	0	0
Score 2	146	0
Score 3	1262	0
Score 4	2419	226
Score 5	7511	11112
Score 5	**66.2%**	**98.1%**

F1-score. We compute the harmonic mean of Precision and Recall for each label category. A detection is considered a True Positive (TP) if (i) the predicted category matches the ground-truth category and (ii) the overlap between predicted and ground-truth regions exceeds a predefined threshold (typically based on IoU). The metric is defined as:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

In practice, F1-score summarizes the trade-off between missing elements (low Recall) and producing spurious detections (low Precision).

mIoU (mean Intersection over Union). To measure geometric localization accuracy, we compute the Intersection over Union (IoU) between each predicted bounding box B_p and the corresponding ground-truth box B_{gt}. The mean IoU over N matched instances is reported as:

$$mIoU = \frac{1}{N} \sum_{i=1}^{N} \frac{Area(B_{p,i} \cap B_{gt,i})}{Area(B_{p,i} \cup B_{gt,i})}$$

Compared to F1-score, mIoU is more sensitive to precise boundary alignment, which is critical for downstream tasks such as figure-to-caption association or table cell reconstruction.

2. Logical Reading Order: Normalized Edit Distance (NED)

To evaluate Reading Order (RO), we represent the sequence of detected blocks as a string of identifiers and compare it with the ground-truth sequence. We use the Levenshtein edit distance to measure the minimal number of insertions, deletions, and substitutions needed to transform the predicted order into the correct order. To ensure comparability across documents with different numbers of blocks, we normalize the distance and define the RO score as:

$$RO_{Score} = 1 - \frac{Levenshtein(S_p, S_{gt})}{\max(|S_p|, |S_{gt}|)}$$

where S_p and S_{gt} denote the predicted and ground-truth block sequences, respectively. A higher score indicates a more faithful logical flow, which is essential for reflowable rendering and accessible reading.

3. Textual and Tabular Accuracy: ED and TEDS

OCR accuracy (Edit Distance, ED). For general text extraction, we compute character-level edit distance between the recognized text and the reference transcription. This metric directly reflects transcription fidelity and is particularly informative for scientific documents containing symbols, subscripts, and uncommon terminology.

Table reconstruction (TEDS). For table parsing, we use the Tree Edit Distance Based Similarity (TEDS), which evaluates both the global structure (row/column organization) and the cell content by representing tables as tree structures. The similarity is defined as:

$$TEDS(T_a, T_b) = 1 - \frac{Dist_{tree}(T_a, T_b)}{\max(|T_a|, |T_b|)}$$

where T_a and T_b are the predicted and ground-truth table trees. This metric is more aligned with human judgment than plain text metrics, because it penalizes structural errors (e.g., merged/split cells) in addition to textual mistakes.

4. Accessibility Metadata Evaluation: METEOR Score and BERTScore

To evaluate the quality of the generated Alt-text for figures and tables, we employ the METEOR and BERTScore metrics. While metrics like BLEU primarily focus on exact n-gram overlaps, METEOR and BERTScore are designed to capture the semantic nuances of scholarly descriptions more effectively.

The Rationale for Choosing METEOR In the context of academic document reconstruction, the description of a figure (Alt-text) can be linguistically diverse. METEOR is superior for this task due to its unique features: Synonym Matching: It utilizes a dictionary (like WordNet) to match synonyms. For instance, if the ground truth says "Diagram" and the model generates "Schematic," METEOR recognizes this as a valid match, whereas BLEU would penalize it. Stemming: It matches words with the same root (e.g., "clustering" and "clustered"), which is crucial for maintaining the technical accuracy of alt-text.Alignment-based Scoring: It creates an explicit alignment between the predicted string and the reference, penalizing incorrect word orders through a "fragmentation penalty."

Mathematical Formulation The METEOR score is calculated by combining the harmonic mean of precision (P) and recall (R), with a specific emphasis on recall, followed by a penalty for incorrect ordering. First, the F-score is computed as:

$$F_{mean} = \frac{P \cdot R}{\alpha \cdot P + (1-\alpha) \cdot R}$$

Typically, α is set to give more weight to recall, ensuring that the descriptive content of the figure is fully captured. Then a Penalty (Pen) is calculated based on the "chunkiness" of the alignment (how fragmented the matches are):

$$Pen = \gamma \cdot \left(\frac{\#chunks}{\#unigrams_matched} \right)^{\beta}$$

The final METEOR Score is:

$$Score = F_{mean} \cdot (1 - Pen)$$

The Rationale for Choosing BERTScore While METEOR effectively accounts for lexical and morphological variations, scholarly descriptions often convey identical visual evidence through substantially different syntactic patterns. BERTScore is selected to capture this deep semantic equivalence beyond surface-level n-gram overlap. By generating contextualized token embeddings from pre-trained language models, it dynamically evaluates polysemous technical terms within their precise scholarly contexts. Furthermore, BERTScore relies on continuous pairwise cosine similarities rather than rigid discrete lookups, which fairly credits semantically rich paraphrases without external dictionaries. This focus on contextualized conceptual alignment rather than rigid syntactic templates ensures a reliable assessment of whether the essential semantic takeaways of complex scientific figures are preserved for visually impaired readers.

Mathematical Formulation BERTScore computes greedy token matching over contextualized representations. Let $\hat{x} = \langle \hat{x}_1, \ldots, \hat{x}_l \rangle$ and $x = \langle x_1, \ldots, x_k \rangle$ denote the generated description and ground-truth reference, with $\hat{\mathbf{x}}_j$ and $\mathbf{x}_i$ representing their respective pre-trained normalized token embeddings. The pairwise cosine similarities are greedily aligned to derive contextual recall (R_{BERT}) and precision (P_{BERT}):

$$R_{BERT} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j, \quad P_{BERT} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j. \tag{8}$$

The final reported semantic metric is the harmonic mean (F_{BERT}) combining both directional alignments:

$$F_{BERT} = 2 \cdot \frac{P_{BERT} \cdot R_{BERT}}{P_{BERT} + R_{BERT}}. \tag{9}$$

D Evaluation Details

All reported systems are converted into the same element schema before scoring. Layout matching requires category agreement and an IoU threshold of 0.5. DLA-F1 is computed over matched layout elements, and mIoU averages the IoU of true-positive matches. Reading order is evaluated over matched element identifiers, so a model must both find the relevant element and place it correctly in the sequence. OCR is normalized before edit-distance scoring, and table content is scored through the serialized `html` field when a system produces it. Visual descriptions are scored only for visual instances with reference alternative text; a missing description for an applicable object receives zero for the description metric.

Unparsable predictions are not discarded globally. Minor syntax errors are repaired deterministically when the intended field structure is recoverable. If a prediction remains unparsable, only the affected fields are treated as empty. This localizes a syntax failure to the fields it prevents from being evaluated and avoids giving a model credit for content that cannot be reliably associated with an element object. To ensure reproducibility and provide a clear overview of our experimental setup, the primary

Table 13: Hyperparameters and implementation details for model fine-tuning.

Hyperparameter / Setting	**Value**
Model & Strategy	
Base Model	Qwen3.5-2B
Training Strategy	Full Fine-Tuning
Trainable Modules	Aligner, LLM
Frozen Modules	ViT (`freeze_vit=True`)
Precision	bfloat16
Max Sequence Length	8192
Optimization	
Training Epochs	1
Peak Learning Rate	2e-5
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$)
LR Scheduler	Cosine Decay
Batch Size per Device	2
Gradient Accumulation Steps	4
Effective Batch Size	$8 \times$ Number of GPUs
Data & Environment	
Data Augmentation	Dynamic Cropping
Hardware	$1 \times$ NVIDIA A100 (40GB)
Peak VRAM per GPU	$\sim$38.5 GB
Total Training Time	$\sim$20–24 Hours

training hyperparameters, optimization configurations, and computing resources for model fine-tuning are summarized in Table 13. Unless specified otherwise, other baseline models and our proposed adapter follow comparable training schedules and hardware environments for fair comparison.

E Training-data Scaling

We compare full fine-tuning and SARA on Qwen3.5-2B using 10%, 30%, 50%, and 100% of the training data under the same reported evaluation protocol. Table 14 reports all six metrics. At 10%, SARA trails full fine-tuning on all four structural metrics. Results are mixed at 30%; at 50%, it improves F1, mIoU, and OCR while matching RO, and at 100% it improves all four. The F1 difference changes from -0.0756 at 10% to $+0.1790$ at 50% and $+0.1853$ at 100%. Thus, the structural advantage emerges at larger training fractions in this experiment. METEOR favors the adapter at each fraction, while BERTScore changes are mixed.

F Resolution Sensitivity

We evaluate Qwen3.5-2B+SARA at 100%, 50%, and 25% of the reported native crop resolution while keeping the model and evaluation settings fixed. As shown in Table 15, reducing resolution to

Table 14: Training-data scaling on Qwen3.5-2B. Fractions indicate the reported proportion of the training data.

Data	Method	F1	mIoU	OCR	RO	METEOR	BERTScore
10%	Full FT	0.6487	0.7563	0.8476	0.7996	0.1256	0.8106
10%	SARA	0.5731	0.7144	0.7311	0.7805	0.1672	0.8002
30%	Full FT	0.6771	0.7861	0.8756	0.8504	0.2651	0.8288
30%	SARA	0.6567	0.8061	0.8087	0.8226	0.2871	0.8372
50%	Full FT	0.6991	0.8232	0.9362	0.9623	0.3072	0.8540
50%	SARA	0.8781	0.8533	0.9562	0.9623	0.3340	0.8533
100%	Full FT	0.7336	0.8350	0.9408	0.9628	0.3741	0.8928
100%	SARA	0.9189	0.8919	0.9636	0.9774	0.3834	0.8946

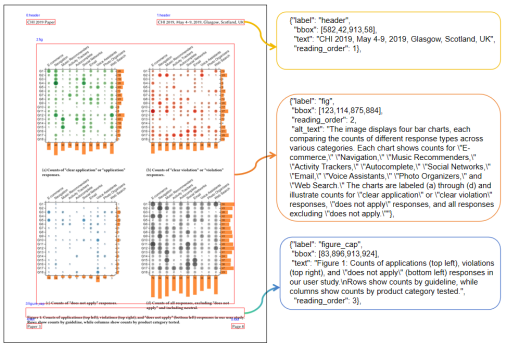
Table 15: Resolution sensitivity of Qwen3.5-2B+SARA under the reported crop-resolution.

Resolution	F1	mIoU	OCR	RO
100%	0.9189	0.8919	0.9636	0.9774
50%	0.8389	0.8673	0.9248	0.9534
25%	0.7589	0.8194	0.7648	0.8864

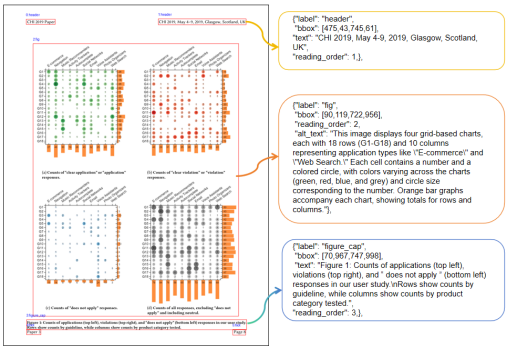
25% decreases F1 by 0.1600, mIoU by 0.0725, OCR by 0.1988, and RO by 0.0910. Because mIoU is averaged over matched true positives, its remaining value describes overlap among surviving matches and does not indicate preservation of all element boundaries. This experiment measures absolute input-resolution sensitivity; it does not establish a robustness advantage over full fine-tuning without a corresponding control evaluated at the same resolutions.

G Failure cases.

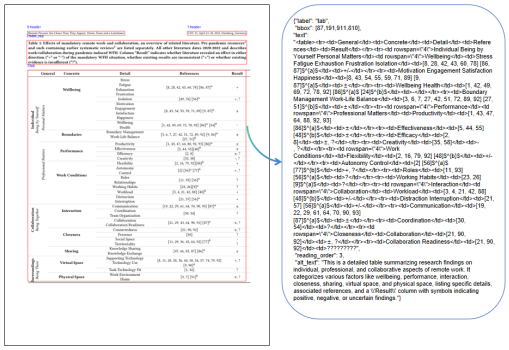
To better understand the limitations of SARA, we analyze two representative failure modes, shown in Figure 5. First, borderless composite figures can confuse the model when multiple sub-figures and intra-image text are densely arranged. In such cases, visual and textual cues become entangled, which may shift predicted figure boxes away from their corresponding captions. Second, excessively dense tables can stress the decoding process. Although the table region may still be localized accurately, long serialized outputs may be truncated near the end, and the model may occasionally introduce minor irrelevant artifacts.



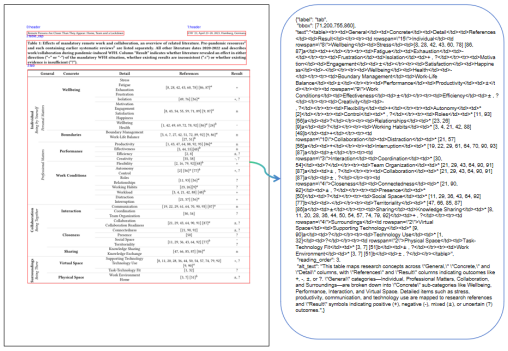
(a) Prediction: borderless composite figure



(b) Ground truth for Case 1



(c) Prediction: dense table



(d) Ground truth for Case 2

Figure 5: Representative failure cases of SARA. Top row: a borderless composite figure where intra-image text can cause a spatial shift between figure regions and their captions. Bottom row: a dense table where localization remains relatively accurate, but decoding may truncate long content and introduce minor hallucinated artifacts.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the benchmark, schema, dataset scale, evaluation axes, SARA adapter, and empirical scope. The limitations section states the main caveats.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 7 discusses source-domain coverage, first-eight-page curation, LLM-assisted annotation limits, single-run reporting, and possible public-web pre-training exposure.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: [N/A]

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies the output schema, train/validation/test split policy, metrics, parsing failure handling, deterministic decoding policy when supported, full fine-tuning baselines, and the primary frozen-backbone SARA training configuration. Anonymous code, data documentation, and evaluation scripts are planned for release.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The evaluation code would be provided through the OpenReview platform.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 6 describes baseline families, zero-shot and fine-tuned settings, output conversion, failure handling, and Table 13 gives the primary fine-tuning hyperparameters.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes].

   Justification: The current results are single-run measurements from the available experiments. This is explicitly acknowledged as a limitation.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Table 13 reports GPU type, memory, peak VRAM, batch configuration, and training time for the primary fine-tuning experiment. Released baselines are evaluated with official checkpoints or APIs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses open-access documents, preserves anonymity in the submission, reports limitations, and documents LLM-assisted annotation and release assumptions.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Section 7 discusses accessibility benefits and risks including hallucinated descriptions, privacy concerns, and overreliance on generated metadata.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: The release is planned around open-access sources, provenance preservation, schema documentation, and evaluation scripts rather than unrestricted deployment claims. Generated alt-text is presented as benchmark annotation requiring validation.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites existing datasets, models, metrics, and systems. The dataset construction section states that documents are collected from open-access PDFs; release documentation should include source and license metadata.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Read-Parse-Describe is a new benchmark asset. The paper describes its schema, construction, deduplication, split policy, annotation process, limitations, and planned anonymized documentation.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The benchmark uses Gemini 3 Pro for initial alt-text generation and Gemini 3 Flash for LLM-as-a-judge scoring, followed by human refinement for low-scoring samples. This usage is described in Section 3.4 and Appendix B.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.