Infinity-Parser: Layout-Aware Reinforcement Learning with High-quality Document Parsing Dataset

Baode Wang[1*] Biao Wu[2*] Weizhen Li[1*] Meng Fang[3*] Zuming Huang[1†]
Jun Huang[1] Yanjie Liang[1] Haozhe Wang[1] Ling Chen[2] Wei Chu[1] Yuan Qi[1§]

[1] INFLY Tech [2] Australian Artificial Intelligence Institute [3] University of Liverpool
[†] Project Lead [§] Corresponding Author

Abstract

Document parsing from scanned images into structured formats remains a significant challenge due to its complexly intertwined elements such as text paragraphs, figures, formulas, and tables. The lack of high-quality training data for layout-aware document parsing fundamentally limits models' ability to learn effective layout analysis. At the same time, existing supervised fine-tuning methods fail to generalize across diverse document types and perform poorly on out-of-distribution data. To address these challenges, we build Infinity-Doc-400K, a large-scale dataset that reduces the lack of high-quality training data for layout-aware parsing. Building on this dataset, we propose Layout-Aware RL, a reinforcement learning framework that improves layout analysis using composite rewards based on normalized edit distance, paragraph count accuracy, and reading order preservation. Using this framework, we train Infinity-Parser, a vision-language model that generalizes well across diverse domains. Extensive evaluations on benchmarks including OmniDocBench, olmOCR-Bench, PubTabNet, and FinTabNet show that Infinity-Parser consistently achieves state-of-the-art performance across a broad range of document types, languages, and structural complexities, substantially outperforming both specialized document parsing systems and general-purpose vision-language models. We will release our code, dataset, and model to facilitate reproducible research in document parsing.

1 Introduction

Document parsing aims to convert scanned documents into structured, machine-readable formats and represents one of the core tasks in document intelligence (Hwang et al., 2021; Wang et al., 2024b; Wei et al., 2024; Xia et al., 2024; Zhang et al., 2024). As shown in Figure 1, unlike traditional OCR that focuses solely on text recognition, document parsing requires comprehensive recovery

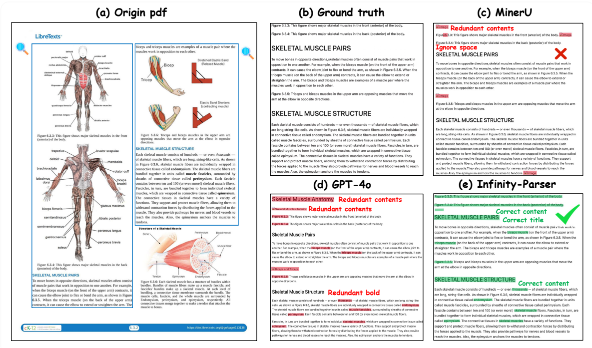

Figure 1: Comparison of Markdown extraction from a colorful textbook-style PDF.

of hierarchical document structures, including the dependency relationships among elements such as paragraphs, headers, tables, and formulas—a capability that is crucial for downstream applications including legal contract analysis, scientific literature mining, and financial report processing (Poznanski et al., 2025). Traditional approaches typically rely on multi-stage pipelines that decompose document parsing into several supervised subtasks, such as layout detection, OCR, table recognition, and formula recognition. These components are then combined through heuristic post-processing to reconstruct the document structure. structure (Blecher et al., 2024; Wei et al., 2025a; Liu et al., 2024; Wei et al., 2024; Bai et al., 2024; Chen et al., 2024). However, such pipeline-based methods are prone to error propagation and exhibit limited adaptability when confronted with diverse layout variations (Wang et al., 2024b).

Recently, document parsing has increasingly been reformulated as an end-to-end perception task using vision-language models (VLMs) trained via supervised fine-tuning (SFT) (Wei et al., 2025b; Li et al., 2025). However, the lack of large-scale, high-quality training data has significantly limited models' ability to learn layout-aware representations. Existing datasets either focus on text recognition tasks such as OCR (Hu et al., 2024a) or sup-

Findings of the Association for Computational Linguistics: ACL 2026, pages 1647–1667

Datasets	Annotation Type					End-to-End Task				Layout Analysis	Exact Match
	BBox	Text	Table	Formula	Attr.	OCR	TR	MFR	ROD		
End-to-End Eval Benchmarks											
Fox (Liu et al., 2024)	✓	✓				✓					
Nougat (Blecher et al., 2024)		✓	✓	✓		✓	✓	✓			
GOT OCR 2.0 (Wei et al., 2024)		✓	✓	✓		✓	✓	✓		✓	
OmniDocBench (Ouyang et al., 2024)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
End-to-End Train Dataset											
DocStruct4M (Hu et al., 2024a)		✓				✓					
olmOCR-mix (Poznanski et al., 2025)		✓	✓	✓		✓	✓	✓	✓	✓	
Infinity-Doc-400K	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Comparison between Infinity-Doc-400K and existing datasets.

port layout analysis without strict character-level alignment (Poznanski et al., 2025), making them insufficient for end-to-end document parsing.

To address this data bottleneck, we construct Infinity-Doc-400K, a large-scale document parsing dataset that provides high-quality, scalable supervision. It integrates two complementary data sources: high-fidelity synthetic scanned document parsing data generated via HTML templates and browser rendering, ensuring precise character-level alignment and structural consistency; and real-world document samples processed through a cross-model consistency–based pseudo-labeling pipeline with expert-guided rule refinement. By combining these two sources, Infinity-Doc-400K achieves a favorable balance between dataset scale, annotation quality, and layout diversity.

Beyond data limitations, SFT as a training paradigm still exhibits inherent limitations, even with high-quality training data. Although SFT provides token-level supervision, models tend to overfit surface patterns rather than learning underlying principles with strong generalization capability (Chu et al., 2025). Recent studies have shown that reinforcement learning (RL) exhibits notable advantages in vision and multimodal tasks, where outcome-based reward signals guide models to learn transferable representations and thus improve generalization (Huang et al., 2025; Liu et al., 2025a; Wang et al., 2025b). Motivated by this observation, we introduce RL into document parsing. This process does not rely on explicit reasoning; instead, it treats the entire document parsing result as the final prediction and guides model learning through carefully designed reward signals.

To systematically evaluate our approach, we conduct extensive experiments across multiple document parsing benchmarks and downstream tasks. Results show that Infinity-Doc-400K enables consistent performance gains under SFT across different model backbones, validating the robustness of its annotation quality. Under the same data conditions, RL consistently outperforms SFT, demonstrating greater stability across subtasks, stronger robustness to complex layouts, and smoother training convergence, while achieving clear improvements in both out-of-distribution (OOD) scenarios and overall document parsing metrics. Leveraging the synergy between high-quality data and the RL-based training paradigm, our method, Infinity-Parser, achieves new SOTA results on OmniDocBench (Ouyang et al., 2024), olmOCR (Poznanski et al., 2025), PubTabNet (Zhong et al., 2020a), and FinTabNet (Zheng et al., 2021).

We make the following contributions:

- We introduce Infinity-Doc-400K, a large-scale dataset of 400,066 scanned documents that combines high-quality synthetic data with diverse real-world samples.
- We reveal, through extensive experiments, the limitations of SFT for document parsing and show that RL effectively alleviates these issues on both layout analysis and OOD tasks.
- We train a VLM based model, Infinity-Parser, which sets new SOTA performance across English and Chinese benchmarks for OCR, table and formula extraction, and reading-order detection.

2 Related Work

2.1 Reinforcement Learning for LLM

Recent advancements in Large Language Models (LLMs) such as OpenAI's GPT series (OpenAI, 2024)), DeepSeek-R1 (Guo et al., 2025), and Gemini (Team et al., 2023) have highlighted the significant potential of Reinforcement Learning (RL) in enhancing their reasoning capabilities. This RL paradigm has been successfully extended to other domains demanding sophisticated reasoning, including code generation (Li et al., 2022b; Zeng et al., 2025), autonomous tool utilization (Schick et al., 2023; Wang et al., 2025a), and information retrieval (Nakano et al., 2021). Similarly, RL has demonstrated its efficacy in the domain of Visual Language Models (VLMs), including precise object counting (Peng et al., 2025), nuanced visual perception (Liu et al., 2025b), and complex multimodal reasoning (e.g., VL-Rethinker (Wang et al., 2025c), Pixel Reasoner (Su et al., 2025), Vision-R1 (Huang et al., 2025)). These pioneering works have predominantly relied on binary outcome rewards to guide RL training. Complementary to these efforts, our work demonstrates the effectiveness of incorporating layout-aware and layout-based rewards for document parsing, offering a more granular and contextually relevant feedback mechanism.

2.2 VLM-based Document Parsing

Recent advancements in document understanding and optical character recognition (OCR) have highlighted their importance as critical benchmarks for evaluating the perceptual capabilities of vision-language models (VLMs). By incorporating large-scale OCR corpora during pretraining, models such as GPT-4o (Achiam et al., 2023) and Qwen2-VL (Bai et al., 2024) have achieved competitive performance on document content extraction tasks. Building upon these foundations, the emergence of VLMs has further accelerated the progress of end-to-end document parsing, giving rise to a range of models such as Donut (Blecher et al., 2024), Nougat (Blecher et al., 2023), Kosmos-2.5 (Lv et al., 2024), Vary (Wei et al., 2025a), mPLUG-DocOwl (Hu et al., 2024b), Fox (Liu et al., 2024), and GOT (Wei et al., 2024). These models have continued to improve their understanding of visual layouts and textual content by leveraging advancements in visual encoders (Dosovitskiy et al., 2020), language decoders (Bai et al., 2024), and data construction pipelines. Despite the success of these VLM-based approaches in enabling end-to-end document parsing, they still face generalization challenges on downstream layout parsing tasks (Wang et al., 2024b). To address this issue, we propose leveraging reinforcement learning to provide a more effective training paradigm that better aligns with the demands of document parsing.

3 Infinity-Doc-400K

We introduce Infinity-Doc-400K, a large-scale, multimodal dataset of 400,066 richly annotated documents for end-to-end scanned document parsing. Unlike prior benchmarks that target isolated subtasks (e.g., layout detection, OCR, or table recognition), Infinity-Doc-400K provides holistic supervision by pairing rendered scanned document pages with their ground-truth Markdown representations. This design enables training and evaluating models that directly translate visual inputs to layout outputs without relying on brittle, multi-stage pipelines. As shown in Table 1, compared to existing works, Infinity-Doc-400K not only significantly enhances task diversity but also substantially improves overall data quality through our proposed synthetic generation mechanism. More details on the data distribution and quality control are provided in the Data Details section of the Appendix.

To construct Infinity-Doc-400K, we design a dual-pipeline framework that integrates both synthetic and real-world document generation, as illustrated in Figure 2. This design addresses a critical limitation of traditional data construction pipelines, which often rely on weak supervision and pseudo-labeling from a single model applied to crawled, scanned documents. These pipelines frequently suffer from noisy, misaligned, or incomplete annotations, especially in complex layouts or multilingual content, thus hindering model performance and generalization. To overcome these issues, our dual-pipeline framework is motivated by the need to balance annotation quality and structural diversity. The synthetic branch provides highly accurate, clean, and precisely aligned annotations at scale, while the real-world branch introduces naturally occurring layout variability and semantic richness, which are essential for building models that generalize robustly in practical applications.

Real-World Data We develop a real-world data construction pipeline to capture the structural complexity and natural layout variability of documents across practical domains. We collect diverse scanned documents from sources such as financial reports, medical records, academic papers, books, magazines, and web pages, covering both dense and sparse content layouts. To generate structural annotations, we adopt a multi-expert strategy, where specialized models handle different structural elements, such as layout blocks, texts, formulas, and tables. For example, overall lay-

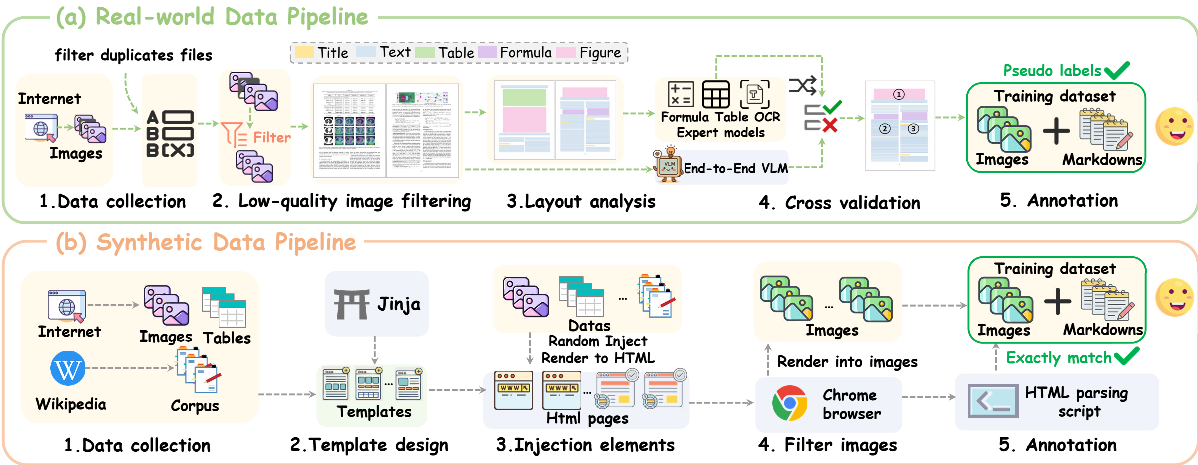

Figure 2: Data construction pipelines for document parsing. (a) Real-world pipelines enhance quality by combining multiple expert models and layout analysis, yielding better-aligned supervision through intersection and reading order reasoning. (b) Synthetic pipeline leverages structured HTML templates and browser rendering to generate clean, exactly-aligned scanned document parsing data, ensuring high-quality supervision for end-to-end parsing.

outs are analyzed by a visual layout model (Huang et al., 2022), formula regions are processed by a dedicated formula recognition model (Wang et al., 2024a), and tables are parsed by a transformer-based table extractor (Blecher et al., 2024). A cross-validation mechanism is then applied to filter out inconsistencies by comparing the outputs of expert models and VLMs. Only regions with consistent predictions across models are retained as high-confidence pseudo-ground-truth annotations. This layout-aware filtering results in a rich and reliable dataset that reflects the complexity of real-world documents and supports robust document parsing model training.

Synthetic Data We design a synthetic data construction pipeline. We collect text and images from sources such as Wikipedia, web crawlers, and online corpora, and use Jinja (Nipkow, 2003) templates to inject sampled content into predefined single-, double-, or triple-column HTML layouts. These pages are rendered into scanned documents using a browser engine, followed by automated filtering to remove low-quality or overlapping images. Ground-truth annotations are extracted by parsing the original HTML to produce aligned Markdown representations. This synthetic approach not only significantly reduces construction costs and ensures annotation accuracy and structural diversity, but more importantly, it addresses the longstanding issue of imprecise or inconsistent supervision commonly found in pseudo-labeled datasets, providing high-quality and well-aligned supervision for training end-to-end models.

Quality Control Measures To ensure reliable annotations at scale, we designed a hybrid quality control strategy for Infinity-Doc-400K. As shown in Figure 3, first, three domain experts with doctoral degrees in document analysis manually inspected about 5% of the data. Their feedback not only identified potential errors but also served as a quality anchor for evaluating annotation consistency. Guided by these inspections, we iteratively refined the screening rules no fewer than five times, continuously improving the labeling pipeline. Finally, to achieve scalability, we employed a model-based cross-verification mechanism in which multiple models generate annotations for real-world samples. Outputs with high consistency are retained, while inconsistent cases are fed back for further refinement of the rule set. This layered framework is anchored by expert inspection, strengthened through iterative rule optimization, and scaled via model cross-checking. Together, these components effectively balance annotation reliability and dataset scalability.

4 Layout-Aware Reinforcement Learning

As illustrated in Figure 4, we employ an RL framework to directly optimize scanned document parsers, aiming to enhance both structural fidelity and semantic accuracy. Specifically, we utilize GRPO (Shao et al., 2024), which enables learning from rule-based reward signals without relying on absolute values. GRPO operates by generating a set of candidate Markdown outputs for each document and evaluating them using a *multi-aspect*

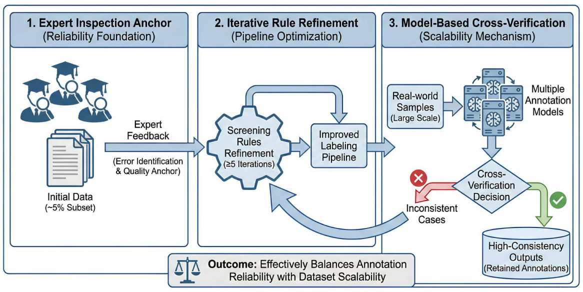

Figure 3: Overview of a three-stage annotation framework combining expert inspection, iterative rule refinement, and model-based cross-verification.

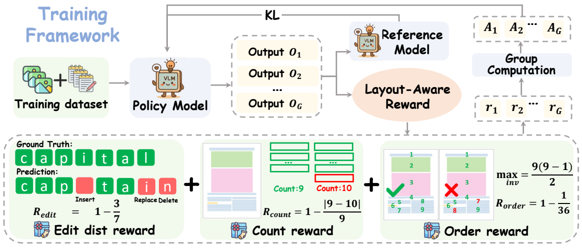

Figure 4: Overview of Infinity-Parser training framework. Our model is optimized via reinforcement finetuning with edit distance, layout, and order-based rewards.

reward, denoted as $R_{\text{Multi-Aspect}}$, which integrates multiple rule-based criteria into a unified supervisory signal. These raw rewards are then converted into relative advantage scores by comparing each candidate against others within the same group. This relative evaluation promotes training stability and encourages the selection of higher-quality outputs, eliminating the need for a learned value function or critic. Notably, our reinforcement learning approach avoids any explicit thinking or intermediate reasoning process; instead, all outputs are treated as final answers, with the model receiving verifiable rewards based on these outputs. The multi-aspect reward $R_{\text{Multi-Aspect}}$ consists of three complementary components, each capturing a different aspect of parsing quality:

Edit Distance Reward (R_{dist}) We define the edit distance reward based on the normalized Levenshtein distance $D(y, \hat{y})$ between the predicted output $\hat{y}$ and the reference output y. Let $N = |y|$ and $M = |\hat{y}|$ denote the lengths of the reference and predicted sequences, respectively. The reward is computed as $R_{\text{dist}} = 1 - D(y, \hat{y}) / \max(N, M)$, where $D(y, \hat{y})$ measures the minimum number of single-character insertions, deletions, or substitutions required to transform $\hat{y}$ into y, thereby capturing both semantic and formatting discrepancies. This reward is bounded within $[0, 1]$, with higher values indicating better alignment between the prediction and the reference. The reference output y is synthesized through two data generation pipelines proposed in this work, using rigorous rule-based filtering and consistency validation, and serves as a high-quality surrogate for ground-truth annotations when evaluating model outputs.

Count Reward (R_{count}) To encourage accurate paragraph segmentation, let N_Y and $N_{\hat{Y}}$ denote the numbers of reference and predicted paragraphs, respectively. The count reward is defined as $R_{\text{count}} = 1 - \frac{|N_Y - N_{\hat{Y}}|}{N_Y}$, which penalizes missing or spurious paragraphs.

Order Reward (R_{order}) We measure sequence-level fidelity by counting the number of pairwise inversions D_{order} between the reference and predicted paragraphs. Let $\max_{\text{inv}} = N_Y(N_Y - 1)/2$ denote the maximum possible number of inversions. The order reward is defined as $R_{\text{order}} = 1 - D_{\text{order}} / \max_{\text{inv}}$, which rewards the preservation of the original reading order.

Multi-aspect Reward The final multi-aspect reward is a weighted combination of these three components. Specifically, we begin by applying the Hungarian algorithm (Kuhn, 1955) to establish the optimal one-to-one matching between predicted and ground-truth segments, identifying both pairings and their relative order. Based on the matched segment count, we compute the count reward to reflect alignment in the number of segments. Using the relative sequence of matched pairs, we calculate the order reward to measure structural consistency. On top of these matchings, we compute the edit reward by averaging the edit similarities of each matched segment pair. Combining these terms yields the final reward:

$$R_{\text{Multi-Aspect}} = R_{\text{dist}} + R_{\text{count}} + R_{\text{order}} \quad (1)$$

This multi-aspect design balances content fidelity with structural correctness and order preservation, providing rich supervision for end-to-end document parsing.

5 Experiments

Implementation Details We fine-tune the Qwen2.5-VL-7B (Bai et al., 2025b) model using GRPO within a distributed training setup based on Verl (Sheng et al., 2024; Yaowei Zheng, 2025),

Methods	Overall$^{\text{Edit}}\downarrow$		Text$^{\text{Edit}}\downarrow$		Form.$^{\text{Edit}}\downarrow$		Table$^{\text{TEDS}}\uparrow$		Table$^{\text{Edit}}\downarrow$		Read Order$^{\text{Edit}}\downarrow$	
	EN	*ZH*	*EN*	*ZH*	*EN*	*ZH*	*EN*	*ZH*	*EN*	*ZH*	*EN*	*ZH*
Based on Pipeline Tools												
MinerU (Wang et al., 2024b)	0.15	0.357	0.061	0.215	0.278	0.577	78.6	62.1	0.18	0.344	0.079	0.292
Marker (Paruchuri, 2024)	0.336	0.556	0.080	0.315	0.530	0.883	67.6	49.2	0.619	0.685	0.114	0.340
Mathpix	0.191	0.365	0.105	0.384	0.306	0.454	77.0	67.1	0.243	0.320	0.108	0.304
Docling (Livathinos et al., 2025)	0.589	0.909	0.416	0.987	0.999	1.000	61.3	25.0	0.627	0.810	0.313	0.837
Pix2Text (Gurgurov and Morshnev, 2024)	0.320	0.528	0.138	0.356	0.276	0.611	73.6	66.2	0.584	0.645	0.281	0.499
Unstructured-0.17.2	0.586	0.716	0.198	0.481	0.999	1.000	-	-	1.000	0.998	0.145	0.387
OpenParse-0.7.0	0.646	0.814	0.681	0.974	0.996	1.000	64.8	27.5	0.284	0.639	0.595	0.641
Based on Expert VLMs												
GOT-OCR (Wei et al., 2024)	0.287	0.411	0.189	0.315	0.360	0.528	53.2	47.2	0.459	0.520	0.141	0.280
Nougat (Blecher et al., 2024)	0.452	0.973	0.365	0.998	0.488	0.941	39.9	0.0	0.572	1.000	0.382	0.954
Mistral OCR	0.268	0.439	0.072	0.325	0.318	0.495	75.8	63.6	0.600	0.650	0.083	0.284
OLMOCR-sglang	0.326	0.469	0.097	0.293	0.455	0.655	68.1	61.3	0.608	0.652	0.145	0.277
SmolDocling-256M	0.493	0.816	0.262	0.838	0.753	0.997	44.9	16.5	0.729	0.907	0.227	0.522
Based on General VLMs												
GPT-4o (Achiam et al., 2023)	0.233	0.399	0.144	0.409	0.425	0.606	72.0	62.9	0.234	0.329	0.128	0.251
Qwen2-VL-72B (Wang et al., 2024c)	0.252	0.327	0.096	0.218	0.404	0.487	76.8	76.4	0.387	0.408	0.119	0.193
InternVL2-76B (Chen et al., 2024)	0.440	0.443	0.353	0.290	0.543	0.701	63.0	60.2	0.547	0.555	0.317	0.228
Qwen2.5-VL-7B (Bai et al., 2025a)	0.220	0.265	0.142	0.205	0.393	0.530	78.7	78.3	0.155	0.162	0.191	0.169
InternVL3-8B (Zhu et al., 2025)	0.426	0.385	0.315	0.345	0.714	0.729	59.0	71.5	0.352	0.211	0.324	0.257
Based on Reinforcement Learning												
Infinity-Parser-7B	**0.122**	**0.174**	**0.053**	**0.079**	**0.250**	**0.366**	**83.3**	**80.9**	**0.122**	**0.156**	**0.062**	**0.095**

Table 2: Comprehensive evaluation of document parsing algorithms on OmniDocBench: performance metrics for text, formula, table, and reading order extraction, with overall scores derived from ground truth comparisons.

utilizing 8 A100 GPUs (80GB). Throughout our experiments, we set the KL coefficient $\beta = 1.0 \times 10^{-2}$. And for each problem instance, we sample 8 responses, each with a maximum length of 8192 tokens and a temperature of 1.0. Both the rollout batch size and the global batch size are set to 128. The actor model is updated using the AdamW optimizer with parameters $(\beta_1 = 0.9, \beta_2 = 0.99)$ and a learning rate 1.0×10^{-6}. The model is trained for 1.0 epoch for all experiments. Due to limited computational resources, we randomly sampled 43K documents from the 400K corpus for training. In our main results, we directly performed RL on the base model using the 43K subset.

Experiment Setup To evaluate layout analysis capability, we conduct experiments *OmniDocBench* (Ouyang et al., 2024), *PubTabNet* (Zhong et al., 2020b), and *FinTabNet* (Zheng et al., 2021), which emphasize document structure understanding, including reading order and layout organization. In addition, to assess OCR performance at the character level, we evaluate our method on *olmOCR-Bench* (Poznanski et al., 2025) and the *Overall_Cat* task of *OmniDocBench* (Ouyang et al., 2024), both of which focus on token- and character-level text matching independent of structural recovery. Detailed descriptions of these benchmarks are provided in Appendix.

5.1 Main Results

Overall Evaluation on OmniDocBench As shown in Table 2, pipeline-based methods such as MinerU (Wang et al., 2024b) and Mathpix achieve superior performance across individual sub-tasks including text recognition and formula recognition. Meanwhile, general-purpose vision-language models like Qwen2.5-VL-7B and GPT-4o also demonstrate competitive results. Notably, most methods perform better on English pages compared to Chinese pages, reflecting language-dependent challenges. In contrast, our proposed Infinity-Parser-7B achieves a more balanced performance across all sub-tasks and languages, setting new SOTA results with overall edit distances of 0.122 and 0.174. This highlights the advantage of RL with multi-aspect rewards in enabling robust, end-to-end document parsing.

Document-level OCR Evaluation Table 3 reports performance on the olmOCR-Bench benchmark, which evaluates document-level OCR across diverse layouts and domains. Infinity-Parser-7B achieves the highest overall score (82.5), followed closely by olmOCR (Anchored), both demonstrating strong performance in complex categories like multi-column layouts and scanned math content.

Table Recognition Evaluation To evaluate the model's generalization ability, we introduce task-specific test cases. In Table 4, we compare Infinity-Parser-7B with end-to-end table recognition mod-

Model	Overall	ArXiv	Old Scans Math	Tables	Old Scans	Headers&Footers	Multi Col.	Long-Tiny Text	Base
GOT OCR	48.3	52.7	52.0	0.2	22.1	93.6	42.0	29.9	94.0
Marker v1.6.2	59.4	24.3	22.1	69.8	24.3	87.1	71.0	76.9	99.5
MinerU v1.3.10	61.5	75.4	47.4	60.9	17.3	**96.6**	59.0	39.1	96.6
Mistral OCR API	72.0	77.2	67.5	60.6	29.3	93.6	71.3	77.1	99.4
GPT-4o (No Anchor)	68.9	51.5	75.5	69.1	40.9	94.2	68.9	54.1	96.7
GPT-4o (Anchored)	69.9	53.5	74.5	70.0	40.7	93.8	69.3	60.6	96.8
Gemini Flash 2 (No Anchor)	57.8	32.1	56.3	61.4	27.8	48.0	58.7	84.4	94.0
Gemini Flash 2 (Anchored)	63.8	54.5	56.1	72.1	34.2	64.7	61.5	71.5	95.6
Qwen 2 VL (No Anchor)	31.5	19.7	31.7	24.2	17.1	88.9	8.3	6.8	55.5
Qwen 2.5 VL (No Anchor)	65.5	63.1	65.7	67.3	38.6	73.6	68.3	49.1	98.3
olmOCR v0.1.68 (No Anchor)	76.3	72.1	74.7	71.5	43.7	91.6	78.5	80.5	98.1
olmOCR v0.1.68 (Anchored)	77.4	75.6	75.1	70.2	44.5	93.4	79.4	81.7	99.0
Infinity-Parser-7B	**82.5**	**84.4**	**83.8**	**85.0**	**47.9**	88.7	**84.2**	**86.4**	**99.8**

Table 3: Performance comparison on the olmOCR (Poznanski et al., 2025) benchmark across multiple document domains and structural challenges. Higher is better.

Model	PubTabNet		FinTabNet	
	TEDS-S	TEDS	TEDS-S	TEDS
EDD	89.9	88.3	90.6	-
OmniParser	90.45	88.83	91.55	89.75
InternVL3-8B	87.48	83.02	86.73	84.01
InternVL3-78B	89.63	82.11	92.51	89.21
Qwen2.5-VL-7B	86.78	81.60	87.46	82.58
Qwen2.5-VL-72B	87.91	84.39	87.13	82.90
GPT-4o	86.16	76.53	87.00	83.96
Infinity-Parser-7B	**93.46**	**91.82**	**97.16**	**95.92**

Table 4: Comparisons of end-to-end table recognition methods on PubTabNet and FinTabNet.

Model	Method	Data Size	OA-EN	OA-ZH	OA-Cat
Qwen-2.5-VL-7B	Zero Shot	–	0.220	0.265	0.183
Qwen-2.5-VL-7B	SFT	400K	0.154	0.240	0.122
InternVL-3.5-8B	Zero Shot	–	0.232	0.359	0.201
InternVL-3.5-8B	SFT	400K	0.171	0.286	0.162
Llama-3.2-11B-Vision	Zero Shot	–	0.448	0.760	0.640
Llama-3.2-11B-Vision	SFT	400K	0.216	0.392	0.246

Table 5: Comparison of different vision-language models under zero-shot and SFT settings. Lower is better.

els on PubTabNet and FinTabNet using the TEDS metric, which evaluates both structure and content. We also report TEDS-S for structure-only assessment. The evaluation results for InternVL3, Qwen2.5-VL, and GPT-4o were generated through our standardized benchmarking pipeline. Infinity-Parser-7B achieves the highest TEDS-S and TEDS scores on both datasets.

5.2 Ablation Study

We perform ablation experiments to evaluate the individual contributions of our three core design choices: (1) data quality verification and (2) layout-aware RL. We report all ablation results using two primary evaluation metrics: Overall$^{\text{Edit}}$ (OA-ZH or OA-EN) and Overall$^{\text{Cat.}}$ (OA-Cat). Overall$^{\text{Edit}}$ represents the average edit-based overall score across English and Chinese pages, as shown in Table 2. In contrast, Overall$^{\text{Cat.}}$ reflects the mean category-level performance across nine types of scanned document pages, following the same evaluation setting as Table 9 in the Appendix.

Data Validity Table 5 compares several vision-language models (Bai et al., 2025b; Zhu et al., 2025; Meta, 2024) under zero-shot and SFT settings. Overall, all models exhibit consistent error reductions across English, Chinese, and category-level evaluations after SFT. These results indicate that high-quality training data is inherently effective and can substantially improve document parsing performance across different model architectures and scales.

Layout-Aware RL Table 6 presents an ablation study analyzing the effects of SFT and different RL reward components. Starting from the zero-shot baseline, SFT alone yields consistent performance improvements as the data scale increases from 43K to 400K, thereby confirming the high quality and effectiveness of the Infinity-Doc-400K dataset. However, RL demonstrates a stronger and more targeted impact on both structural and category-level metrics. Even without any SFT initialization, applying RL with an edit-distance reward already significantly improves performance over zero-shot training. Introducing additional count and order rewards further enhances structural consistency, leading to substantial reductions across all metrics, with the full multi-aspect reward achieving the best results among RL-only settings. Notably, even when initialized from the strongest 400K SFT baseline, RL continues to deliver clear improvements. However, large-scale SFT may introduce token-level inductive biases that restrict exploration, which helps explain why RL-only training can be surprisingly effective when starting from a strong pretrained VLM.

Furthermore, we investigate whether RL can be effectively trained using synthetic data alone. As shown in Table 6, Synthetic + RL (43K) achieves OA-EN 0.182 / OA-ZH 0.224, and Synthetic + RL (69K) achieves OA-EN 0.166 / OA-ZH 0.198, both of which are substantially worse than RL trained on the combined dataset (0.122 / 0.174). These results suggest that while RL is effective for structure-level optimization, its success critically depends on sufficient structural diversity in the training data. Synthetic-only data lacks the realistic layout variability and noise patterns present in real-world documents, thereby limiting the effectiveness of reward-driven optimization.

Train Data	SFT	RL	Edit Dist.	Count.	Order.	OA-EN	OA-ZH
-	-	-	-	-	-	0.220	0.265
Combined	43K	-	-	-	-	0.198	0.261
Combined	400K	-	-	-	-	0.154	0.240
Combined	-	43K	✓	-	-	0.169	0.224
Combined	-	43K	✓	✓	-	0.159	0.200
Combined	-	43K	✓	✓	✓	0.122	0.174
Combined	43K	43K	✓	✓	✓	0.163	0.195
Combined	400K	43K	✓	✓	✓	0.151	0.206
Synthetic	-	43K	✓	✓	✓	0.182	0.224
Synthetic	-	69K	✓	✓	✓	0.166	0.198

Table 6: Ablation study on different training settings. Lower is better.

Data Type Table 7 shows that, at the same data scale, using synthetic data alone achieves performance comparable to, or slightly better than, real data, indicating the high quality of the synthetic annotations. This advantage mainly stems from the fact that synthetic data provides fully accurate labels. However, compared to real-world data, synthetic data exhibits more homogeneous layout patterns and limited diversity, making it insufficient to fully represent real document distributions. Therefore, synthetic data is primarily used to complement real data by covering complex layout structures that are difficult to collect at scale in real-world settings, rather than replacing real data.

Method	Data Type	Data Size	Overall^{Edit} ↓	Overall$^{Cat.}$ ↓
SFT	Real	5k	0.337	0.279
SFT	Synth	5k	0.290	0.214
RL	Real	5k	0.285	0.164
RL	Synth	5k	0.262	0.156
RL	Synth+Real	2.5k+2.5k	0.273	0.155

Table 7: Results with different data types.

5.3 Further Analysis of LayoutRL

To gain deeper insights into the behavior of LayoutRL, we conduct further analyses across multiple document parsing benchmarks. We evaluate models using two complementary metrics: paragraph-level accuracy, which primarily reflects character-level matching performance and thus correlates closely with OCR quality, and page-level accuracy, which emphasizes holistic layout understanding and the correct organization of document structures. As shown in Figure 5 (left), while SFT achieves strong performance on paragraph-level accuracy, its page-level performance improves only marginally as the training data scale increases. This suggests that SFT tends to focus on surface-level pattern matching rather than effectively modeling the hierarchical dependencies among document structures and elements. In contrast, our layout-aware RL approach significantly outperforms SFT on page-level accuracy, while maintaining competitive paragraph-level performance.

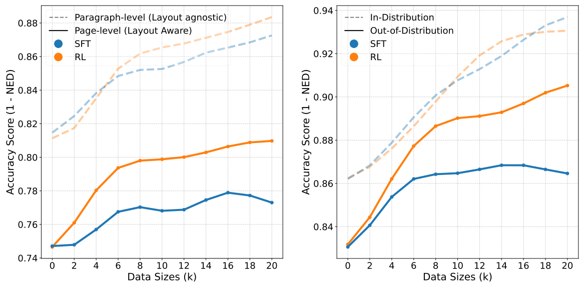

Figure 5: Impact of training strategies and data scale on document parsing performance on OmniDocBench.

Figure 5 (right) reports the in-distribution and out-of-distribution generalization performance of different training paradigms using the accuracy score. While all methods improve as the training data scale up, a consistent gap remains between in-distribution and out-of-distribution results. Notably, RL consistently outperforms SFT in both settings, with a larger margin observed under the out-of-distribution scenario, indicating stronger robustness to domain shifts. To further analyze this behavior, we consider a representative generalization setting in which the training data exclude the textbook and slides domains and are evaluated only on these unseen domains. This setup naturally forms an out-of-distribution scenario and enables a more detailed assessment of each model's ability to generalize to novel document types and layout structures. Additional results are provided in Appendix E.

6 Conclusion

We introduced LayoutRL, an end-to-end reinforcement learning framework that explicitly incorporates layout awareness into document parsing through verifiable, multi-aspect rewards. To support this training, we built Infinity-Doc-400K, a large-scale dataset combining synthetic and real-world documents with diverse layouts, and trained Infinity-Parser, a VLM-based parser. Experiments on OmniDocBench, olmOCR-Bench, PubTabNet, and FinTabNet show that our approach achieves state-of-the-art performance across languages and document types, outperforming both specialized pipelines and general-purpose VLMs. Beyond accuracy, LayoutRL improves training stability and demonstrates robustness across diverse document tasks, highlighting reinforcement learning as a promising direction for robust and transferable document intelligence.

7 Limitations

Our experiments are conducted on models up to a fixed scale, and we do not evaluate the proposed method on substantially larger models (e.g., 32B or 70B parameters). While we expect the layout-aware reinforcement learning framework to remain applicable, its behavior and scalability at much larger model sizes remain to be validated.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv:2303.08774*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2024. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv:2308.12966*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025a. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025b. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

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

Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. 2024. Nougat: Neural optical understanding for academic documents. *arXiv:2308.13418*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. 2025. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. *arXiv preprint arXiv:2501.17161*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, and 1 others. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Daniil Gurgurov and Aleksey Morshnev. 2024. Image-to-latex converter for mathematical formulas and text. *arXiv preprint arXiv:2408.04015*.

Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Chen Li, Ji Zhang, Qin Jin, Fei Huang, and 1 others. 2024a. mplug-docowl 1.5: Unified structure learning for ocr-free document understanding. *arXiv preprint arXiv:2403.12895*.

Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. 2024b. mplug-docowl2: High-resolution compressing for ocr-free multi-page document understanding. *arXiv preprint arXiv:2409.03420*.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Yao Hu, and Shaohui Lin. 2025. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking. *Preprint*, arXiv:2204.08387.

Wonseok Hwang, Jinyeong Yim, Seunghyun Park, Sohee Yang, and Minjoon Seo. 2021. Spatial dependency parsing for semi-structured document information extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, pages 330–343. Association for Computational Linguistics (ACL).

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Chenxia Li, Weiwei Liu, Ruoyu Guo, Xiaoting Yin, Kaitao Jiang, Yongkun Du, Yuning Du, Lingfeng Zhu, Baohua Lai, Xiaoguang Hu, Dianhai Yu, and Yanjun Ma. 2022a. Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system. *Preprint*, arXiv:2206.03001.

Yujia Li, David Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, Rémi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, and 1 others. 2022b. Competition-level code generation with alphacode. *Science*, 378(6624):1092–1097.

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

Chenglong Liu, Haoran Wei, Jinyue Chen, Lingyu Kong, Zheng Ge, Zining Zhu, Liang Zhao, Jianjian Sun, Chunrui Han, and Xiangyu Zhang. 2024. Focus anywhere for fine-grained multi-page document understanding. *arXiv:2405.14295*.

Yuqi Liu, Tianyuan Qu, Zhisheng Zhong, Bohao Peng, Shu Liu, Bei Yu, and Jiaya Jia. 2025a. Visionreasoner: Unified visual perception and reasoning via reinforcement learning. *arXiv preprint arXiv:2505.12081*.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. 2025b. Visual-rft: Visual reinforcement fine-tuning. *arXiv preprint arXiv:2503.01785*.

Nikolaos Livathinos, Christoph Auer, Maksym Lysak, Ahmed Nassar, Michele Dolfi, Panos Vagenas, Cesar Berrospi Ramis, Matteo Omenetti, Kasper Dinkla, Yusik Kim, and 1 others. 2025. Docling: An efficient open-source toolkit for ai-driven document conversion. *arXiv preprint arXiv:2501.17887*.

Tengchao Lv, Yupan Huang, Jingye Chen, Yuzhong Zhao, Yilin Jia, Lei Cui, Shuming Ma, Yaoyao Chang, Shaohan Huang, Wenhui Wang, Li Dong, Weiyao Luo, Shaoxiang Wu, Guoxin Wang, Cha Zhang, and Furu Wei. 2024. Kosmos-2.5: A multimodal literate model. *Preprint*, arXiv:2309.11419.

AI Meta. 2024. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models. *Meta AI Blog. Retrieved December*, 20:2024.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, and 1 others. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

Tobias Nipkow. 2003. Jinja: Towards a comprehensive formal semantics for a java-like language. In *Proc. Marktobderdorf Summer School*. IOS Press Amsterdam.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, and 1 others. 2024. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. *arXiv preprint arXiv:2412.07626*.

Vik Paruchuri. 2024. Marker.

Yi Peng, Xiaokun Wang, Yichen Wei, Jiangbo Pei, Weijie Qiu, Ai Jian, Yunzhuo Hao, Jiachun Pan, Tianyidan Xie, Li Ge, and 1 others. 2025. Skywork r1v: pioneering multimodal reasoning with chain-of-thought. *arXiv preprint arXiv:2504.05599*.

Jake Poznanski, Jon Borchardt, Jason Dunkelberger, Regan Huff, Daniel Lin, Aman Rangapur, Christopher Wilhelm, Kyle Lo, and Luca Soldaini. 2025. olmocr: Unlocking trillions of tokens in pdfs with vision language models. *arXiv preprint arXiv:2502.18443*.

RapidAI. 2023. Rapidtable. `https://github.com/RapidAI/RapidTable`.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Alex Su, Haozhe Wang, Weimin Ren, Fangzhen Lin, and Wenhu Chen. 2025. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Bin Wang, Zhuangcheng Gu, Guang Liang, Chao Xu, Bo Zhang, Botian Shi, and Conghui He. 2024a. Unimernet: A universal network for real-world mathematical expression recognition. *Preprint*, arXiv:2404.15254.

Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, Bo Zhang, Liqun Wei, Zhihao Sui, Wei Li, Botian Shi, Yu Qiao, Dahua Lin, and Conghui He. 2024b. Mineru: An open-source solution for precise document content extraction. *arXiv:2409.18839*.

Haozhe Wang, Long Li, Chao Qu, Fengming Zhu, Weidi Xu, Wei Chu, and Fangzhen Lin. 2025a. To code or not to code? adaptive tool integration for math language models via expectation-maximization. *arXiv preprint arXiv:2502.00691*.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. 2025b. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. 2025c. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024c. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Haoran Wei, Lingyu Kong, Jinyue Chen, Liang Zhao, Zheng Ge, Jinrong Yang, Jianjian Sun, Chunrui Han, and Xiangyu Zhang. 2025a. Vary: Scaling up the vision vocabulary for large vision-language model. In *European Conference on Computer Vision*, pages 408–424. Springer.

Haoran Wei, Chenglong Liu, Jinyue Chen, Jia Wang, Lingyu Kong, Yanming Xu, Zheng Ge, Liang Zhao, Jianjian Sun, Yuang Peng, and 1 others. 2024. General ocr theory: Towards ocr-2.0 via a unified end-to-end model. *arXiv:2409.01704*.

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

Renqiu Xia, Song Mao, Xiangchao Yan, Hongbin Zhou, Bo Zhang, Haoyang Peng, Jiahao Pi, Daocheng Fu, Wenjie Wu, Hancheng Ye, and 1 others. 2024. Docgenome: An open large-scale scientific document benchmark for training and testing multi-modal large language models. *arXiv preprint arXiv:2406.11633*.

Junting Lu Yaowei Zheng. 2025. Easyr1: An efficient, scalable, multi-modality rl training framework. `https://github.com/hiyouga/EasyR1`.

Huaye Zeng, Dongfu Jiang, Haozhe Wang, Ping Nie, Xiaotong Chen, and Wenhu Chen. 2025. Acecoder: Acing coder rl via automated test-case synthesis. *arXiv preprint arXiv:2502.01718*.

Qintong Zhang, Victor Shea-Jay Huang, Bin Wang, Junyuan Zhang, Zhengren Wang, Hao Liang, Shawn Wang, Matthieu Lin, Wentao Zhang, and Conghui He. 2024. Document parsing unveiled: Techniques, challenges, and prospects for structured information extraction. *arXiv preprint arXiv:2410.21169*.

Xinyi Zheng, Douglas Burdick, Lucian Popa, Xu Zhong, and Nancy Xin Ru Wang. 2021. Global table extractor (gte): A framework for joint table identification and cell structure recognition using visual context. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 697–706.

Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. 2020a. Image-based table recognition: data, model, and evaluation. In *European conference on computer vision*, pages 564–580.

Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. 2020b. Image-based table recognition: data, model, and evaluation. In *European conference on computer vision*, pages 564–580. Springer.

Hongbin Zhou, Xiangchao Yan, and Bo Zhang. 2024. Structeqtable-deploy: A high-efficiency open-source toolkit for table-to-latex transformation. `https://github.com/UniModal4Reasoning/StructEqTable-Deploy`.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, and 1 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

A Ethics Statement

This work does not involve human subjects, personal data, or sensitive user information. All experiments are conducted on publicly available datasets or synthetic data generated via HTML rendering. The newly introduced Infinity-Doc-400K dataset combines automatically generated synthetic documents with real-world samples that are pseudo-labeled through cross-model agreement and manually filtered to ensure quality. We have taken care

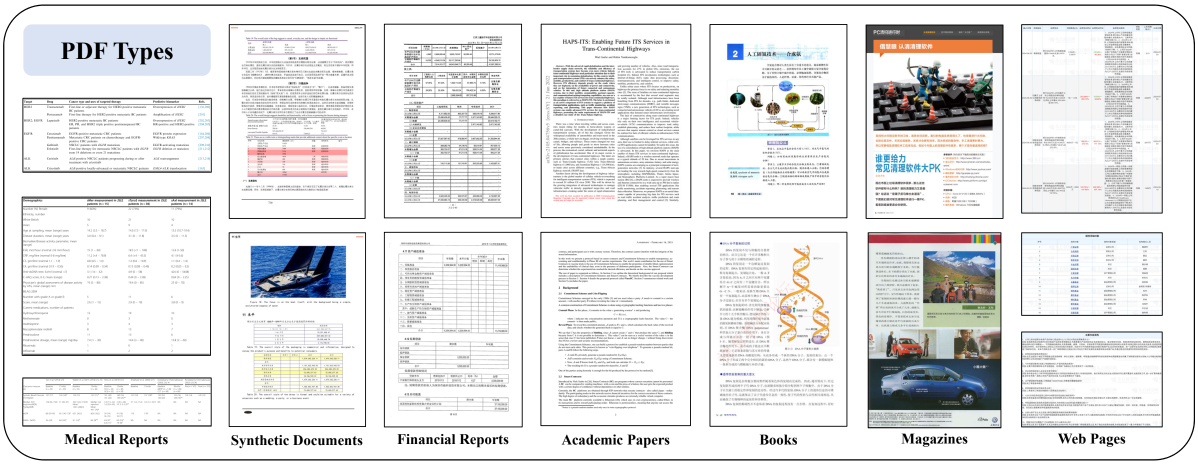

Figure 6: This figure illustrates a diverse collection of PDF document types commonly encountered in Infinity-Doc-55K, grouped into seven categories: Medical Reports, Synthetic Documents, Financial Reports, Academic Papers, Books, Magazines, and Web Pages.

Context Length (tokens)	[1,256)	[256,512)	[512,1K)	[1K,2K)	[2K,4K)	[4K,8K)	[8K,16K)	≥16K
Frequency (count)	25,125	40,680	119,041	102,757	70,554	39,780	2,482	63
Distribution (%)	6.27	10.16	29.72	25.66	17.62	9.93	0.62	0.02

Table 8: Distribution of training samples across different context length intervals.

Data Source	**Document Types**	**Size**	**Annotation Method**	**Cost**
Real-World Doc	Financial Reports	58.0K	Web + Pseudo-Label	High
Real-World Doc	Medical Reports	5.0K	Web + Pseudo-Label	High
Real-World Doc	Academic Papers	71.7K	Web + Pseudo-Label	High
Real-World Doc	Books	11.3K	Web + Pseudo-Label	High
Real-World Doc	Magazines	180.0K	Web + Pseudo-Label	High
Real-World Doc	Web Pages	5.0K	Web + Pseudo-Label	High
Synthetic	Synthetic Documents	69.0K	CC3M + Web + Wiki	Low

Table 9: Overview of document types in the Infinity-Doc-400K dataset, including data source, document type, annotation method, and collection cost.

to remove potentially harmful or inappropriate content, and the dataset will be released strictly for research purposes under an academic license. We believe our contributions do not pose risks of discrimination, bias, or privacy violations, and instead aim to advance the robustness and reliability of document parsing technologies for broad scientific and practical use.

B Data Details

As illustrated in Figure 6, the dataset spans seven diverse document domains, making it one of the most richly annotated and structurally varied resources to date. Each domain is represented by two sample pages, highlighting the broad variability in layout design, content structure, and semantic density. For instance, Medical Reports typically contain structured tables with clinical measurements and diagnostic notes. Synthetic Documents are algorithmically generated to replicate real-world formats, providing layout diversity for training robust parsers. Financial Reports feature dense tables and formal accounting records, while Academic Papers often follow two-column layouts with references, equations, and figures. Books combine narrative content with visual illustrations, and Magazines blend images and stylized text for reader engagement. Finally, Web Pages, when saved as PDFs, preserve HTML-based structures that integrate tables, lists, and dynamic elements. This visual taxonomy exemplifies the structural and semantic diversity present in real-world documents, highlighting the core challenge faced by document AI systems: reliably parsing heterogeneous layouts and extracting structured information across a wide variety of formats.

Table 9 provides an overview of the document types included in the Infinity-Doc-400K dataset, detailing the composition across both real-world and synthetic sources. The real-world portion consists of 48.6k documents spanning six domains: financial reports, medical reports, academic papers, books, magazines, and web pages. These documents were collected from the web and annotated using a pseudo-labeling pipeline based on expert model agreement. While this approach en-

Models	Book	Slides	Financial Report	Textbook	Exam Paper	Magazine	Academic Papers	Notes	Newspaper	Overall ↓
Based on Pipeline Tools										
MinerU	**0.055**	0.124	**0.033**	0.102	0.159	**0.072**	**0.025**	0.984	0.171	0.206
Marker	0.074	0.34	0.089	0.319	0.452	0.153	0.059	0.651	0.192	0.274
Mathpix	0.131	0.22	0.202	0.216	0.278	0.147	0.091	0.634	0.69	0.3
Based on Expert VLMs										
GOT-OCR	0.111	0.222	0.067	0.132	0.204	0.198	0.179	0.388	0.771	0.267
Nougat	0.734	0.958	1.000	0.820	0.930	0.83	0.214	0.991	0.871	0.806
Based on General VLMs										
GPT-4o	0.157	0.163	0.348	0.187	0.281	0.173	0.146	0.607	0.751	0.316
Qwen2-VL-72B	0.096	**0.061**	0.047	0.149	0.195	0.071	0.085	0.168	0.676	0.179
InternVL2-76B	0.216	0.098	0.162	0.184	0.247	0.150	0.419	0.226	0.903	0.3
Qwen2.5-VL-7B	0.222	0.131	0.194	0.268	0.203	0.230	0.195	0.249	0.394	0.230
InternVL3-8B	0.311	0.233	0.320	0.222	0.238	0.157	0.438	0.268	0.726	0.328
Based on Reinforcement Learning										
Infinity-Parser-7B	0.112	0.107	0.070	**0.093**	**0.082**	0.082	0.087	**0.141**	**0.153**	**0.104**

Table 10: End-to-end text recognition performance on OmniDocBench: evaluation using edit distance across 9 PDF page types. We compare with Mathpix, MinerU (Wang et al., 2024b), Marker (Paruchuri, 2024), GOT-OCR (Wei et al., 2024), Nougat (Blecher et al., 2024), GPT-4o (Brown et al., 2020), Qwen2-VL-72B (Wang et al., 2024c), InternVL2-76B (Chen et al., 2024), Qwen2.5-VL-7B (Bai et al., 2025a), InternVL3-8B (Zhu et al., 2025).

ables large-scale data acquisition, the resulting label quality is relatively low due to occasional inconsistencies across models. Additionally, real-world data collection incurs a high cost, especially in terms of manual filtering, formatting normalization, and layout validation.

C Training Context Length Distribution

Our model was trained with a maximum context length of 8K tokens. To provide a clearer picture of the training data, we report detailed statistics of the context length distribution in Table 14 and Table 8. The average context length is 1,765 tokens, with a maximum of 31,147 tokens. More than 73% of the samples fall within the [512, 4K) range. For sequences exceeding the 8K limit, we applied a left-truncation strategy to retain the semantically more relevant content at the end of the sequence.

D Implementation Details

We fine-tune the Qwen2.5-VL-7B model using GRPO within a distributed training setup based on Verl (Sheng et al., 2024; Yaowei Zheng, 2025), utilizing 8 A100 GPUs (80GB). Throughout our experiments, we set the KL coefficient $\beta = 1.0 \times 10^{-2}$. And for each problem instance, we sample 8 responses, each with a maximum length of 8192 tokens and a temperature of 1.0. Both the rollout batch size and the global batch size are set to 128. The actor model is updated using the AdamW optimizer with parameters $(\beta_1 = 0.9, \beta_2 = 0.99)$ and a learning rate 1.0×10^{-6}. The model is trained for 1.0 epoch for all experiments.

E More Results

Diverse Page Types Evaluation To further investigate model behavior across diverse document types, we evaluated text recognition performance on nine distinct page categories. As shown in Table 10, pipeline-based systems such as MinerU (Wang et al., 2024b) and Mathpix achieved strong results on structured formats like academic papers and financial reports. General-purpose vision-language models (VLMs) demonstrated better generalization on less formal page types, including presentation slides and handwritten notes. However, for challenging formats such as newspapers, most VLMs underperformed, while pipeline tools maintained relatively lower error rates. Notably, our proposed Infinity-Parser-7B achieved consistently low edit distances across all document types, outperforming both pipeline-based systems and general-purpose VLMs in overall accuracy. This highlights the robustness and adaptability of our reinforcement learning approach across diverse and complex document layouts.

Table Recognition evaluation Table 11 summarizes the performance of various models on the OmniDocBench table subset, evaluated along three dimensions: language diversity, table frame types, and special layout conditions. Notably, Infinity-Parser-7B achieves the best overall performance with an impressive score of 86.4, outperforming all other models across most individual metrics. It leads in nearly every category, including mixed-language settings (94.8), complex frame layouts

Model	Language			Table Frame Type				Special Situation				Overall ↑
	EN	*ZH*	*Mixed*	*Full*	*Omission*	*Three*	*Zero*	*Merge Cell*(+/-)	*Formula*(+/-)	*Colorful*(+/-)	*Rotate*(+/-)	
PaddleOCR (Li et al., 2022a)	76.8	71.8	80.1	67.9	74.3	81.1	74.5	70.6/75.2	71.3/74.1	72.7/74.0	23.3/74.6	73.6
RapidTable (RapidAI, 2023)	80.0	83.2	91.2	83.0	79.7	83.4	78.4	77.1/85.4	76.7/83.9	77.6/84.9	25.2/83.7	82.5
StructEqTable (Zhou et al., 2024)	72.8	75.9	83.4	72.9	76.2	76.9	88.0	64.5/81.0	69.2/76.6	72.8/76.4	30.5/76.2	75.8
GOT-OCR (Wei et al., 2024)	72.2	75.5	85.4	73.1	72.7	78.2	75.7	65.0/80.2	64.3/77.3	70.8/76.9	8.5/76.3	74.9
Qwen2-VL-7B (Wang et al., 2024c)	70.2	70.7	82.4	70.2	62.8	74.5	80.3	60.8/76.5	63.8/72.6	71.4/70.8	20.0/72.1	71.0
InternVL2-8B (Chen et al., 2024)	70.9	71.5	77.4	69.5	69.2	74.8	75.8	58.7/78.4	62.4/73.6	68.2/73.1	20.4/72.6	71.5
Qwen2.5-VL-7B (Wang et al., 2024c)	87.4	80.7	93.5	86.4	85.1	84.1	88.7	77.5/89.8	82.1/87.2	77.1/87.5	56.5/86.0	85.5
InternVL3-8B (Zhu et al., 2025)	79.5	86.0	91.7	85.5	80.7	83.9	85.9	71.9/90.9	74.0/86.7	82.1/85.3	12.6/85.5	84.3
Infinity-Parser-7B	84.7	**86.7**	**94.8**	85.5	**86.5**	**87.4**	**89.4**	**78.6/90.7**	81.9/ **87.5**	**83.2/88.0**	**68.8/86.7**	**86.4**

Table 11: Component-level Table Recognition evaluation on OmniDocBench table subset. *(+/-)* means *with/without* special situation.

Models	CDM ↑	ExpRate@CDM ↑	BLEU ↑	Norm Edit ↓
GOT-OCR	74.1	28.0	55.07	0.290
Mathpix	86.6	2.8	66.56	0.322
Pix2Tex	73.9	39.5	46.00	0.337
UniMERNet-B	85.0	60.2	60.84	0.238
GPT-4o	86.8	65.5	45.17	0.282
InternVL2-76B	67.4	54.5	47.63	0.308
Qwen2-VL-72B	83.8	55.4	53.71	0.285
Infinity-Parser-7B	**93.0**	**71.1**	**62.30**	**0.253**

Table 12: Quantitative results on formula recognition benchmarks. Our method achieves state-of-the-art performance across CDM, ExpRate@CDM, BLEU, and NED metrics.

(e.g., omission and three-line formats), and challenging special situations such as merged cells, formulas, and rotations. This demonstrates its strong generalization ability and robustness across diverse and noisy table formats.

Quantitative Results for Formula Metrics In addition to structural parsing metrics, formula recognition is an essential component of document understanding. In our methods section, we introduced CDM, NED, and BLEU as evaluation metrics for formula recognition. CDM measures component-level detection accuracy under relaxed matching, while ExpRate@CDM denotes strict recall requiring perfect character-level correctness. NED reflects normalized edit distance, and BLEU evaluates n-gram overlap quality. While CDM and NED results were already included in Table 12, due to space constraints BLEU results were omitted. For completeness, we now provide the full quantitative results across multiple baselines in Table 12. As shown, our model (Infinity-Parser-7B) achieves new state-of-the-art performance across all metrics, surpassing both traditional OCR systems (e.g., Mathpix, Pix2Tex) and advanced vision-language models (e.g., GPT-4o, Qwen2-VL-72B).

Training Stability Across Task Types. As shown in Figure 7, we compare SFT and Layout-Aware RL on four OmniDocBench sub-tasks: table recognition, text recognition, formula parsing, and reading order prediction. Across all tasks, RL consistently achieves better performance, yielding higher scores on TEDS and lower errors on NED. More importantly, the RL curves exhibit smoother trajectories with stable improvements over training, while SFT shows large fluctuations and even performance regressions at certain stages. These results indicate that layout-aware rewards not only improve final accuracy but also enhance training stability throughout optimization.

Robustness Across Diverse Document Tasks. Figure 8 compares RL, SFT, and Zero-Shot (Base) across diverse document parsing tasks. On olmOCR (left), RL consistently achieves higher Levenshtein distance similarity score, especially on challenging cases like old scans and table tasks. At the same time, SFT offers moderate gains over Zero-Shot but remains behind RL. On OmniDocBench (right), RL also outperforms the other methods across most document types, showing notable improvements on books, reports, and academic texts. Overall, RL demonstrates greater robustness and better generalization in both structural parsing on olmOCR and text recognition on omnidocbench.

Analysis of Generalization. The Figure 9 compares document parsing performance across different training steps. The left two plots (magazine, research report) correspond to In-Distribution settings, where training and evaluation domains are aligned, while the right two plots (colorful textbook, slides) correspond to OOD settings, where evaluation involves unseen document types. In the in-distribution case, SFT achieves stable paragraph-level accuracy but its page-level performance tends to plateau, reflecting reliance on surface patterns. By contrast, RL continues to improve with data scale, achieving notable gains in page-level accu-

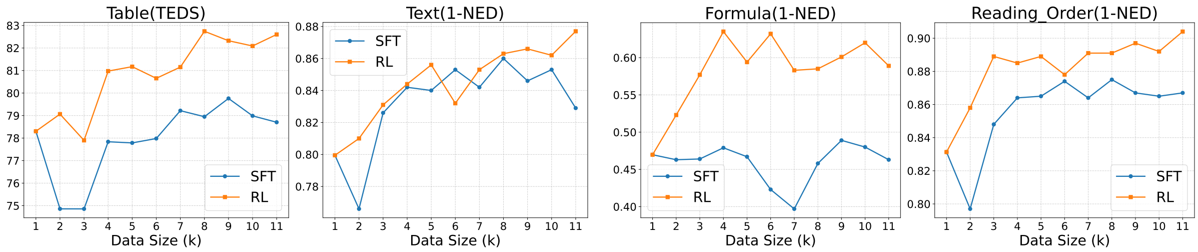

Figure 7: Performance comparison of SFT and Layout-Aware RL on OmniDocBench sub-tasks.

Model	Cond.	OverallEdit ↓		TextEdit ↓		FormEdit ↓		TableTEDS ↑		TableEdit ↓		ReadOrderEdit ↓	
		EN	ZH	EN	ZH	EN	ZH	EN	ZH	EN	ZH	EN	ZH
Infinity-Parser-7B	Normal	0.122	0.174	0.053	0.079	0.250	0.366	83.3	80.9	0.122	0.156	0.062	0.095
Infinity-Parser-7B	Noise	0.180	0.303	0.137	0.271	0.295	0.517	75.8	71.8	0.183	0.210	0.105	0.214
MinerU 2.5	Normal	0.110	0.174	0.050	0.074	0.256	0.470	88.4	89.3	0.090	0.084	0.045	0.067
MinerU 2.5	Noise	0.187	0.283	0.175	0.294	0.287	0.485	80.9	81.2	0.147	0.126	0.137	0.229

Table 13: Performance comparison under normal and noisy conditions. Lower is better for edit-based metrics, while higher is better for TableTEDS.

racy. In the OOD case, SFT performance degrades more severely, while RL maintains robustness and shows stronger improvements, highlighting its ability to capture global structural dependencies and generalize across distributions.

Metric	Min	Max	Average	Median	Std
Value	17	31,147	1,765	1,127	1,692

Table 14: Summary statistics of training context length.

Robustness Evaluation under Realistic Degradations. We further evaluate our method against recent VLM-based document parsing models and assess its robustness under realistic degradation conditions. As shown in Table 13, we include comparisons with MinerU 2.5 and introduce explicit evaluations under noise perturbations. Under standard settings, Infinity-Parser-7B achieves competitive performance (OverallEdit EN 0.122 / ZH 0.174), closely matching MinerU 2.5 (0.110 / 0.174). To systematically evaluate robustness, we design a structured degradation pipeline that simulates realistic "old scan" artifacts, including downsampling, blur, additive noise (Gaussian and salt-and-pepper), perspective distortion, localized text fading, exposure drift, and JPEG compression. Under these challenging conditions, our model exhibits stable and controlled performance degradation (EN 0.180 / ZH 0.303), remaining comparable to MinerU 2.5.

Notably, MinerU 2.5 is trained on approximately 6.9M samples, whereas our model relies on only 43K real annotated documents, highlighting a substantial gap in training scale. These results demonstrate that our approach can achieve strong performance with significantly less supervision. Beyond synthetic perturbations, our evaluation also includes naturally degraded data. As reported in Table3 and Figure8, we evaluate on the "Old Scans" subset of olmOCR-Bench, which contains real-world historical documents with authentic noise, blur, fading, and scanning artifacts. On this challenging benchmark, Infinity-Parser achieves a score of 83.8, substantially outperforming strong baselines such as MinerU (47.4) and GOT-OCR (52.0). These findings provide strong evidence that our method is robust to real-world degradations and highlight the effectiveness of high-quality data construction and layout-aware training strategies for document parsing under limited supervision.

F Benchmarks Details

OmniDocBench (Ouyang et al., 2024) We conduct evaluation on OmniDocBench, a comprehensive benchmark that covers diverse document types and content modalities. To assess parsing performance across different structural elements, we employ two primary evaluation metrics: Normalized Edit Distance (NED), which measures the minimum edit operations required to transform one string into another normalized by the target string length, and Tree Edit Distance-based Similarity (TEDS), which captures structural similarities by comparing tree representations of HTML tables. These metrics are applied to different subtasks: NED is used to evaluate pure text, formula transcription, and reading order; TEDS combined with

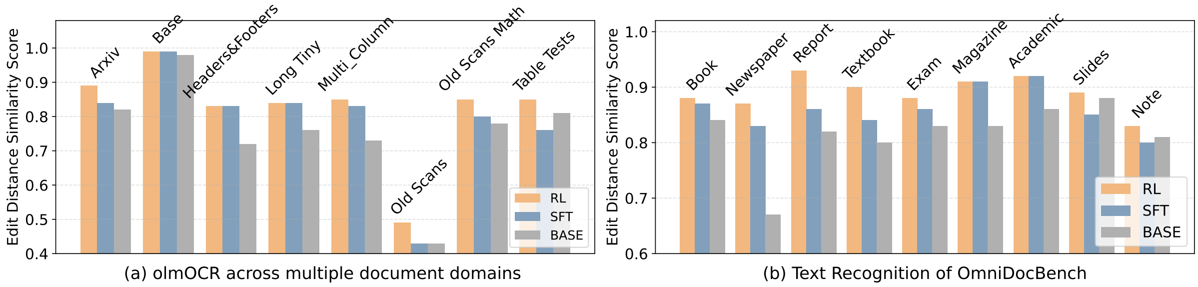

Figure 8: Comparison of model performance on different document parsering tasks.

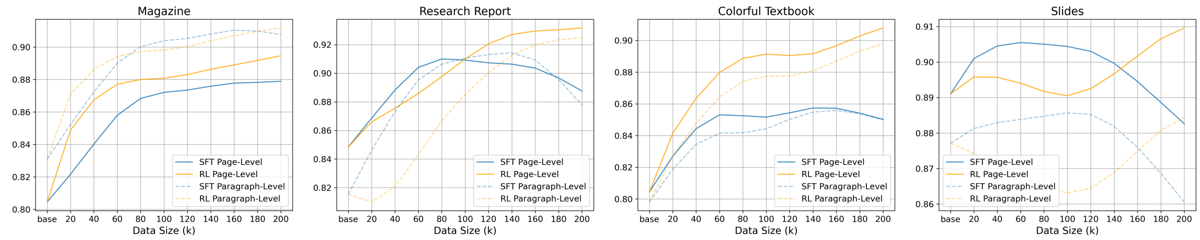

Figure 9: Generalization comparison of SFT and Layout-Aware RL. X-axis represents training steps. Y-axis represents 1-NED scores.

NED is used to evaluate both structural and content accuracy of table parsing.

PubTabNet (Zhong et al., 2020a) A widely used benchmark for table recognition, containing 500,777 training and 9,115 validation images with diverse scientific table structures. Evaluation is conducted on the validation set.

FinTabNet (Zheng et al., 2021) Focused on financial documents, this dataset includes 112,000 single-page scanned documents, with 92,000 cropped training images and 10,656 for testing. It features dense layouts and detailed annotations for both structure and content evaluation.

olmOCR-Bench (Poznanski et al., 2025) This is a benchmark developed to automatically and reliably evaluate document-level OCR performance across a wide range of tools. Unlike traditional evaluation metrics such as edit distance—which may penalize valid variations or fail to capture critical semantic errors—olmOCR-Bench focuses on verifying simple, unambiguous, and machine-checkable "facts" about each document page, similar to unit tests. For instance, it checks whether a specific sentence appears exactly in the OCR output. The benchmark operates directly on single-page PDFs to preserve digital metadata, which can be beneficial for certain OCR systems, and to maintain the integrity of the original document format. Designed for flexibility, olmOCR-Bench supports outputs in Markdown or plain text, allowing for seamless evaluation of both open-source and custom OCR pipelines.

G Prompt Strategy for Parsing Tasks.

Prompt Template summarizes the prompt designs for two key parsing tasks: document parsing and table parsing. For document parsing, the prompts instruct the model to recognize visual regions and convert their contents into structured Markdown. This design ensures consistent region-level extraction across documents with diverse layouts.

For table parsing, although the prompts are phrased differently, they share the same objective: transforming table content from images into HTML. This diversity encourages the model to generalize across variations in phrasing and reduces overfitting to a single instruction template. Notably, HTML is used here to match the evaluation format, but the resulting outputs can be easily converted to Markdown if needed for downstream use.

H Case Analysis

Figure 10 illustrates a progressive improvement in Markdown generation quality across different training strategies. The zero-shot model fails to capture key structural elements, omitting titles and producing redundant or incomplete content. With SFT, the model better identifies section headers and general layout but still suffers from symbol-level errors and repeated outputs. In contrast, the layout-aware RL model demonstrates the most accurate and coherent result, successfully preserving the

Prompt Template

Document Parsing: You are an AI assistant specialized in converting PDF images to Markdown format. Please follow these instructions for the conversion:
1. Text Processing:
- Accurately recognize all text content in the PDF image without guessing or inferring.
- Convert the recognized text into Markdown format.
- Maintain the original document structure, including headings, paragraphs, lists, etc.
2. Mathematical Formula Processing:
- Convert all mathematical formulas to LaTeX format.
- Enclose inline formulas with $ $. For example: This is an inline formula $E = mc^2$
- Enclose block formulas with $$ $$. For example:

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

3. Table Processing:
- Convert tables to Markdown format.
4. Figure Handling:
- Ignore figures content in the PDF image. Do not attempt to describe or convert images.
5. Output Format:
- Ensure the output Markdown document has a clear structure with appropriate line breaks between elements.
- For complex layouts, try to maintain the original document's structure and format as closely as possible.
Please strictly follow these guidelines to ensure accuracy and consistency in the conversion. Your task is to accurately convert the content of the PDF image into Markdown format without adding any extra explanations or comments.

Table Parsing:

1. Please encode the table from the image into HTML format.
2. Render the table in the image as HTML code, please.
3. Please transform the table from the image into HTML format.
4. Convert the image's table data into the HTML structure.
5. Transform the image's table into the HTML format, please.
6. Convert the table found in the image into HTML format.

Example Input: A PDF with headings, paragraphs, and a table.
Example Output: Markdown reconstruction with proper hierarchy.

document hierarchy and eliminating redundancy. This highlights the effectiveness of layout-aware rewards in guiding the model toward semantically and structurally faithful document parsing.

Infinity-Parser exhibits consistent improvements across a wide spectrum of document types, including academic papers, books, colorful textbooks, exam papers, magazines, government notices, newspaper articles, and PowerPoint-style slides. These gains are reflected in structural parsing, title and content recognition, formatting accuracy, and robustness to diverse visual layouts. As shown in Figures 12 through 11, we provide detailed visual comparisons with existing models, where Infinity-Parser consistently achieves superior results. These findings underscore the effectiveness and generalizability of our layout-aware RL approach across complex, real-world PDF formats.

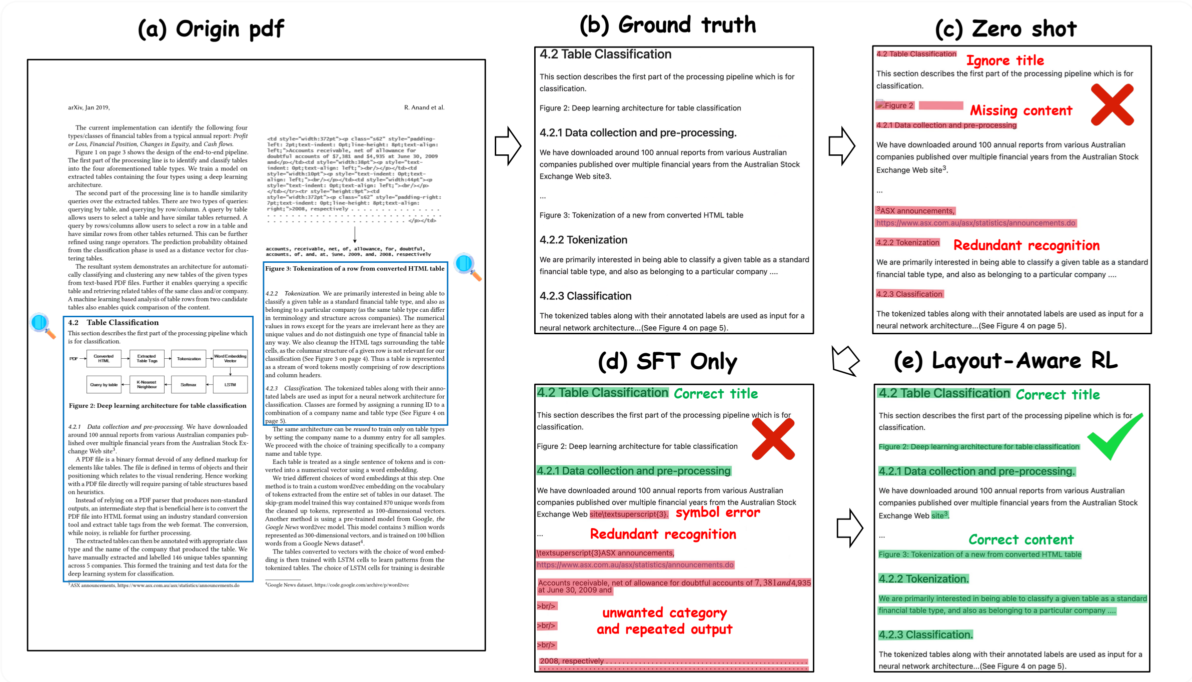

Figure 10: Comparison of four Markdown generation results on a single case, illustrating progressive improvement from direct inference to full reward integration.

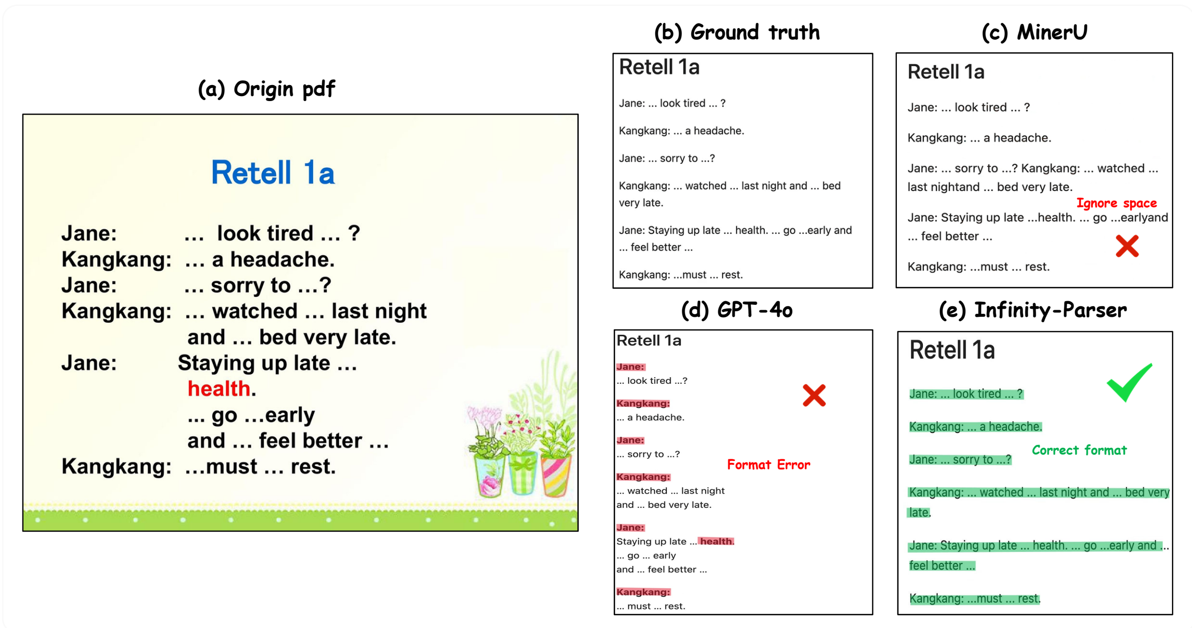

Figure 11: Comparison of Markdown extraction from a PowerPoint-style PDF slide. The figure includes the original slide (a), human-annotated ground truth (b), and outputs from MinerU, GPT-4o, and Infinity-Parser (c–e). MinerU and GPT-4o both struggle with layout fidelity, introducing spacing and formatting errors. In contrast, Infinity-Parser preserves the dialogue structure and formatting accurately, demonstrating its capability to handle informal, visually decorated slides effectively.

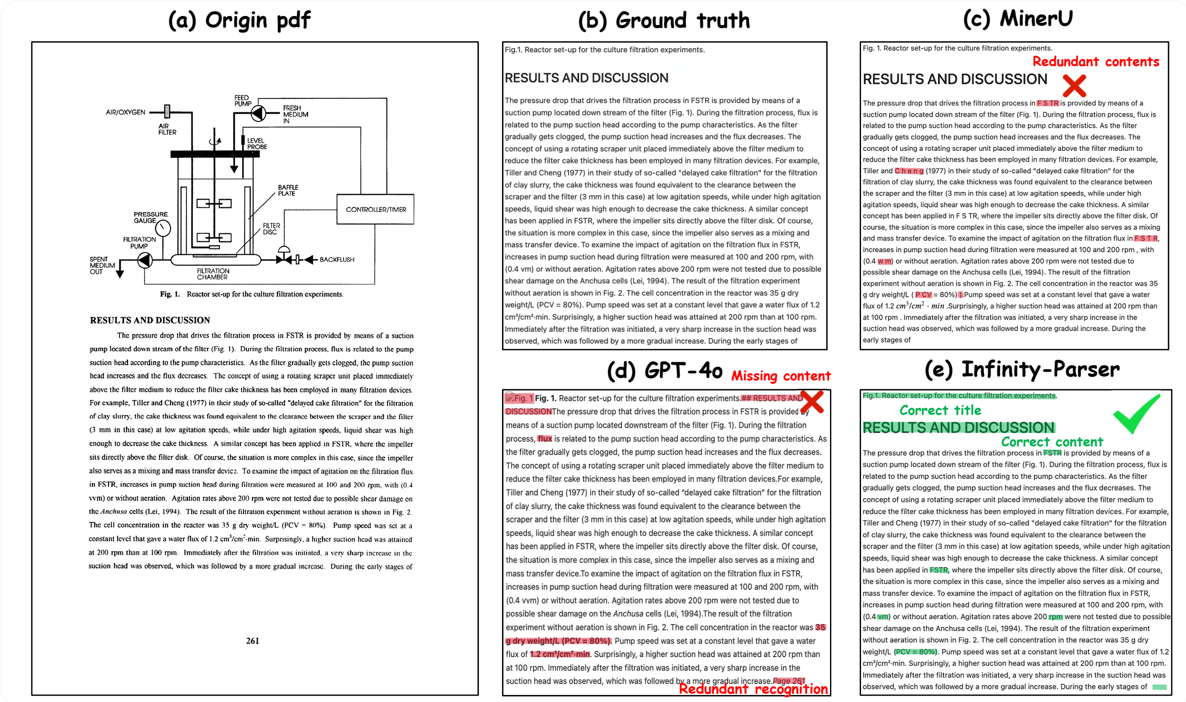

Figure 12: Comparison of Markdown extraction from academic literature using different models. The figure shows the original PDF (a), ground truth annotations (b), and extraction results from three models: MinerU, GPT-4o, and Infinity-Parser (c–e). Infinity-Parser produces the most accurate output, correctly identifying titles and content while avoiding redundancy and omissions.

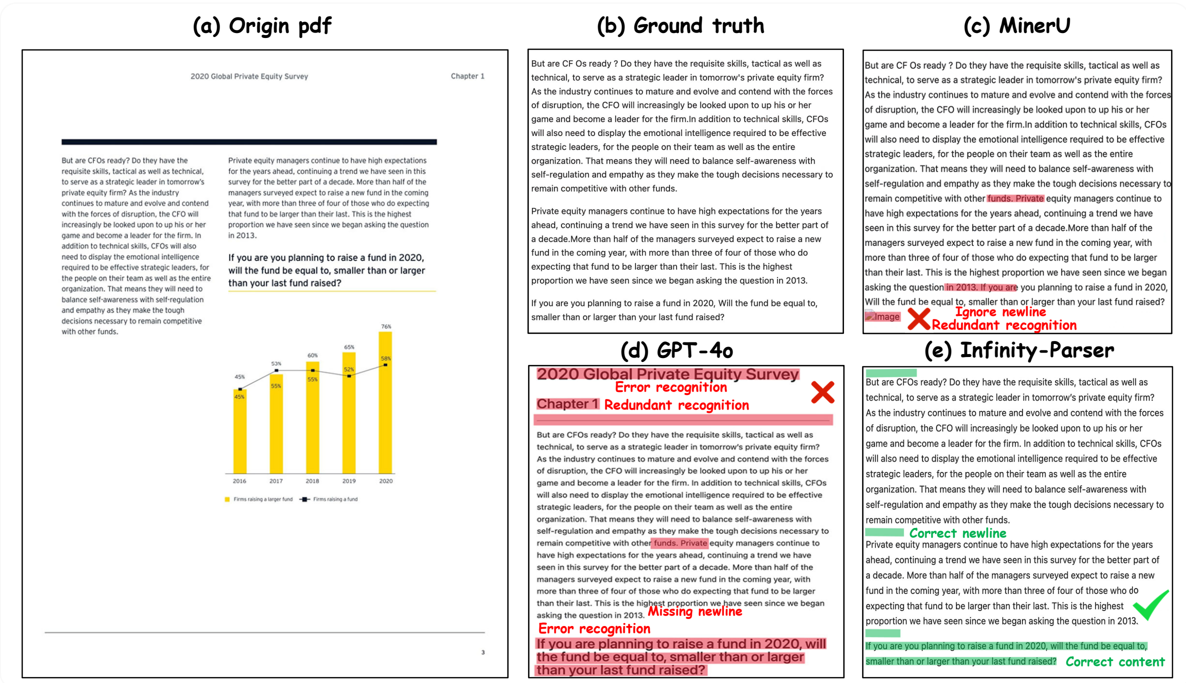

Figure 13: Comparison of Markdown extraction from a book-style PDF using different models. This figure displays the original document (a), human-annotated ground truth (b), and the extraction results from MinerU, GPT-4o, and Infinity-Parser (c–e). GPT-4o introduces multiple errors, such as incorrect headings, redundancy, and missing lines. MinerU retains unnecessary line breaks and repeated text. In contrast, Infinity-Parser correctly identifies the content structure, including titles and paragraphs, producing clean and accurate output.

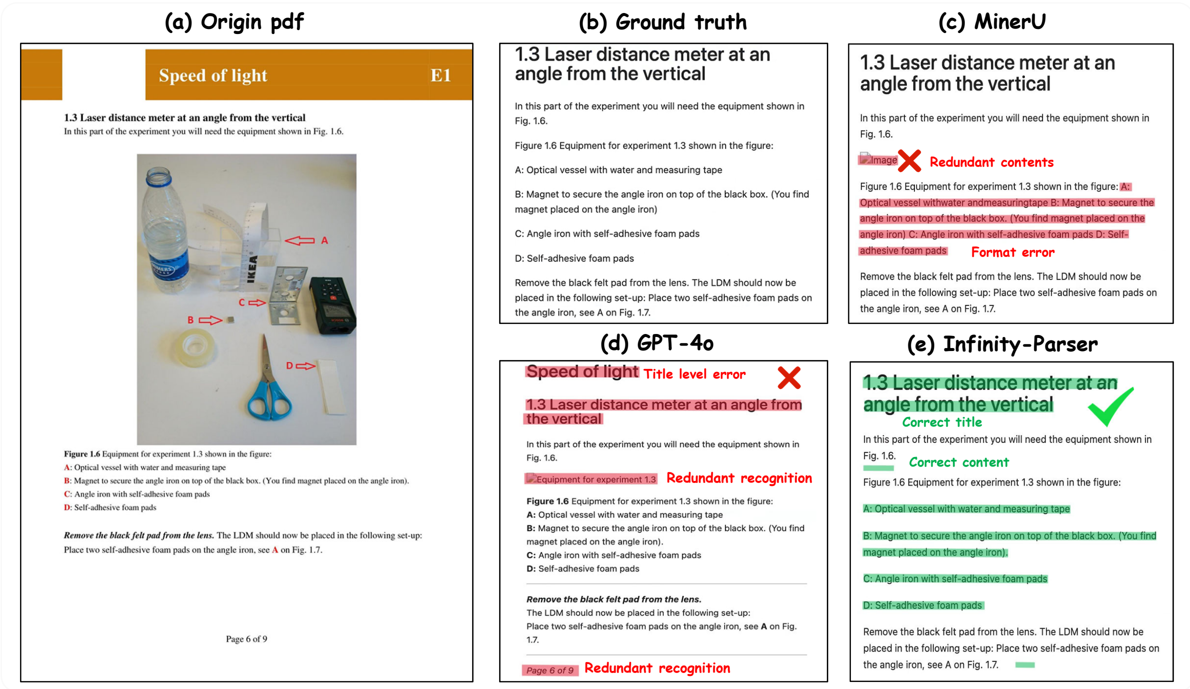

Figure 14: Comparison of Markdown extraction from an exam-style PDF. The figure presents the original exam document (a), ground truth annotations (b), and the extraction results from MinerU, GPT-4o, and Infinity-Parser (c–e). GPT-4o and MinerU both introduce redundant text and formatting errors, such as incorrect title levels and misplaced content. In contrast, Infinity-Parser accurately captures the heading hierarchy and structured list format, faithfully reproducing the content as intended in the original layout.

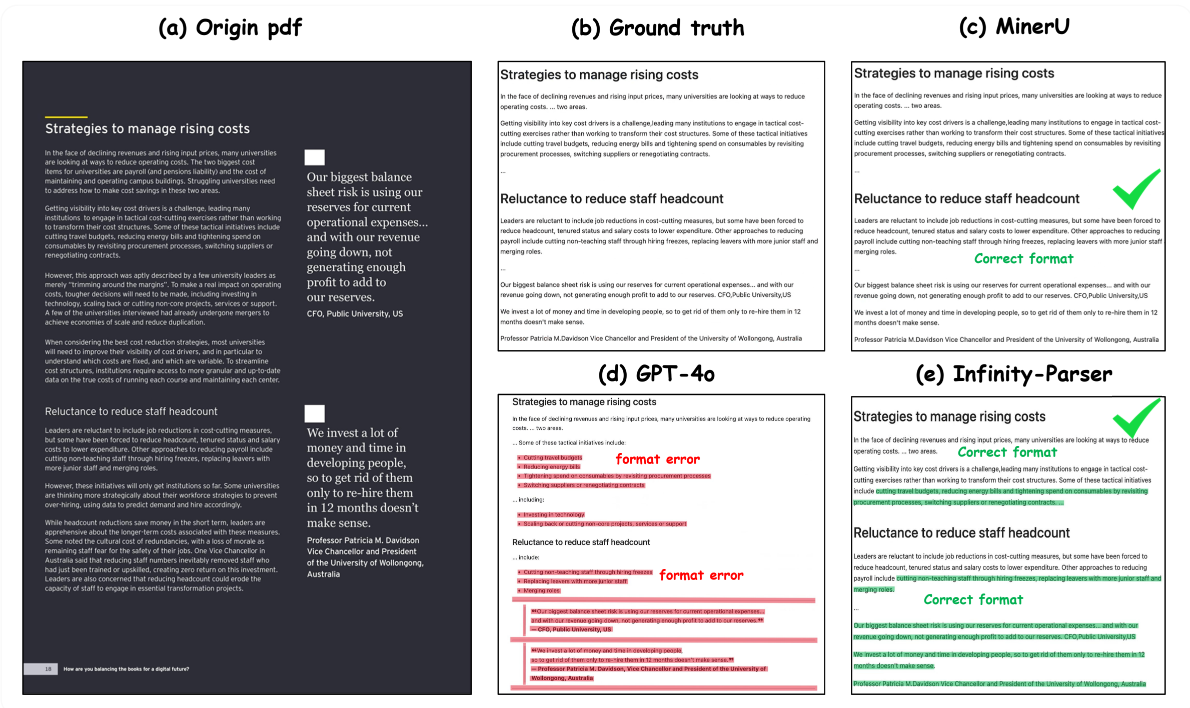

Figure 15: Comparison of Markdown extraction from a magazine-style PDF. The figure shows the original visually-rich page (a), ground truth annotations (b), and results from MinerU, GPT-4o, and Infinity-Parser (c–e). Due to the complex layout and dark background, GPT-4o suffers from significant formatting errors. Infinity-Parser accurately preserves the structural hierarchy and formatting, demonstrating robustness in handling stylized layouts.

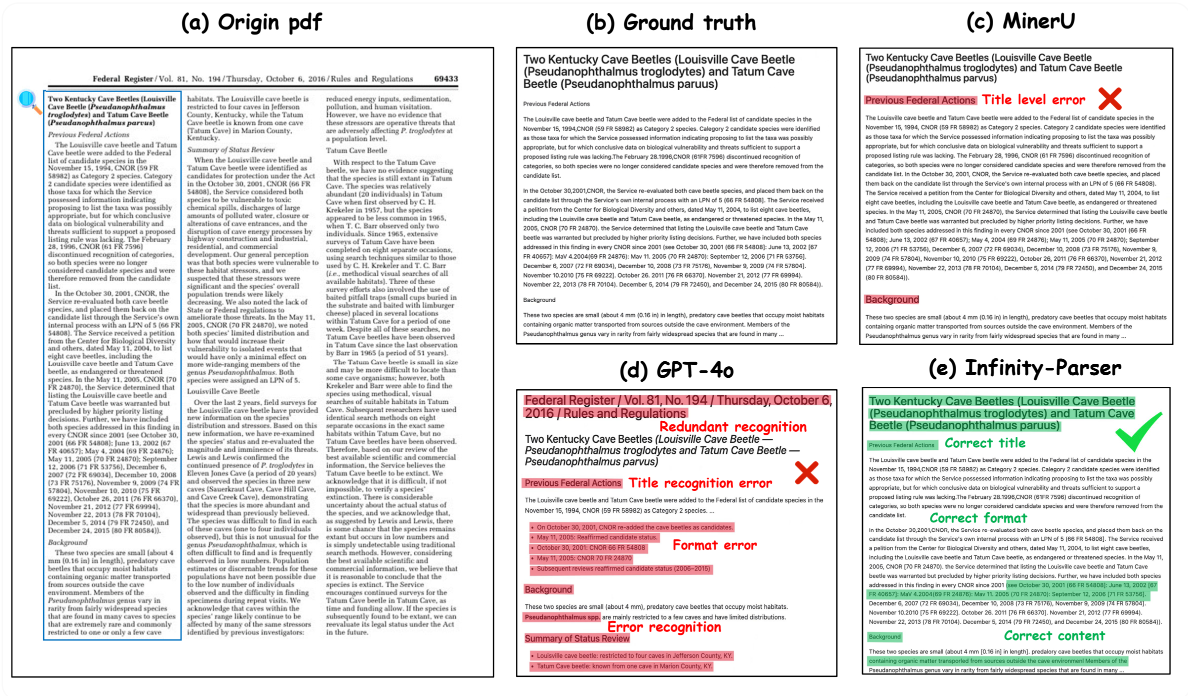

Figure 16: Comparison of Markdown extraction from a newspaper-style PDF. This figure presents the original densely formatted page (a), ground truth annotations (b), and the results from MinerU, GPT-4o, and Infinity-Parser (c–e). Due to the complex multi-column layout and title hierarchy, both MinerU and GPT-4o exhibit issues such as incorrect title levels, redundant content, and format errors. In contrast, Infinity-Parser accurately identifies the main title, maintains structural formatting, and preserves content integrity, demonstrating strong layout understanding in challenging document types.