LENGTH-AGNOSTIC SPARSE ATTENTION FOR UNLIMITED DOCUMENT PARSING

Anonymous authors
Paper under double-blind review

ABSTRACT

Real-world applications such as financial analysis and contract review often involve documents spanning tens or hundreds of pages, making ultra-long document parsing increasingly important. Existing solutions face a fundamental trade-off between scalability and cross-page modeling, as page-by-page parsing loses cross-page dependencies while full-document modeling faces both prohibitive computation and an overwhelming visual context. Fixed-budget sparse attention provides a natural way to bound the context accessed by each query, but reliably selecting the few relevant blocks from an expanding multi-page context remains challenging. Crucially, we observe that document evidence exhibits a hierarchical sparsity pattern, with relevant evidence sparse across pages but locally concentrated within relevant pages. Leveraging this structure, we propose **Length-Agnostic Sparse Attention (LASA)**, which enables reliable fixed-budget routing with an attention scope independent of document length, providing a scalable foundation for unlimited document parsing. LASA combines Page Canonicalization to exploit page structure, constrain the visual search space, and stabilize page-level positional modeling, with Token Bounding to reliably select a fixed budget of query-relevant visual blocks. Together, these designs enable efficient end-to-end cross-page modeling with stable performance as document length increases. We further introduce **MarathonDocBench**, a benchmark for hundred-page document parsing with substantially longer documents and richer cross-page structures than existing benchmarks. The resulting LASA-based parser consistently outperforms strong page-by-page pipelines and general-purpose large-scale VLMs on both MPDocBench and MarathonDocBench. By bounding the effective attention scope of each query, LASA keeps per-step decoding time approximately constant as document length increases and achieves up to $8.2\times$ speedup over full attention at the hundred-page scale.

1 INTRODUCTION

Documents are becoming a key long-context interface for AI, yet real-world documents often span tens or hundreds of pages with structure and semantics coupled across page boundaries. Reliable document understanding therefore requires preserving cross-page dependencies throughout parsing. Existing solutions, however, face a fundamental trade-off between scalability and context preservation: page-by-page parsers scale well but fragment documents into isolated units, whereas end-to-end VLMs preserve document-level context at the cost of rapidly growing computation and an expanding visual search space.

As illustrated in Figure 1, full attention becomes increasingly impractical as more pages are added, since every decoding query must attend to an ever-growing context. UnlimitedOCR Yin et al. (2026b) effectively addresses the text-side decoding bottleneck with R-SWA, while its remote visual context still grows with document length. This makes remote visual access the scalability bottleneck, motivating fixed-budget sparse attention. However, fixed-budget sparse routing becomes increasingly difficult as document length grows, since only a few relevant blocks must be identified from an ever-expanding multi-page visual context.

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

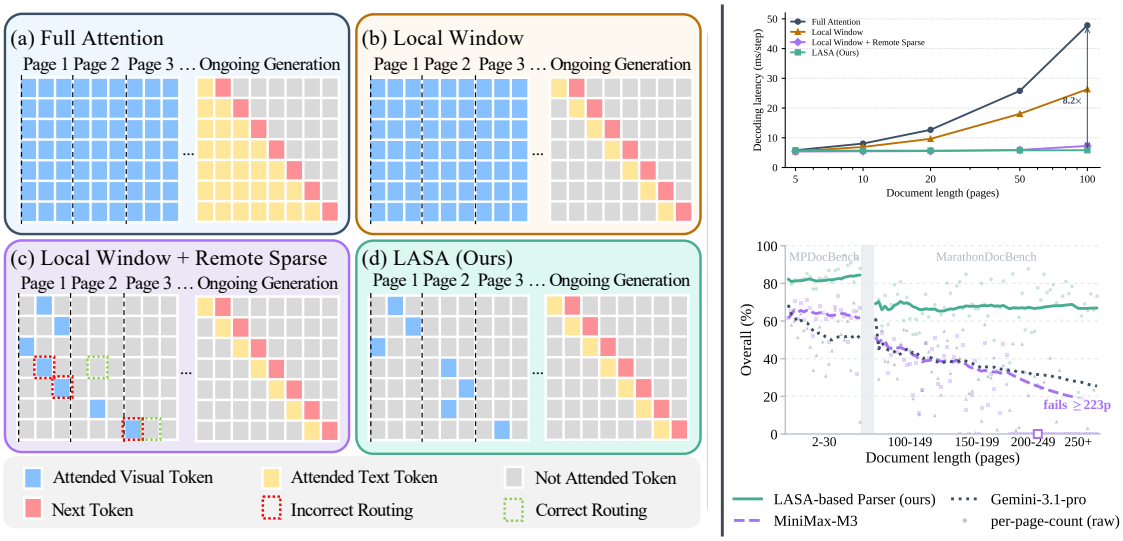


Figure 1: Comparison of attention patterns and scaling behavior. LASA enables reliable fixed-budget routing, yielding nearly length-invariant decoding latency and stable long-document parsing.

Rather than treating all remote visual blocks as a flat candidate pool, we find that useful evidence follows the document hierarchy. Only a small number of pages are typically relevant to each decoding query, and within those pages, remote visual evidence is further concentrated in a few local regions. Motivated by this observation, we propose **Length-Agnostic Sparse Attention (LASA)**, a hierarchical sparse attention framework that keeps the effective per-query attention scope bounded as document length grows. LASA combines Page Canonicalization, which structures the remote visual context by page and stabilizes page-level positional modeling, with Token Bounding, which uses an end-to-end Token-Bounded Router to select a fixed number of query-relevant visual blocks. The former simplifies routing over an expanding multi-page context, while the latter directly optimizes fixed-budget selection for the language modeling objective. Together, they enable reliable cross-page modeling under a bounded attention budget, providing a scalable path toward document parsing at unlimited page scales. To evaluate this capability at the hundred-page scale, we further introduce **MarathonDocBench**, the first benchmark specifically designed for hundred-page document parsing. Experiments on MPDocBench and MarathonDocBench show that LASA consistently outperforms strong page-by-page pipelines and existing large-scale VLMs. Our main contributions are as follows.

- We propose **Length-Agnostic Sparse Attention (LASA)**, a hierarchical sparse attention framework for reliable fixed-budget routing in long documents. LASA combines Page Canonicalization to exploit document structure and simplify multi-page block selection with Token Bounding to directly optimize the selection of a fixed budget of visual blocks under the language modeling objective.
- To systematically evaluate long-document parsing at scale, we introduce **MarathonDocBench**, a benchmark specifically designed for hundred-page document parsing. It substantially extends existing evaluation in document length and cross-page structural complexity, enabling joint assessment of parsing quality and inference scalability in the hundred-page regime.
- We instantiate LASA for end-to-end multi-page parsing and evaluate it on MPDocBench and MarathonDocBench. LASA-based parser consistently outperforms strong page-by-page pipelines and existing general-purpose large-scale VLMs, while maintaining nearly length-invariant per-step

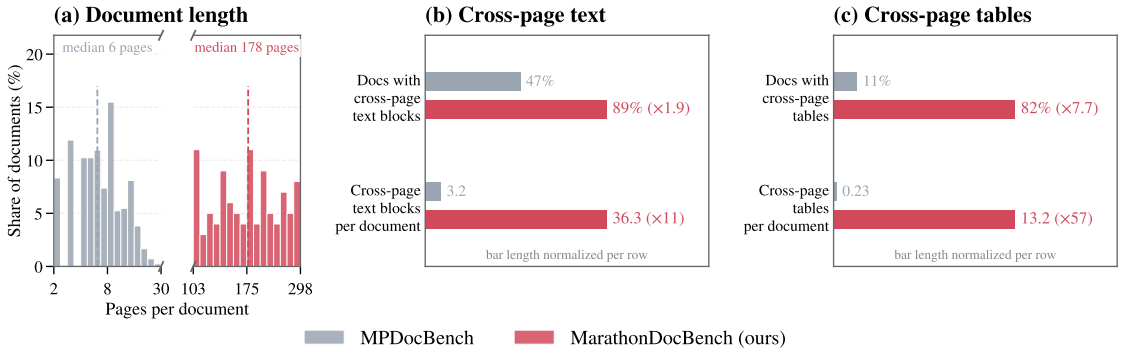


Figure 2: MarathonDocBench versus MPDocBench. MarathonDocBench extends document length from tens to hundreds of pages (a), with substantially richer cross-page structures as shown in (b) and (c).

decoding cost and achieving substantial speedups over R-SWA and full attention at the hundred-page scale.

2 Problem Setup

2.1 Problem Formulation

Let $\mathcal{D} = (P_1, \ldots, P_N)$ denote an ordered document of N pages, where each page P_i is encoded into visual tokens $\mathcal{V}_i = (v_{i,1}, \ldots, v_{i,M_i})$. At decoding step t, the model operates on a unified token sequence $\mathcal{X}_t$ consisting of the multi-page visual tokens and the textual prefix $y_{<t}$:

$$\mathcal{X}_t = \mathrm{Concat}\left(\mathcal{V}_1, \ldots, \mathcal{V}_N, y_{<t}\right). \tag{1}$$

Multi-page document parsing generates a unified structured transcription $\mathcal{Y} = (y_1, \ldots, y_T)$ conditioned on this document-level sequence:

$$p_\theta(\mathcal{Y} \mid \mathcal{D}) = \prod_{t=1}^{T} p_\theta\left(y_t \mid \mathcal{X}_t\right). \tag{2}$$

2.2 MarathonDocBench

Existing multi-page document parsing benchmarks neither extend to the hundred-page scale nor sufficiently capture the rich cross-page structures that emerge in long documents. To address this gap, we introduce MarathonDocBench, the first benchmark specifically designed for hundred-page document parsing. It contains 100 documents spanning 103–300 pages with 205,712 annotated layout elements, extending the document-length range by roughly an order of magnitude beyond MPDocBench Zhou et al. (2026).

Beyond document length, MarathonDocBench contains substantially richer cross-page structures than MPDocBench, with cross-page text and tables appearing much more frequently and densely across documents, as summarized in Figure 2. This makes MarathonDocBench a substantially more challenging testbed for evaluating whether end-to-end parsers can preserve document-level structure while scaling from tens to hundreds of pages. To ensure reliable evaluation, structural annotations, including bounding boxes, block type, hierarchy, reading order, and cross-page continuations of text and tables, are initialized from a strong page-by-page parser and carefully refined by human annotators. Content transcriptions are initialized from

the same parser outputs and verified through an iterative VLM-based refinement pipeline, with flagged disagreements adjudicated by human annotators.

2.3 Challenges of Fixed-Budget Routing

We first examine whether conventional sparse routing remains reliable in the multi-page setting. Following MSA Lai et al. (2026), we train routers on 200K single-page and 200K multi-page samples, respectively, using KL distillation from full-attention block scores. We measure *score recall* as the fraction of full-attention score mass covered by the selected top-k blocks. As shown in Figure 3, it drops from 96.5% on single-page inputs to 47.7% on multi-page documents under the same sparse budget.

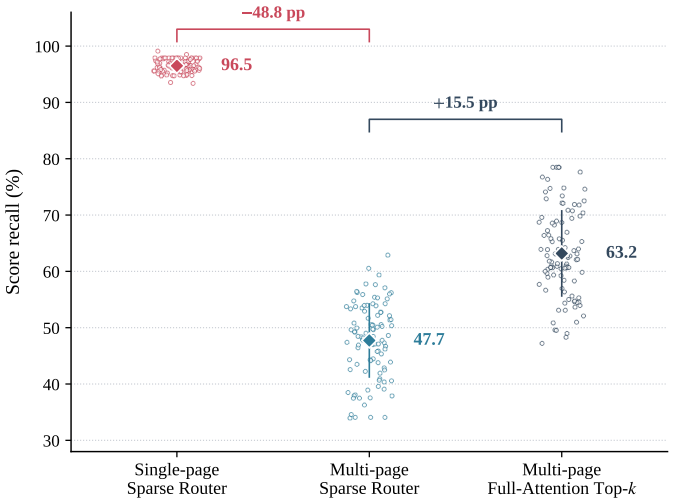


Figure 3: Sparse routing is challenging for multi-page document parsing: score recall drops from 96.5% to 47.7%, while full-attention top-k captures only 63.2% of the score mass.

We attribute this degradation to two document-specific factors. First, pages within the same document are visually homogeneous, making cross-page blocks difficult to distinguish in a global routing space. Second, full-attention scores become diffuse over long contexts: even oracle top-k selection captures only 63.2% of the total score mass. These observations motivate Page Canonicalization to reduce cross-page ambiguity and end-to-end Token Bounding to overcome diffuse attention supervision.

3 Length-Agnostic Sparse Attention

Following standard block-sparse attention Lai et al. (2026); Huang et al. (2026); Xu et al. (2026b;a), we partition $\mathcal{X}_t$ into fixed-size blocks $\{\mathcal{B}_b\}_{b=1}^{B}$ and preserve a fixed local window $\mathcal{L}(q)$ for each query q. As shown in Figure 4, LASA constructs $\mathcal{S}(q)$ hierarchically, using Page Canonicalization for page-level organization and Token Bounding for fixed-budget selection.

3.1 Page Canonicalization

Page Canonicalization decouples page-level modeling from document length by constraining the remote visual context with page structure and reparameterizing page-level positions through Page-RoPE. We use `<PAGE>` tokens to establish the correspondence between source pages and generated text segments.

Page-constrained Visual Access. Let $\mathcal{B}_b$ denote the b-th visual block. For a query q associated with page $g(q)$, we retain only blocks that contain at least one visual token from the corresponding page:

$$\mathcal{C}_{\text{page}}(q) = \left\{ b \;\middle|\; \mathcal{B}_b \cap \mathcal{V}_{g(q)} \neq \emptyset \right\}. \tag{3}$$

Thus, blocks containing no visual tokens from the query-associated page are excluded from remote visual attention. Since block boundaries are independent of page boundaries, a candidate block may straddle adjacent pages.

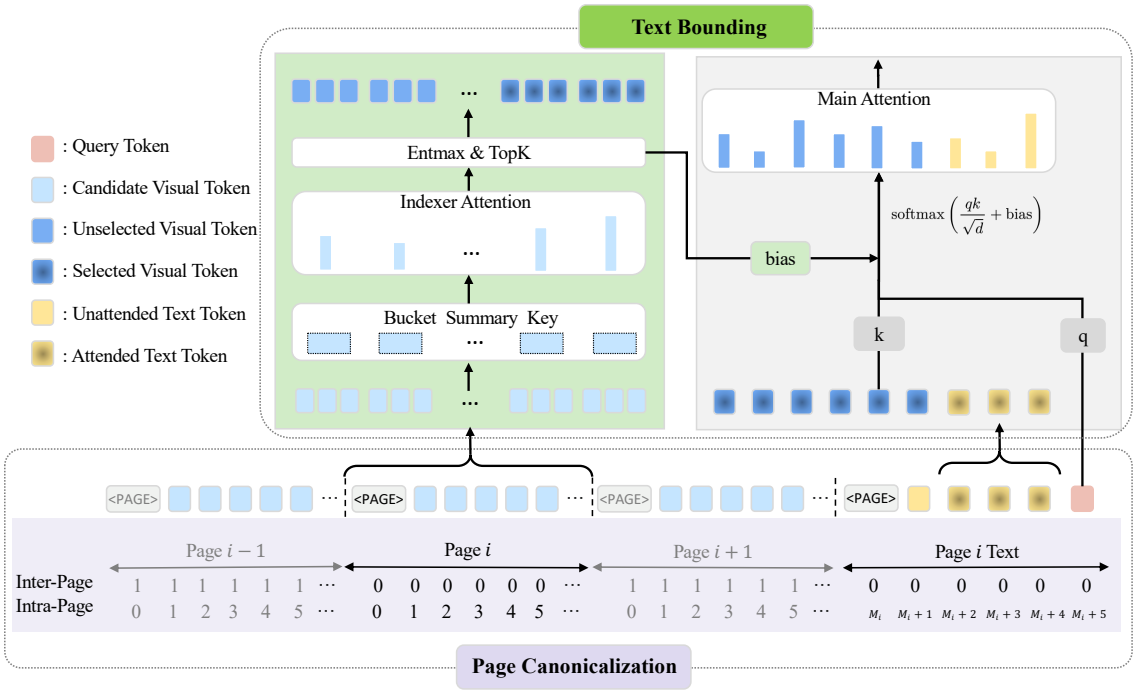


Figure 4: Overview of Length-Agnostic Sparse Attention (LASA). Page Canonicalization uses page structure and Page-RoPE to organize the visual search space, while Token Bounding selects a fixed budget of query-relevant remote blocks.

Canonicalized Page-RoPE. Beyond restricting the visual attention scope, length-agnostic page modeling also requires the positional representation to remain stable as the number of pages increases. A naive extension of 1D RoPE assigns continuously increasing positions across the entire multi-page sequence, causing the positional range to grow with document length. We instead factorize the position of each token into a *page-level coordinate* and an *intra-page coordinate*:

$$\mathbf{p}_{i,j} = \left[\mathbf{p}_i^{\text{page}},\ \mathbf{p}_{i,j}^{\text{intra}}\right]. \tag{4}$$

The page-level coordinate follows an alternating 0–1 pattern across pages, keeping the page-level positional space bounded regardless of document length. Within each page, the intra-page coordinate increases monotonically from visual tokens to the corresponding textual tokens, forming a continuous page-local positional sequence across image and text. Beyond supporting length extrapolation, this canonicalization reduces unnecessary page-dependent positional variation and encourages pages with similar local structures to share a consistent positional representation, which benefits page-local visual-text alignment and parsing accuracy. At the same time, adding more pages does not introduce unseen page-level positions, while the fine-grained ordering within each page is fully preserved.

3.2 Token Bounding

Page Canonicalization reduces the routing space in multi-page documents, but reliable fixed-budget routing still depends on how the selected blocks are learned. Conventional sparse routers typically rely on dense-attention distillation, which is not directly aligned with the final language modeling objective. Inspired by recent end-to-end routing designs Huang et al. (2026); Hu et al. (2026), we introduce an end-to-end Token-Bounded Router that directly optimizes fixed-budget block selection under the language modeling objective.

Given the page-compatible candidate set $\mathcal{C}_{\text{page}}(q)$, we summarize each candidate block b into a compact key representation $\bar{\mathbf{k}}_b$ and compute its relevance to query q. The router then forms a sparse routing distribution and retains only the top-k blocks:

$$s_b(q) = \frac{\tilde{\mathbf{q}}^\top \bar{\mathbf{k}}_b}{\sqrt{d}}, \qquad \boldsymbol{\pi}(q) = \alpha\text{-entmax}\left(\{s_b(q)\}_{b \in \mathcal{C}_{\text{page}}(q)}\right), \\ \mathcal{S}(q) = \text{TopK}\left(\boldsymbol{\pi}(q), k\right). \tag{5}$$

Here, d denotes the hidden dimension. Since $|\mathcal{S}(q)| = k$ is fixed, the amount of remote visual context accessed by each query remains bounded independently of document length. To preserve the router's relative preference among the selected blocks, we further convert the routing probabilities into a mean-centered log-prior:

$$\beta_b(q) = \log \pi_b(q) - \frac{1}{|\mathcal{S}(q)|} \sum_{b' \in \mathcal{S}(q)} \log \pi_{b'}(q), \qquad b \in \mathcal{S}(q). \tag{6}$$

The prior $\beta_b(q)$ is added to the subsequent token-level attention scores, allowing the fine-grained attention computation to inherit the block-level routing preference within the fixed selected set.

3.3 Length-Agnostic Attention Computation

With Page Canonicalization and Token Bounding, each query attends only to the fixed local window $\mathcal{L}(q)$ and at most k selected remote blocks $\mathcal{S}(q)$, regardless of the total document length. We define the retained token set as

$$\mathcal{R}(q) = \mathcal{L}(q) \cup \bigcup_{b \in \mathcal{S}(q)} \mathcal{B}_b. \tag{7}$$

The attention weights over the retained tokens are computed as

$$\alpha_j(q) = \text{Softmax}\left(\frac{q^\top k_j}{\sqrt{d}} + \mathbb{I}[b(j) \in \mathcal{S}(q)]\, \beta_{b(j)}(q)\right), \tag{8}$$

where the softmax is normalized over all tokens in $\mathcal{R}(q)$, $b(j)$ denotes the block containing token j, and k_j denotes its key projection. Tokens within the local window use the standard attention score, while tokens from selected remote blocks additionally receive the corresponding routing bias. Since the routing bias directly participates in the attention logits, gradients from the parsing objective can propagate to the routing module, enabling end-to-end optimization without an auxiliary KL-divergence distillation loss.

4 Experiments

We evaluate our LASA-based parser on MPDocBench and MarathonDocBench. Our model is initialized from the open-source UnlimitedOCR checkpoint, with LASA replacing its original attention design for scalable multi-page parsing. For MPDocBench, we follow the official evaluation protocol, while MarathonDocBench adopts the same metric definitions in the hundred-page setting. We report edit distance for text and reading order, TEDS for tables and headings, CDM for formulas, F1 for figures, and the corresponding overall score. Further implementation, training, and inference details are provided in Appendix A.

4.1 Main Results

Tables 1 and 2 compare **Ours** with general-purpose VLMs, pipeline-based parsers, and end-to-end document parsers. On MPDocBench, Ours achieves the best overall score of 83.13, surpassing PaddleOCR-VL-1.6 at 80.99 and Unlimited OCR at 61.89, while maintaining strong performance on cross-page and structural metrics. On MarathonDocBench, Ours further achieves the best overall score of 71.80, with a T-Text edit

Table 1: Performance comparison on MPDocBench.

Method	Overall↑	Text$^{\text{Edit}}$ ↓	T-Text$^{\text{Edit}}$ ↓	Formula$^{\text{CDM}}$ ↑	Table$^{\text{TEDS}}$ ↑	T-Table$^{\text{TEDS}}$ ↑	Figure$^{\text{F1}}$ ↑	ReadOrder$^{\text{Edit}}$ ↓	Heading$^{\text{TEDS}}$ ↑
General VLMs									
Gemini-3.1-Pro Preview Google DeepMind (2026a)	71.79	0.070	0.232	88.37	81.68	61.26	59.08	0.127	26.94
Qwen3.8-Max TongYi (2025)	78.30	0.057	0.196	**89.24**	**85.89**	67.18	70.19	0.097	48.80
Kimi-K3 Team et al. (2026)	60.77	0.162	0.551	41.16	75.45	81.57	40.47	0.206	39.30
Qwen3.6-plus TongYi (2025)	72.14	0.096	0.245	88.77	82.99	60.50	64.80	0.179	32.06
Qwen3-VL-235B Bai et al. (2025)	73.98	0.087	0.187	84.71	81.35	61.25	63.25	0.136	42.34
InternVL-3.5-38B Wang et al. (2025)	57.01	0.133	0.515	84.30	69.86	51.69	7.97	0.199	27.01
Pipeline-based Specialized VLMs									
GLM-OCR Duan et al. (2026)	74.70	0.060	0.314	85.25	82.41	63.10	71.90	0.125	44.79
MinerU2.5 Niu et al. (2025)	76.70	0.059	0.328	81.01	85.87	**88.14**	72.44	0.120	36.90
Youtu-Parsing Yin et al. (2026a)	74.16	0.093	0.353	87.19	84.82	63.88	71.56	0.132	43.64
MonkeyOCR-pro-3B Li et al. (2025d)	74.37	0.056	0.300	88.59	78.57	61.17	73.79	0.120	40.34
Dolphin-v2 Feng et al. (2026)	73.14	0.104	0.334	80.38	83.84	64.28	64.15	0.137	50.04
PaddleOCR-VL-1.6 Zhang et al. (2026)	80.99	**0.049**	0.147	87.94	84.60	84.27	**74.04**	0.107	47.31
End-to-end Specialized VLMs									
Unlimited OCR Yin et al. (2026b)	61.89	0.167	0.605	73.71	73.51	60.76	53.19	0.204	31.58
Ours	**83.13**	0.050	**0.125**	88.64	84.13	85.17	73.08	**0.083**	**60.00**

Table 2: Performance comparison on MarathonDocBench. Unlimited OCR is not reported on MarathonDocBench due to its 40-page input limitation.

Method	Overall↑	Text$^{\text{Edit}}$ ↓	T-Text$^{\text{Edit}}$ ↓	Formula$^{\text{CDM}}$ ↑	Table$^{\text{TEDS}}$ ↑	T-Table$^{\text{TEDS}}$ ↑	Figure$^{\text{F1}}$ ↑	ReadOrder$^{\text{Edit}}$ ↓	Heading$^{\text{TEDS}}$ ↑
General VLMs									
Gemini-3.1-Pro Preview Google DeepMind (2026a)	26.10	0.524	0.763	30.40	27.80	21.80	18.50	0.660	4.90
Qwen3.8-Max TongYi (2025)	21.89	0.606	0.804	18.90	14.94	13.78	33.38	0.740	9.03
Kimi-K3 Team et al. (2026)	31.63	0.482	0.663	40.18	26.93	26.39	21.38	0.619	**14.51**
Minimax-M3 Lai et al. (2026)	16.46	0.663	0.817	13.54	17.67	14.63	3.67	0.764	6.61
Gemini-3.8-Flash Google DeepMind (2026b)	26.34	0.529	0.658	26.27	22.41	19.64	18.36	0.690	11.83
Qwen-3.8-Flash-Next TongYi (2025)	26.25	0.550	0.753	28.09	24.93	24.09	24.07	0.685	10.88
DeepSeek-4.1-Flash Xu et al. (2026a)	11.67	0.748	0.933	10.16	12.62	12.45	6.95	0.846	3.83
GLM-5.3-Flash Z.AI	7.10	0.862	0.935	4.40	6.90	6.52	6.73	0.925	4.46
Pipeline-based Specialized VLMs									
MinerU2.5-Pro Wang et al. (2026)	67.37	0.119	0.316	88.31	**80.72**	53.32	73.12	**0.144**	1.32
TeleOCR Cai et al. (2026)	64.29	0.173	0.354	87.63	77.16	51.65	68.68	0.288	10.74
PaddleOCR-VL-1.6 Zhang et al. (2026)	69.47	**0.103**	0.308	**91.20**	78.66	55.49	**75.62**	0.174	13.32
End-to-end Specialized VLMs									
Ours	**71.80**	0.105	**0.115**	90.87	70.26	**66.23**	71.16	0.164	14.29

distance of 0.115 and a T-Table TEDS score of 66.23. In contrast, pipeline-based cross-page merging provides only limited overall improvements on long documents. These results demonstrate the effectiveness of end-to-end cross-page modeling for long-document parsing.

4.2 Length-Agnostic Efficiency

Figure 5 examines how decoding efficiency scales with document length. All measurements are conducted in Gundam mode with a batch size of 3 and a fixed output length of 1,024 tokens. As the number of pages increases, Full Attention incurs rapidly growing per-step latency because each query attends to an expanding visual context. R-SWA reduces this cost by bounding local attention, but its remote visual scope still grows with document length. In contrast, LASA keeps the effective per-query attention scope bounded, resulting in nearly constant per-step decoding latency from short to hundred-page documents. Consequently, its efficiency advantage over Full Attention becomes increasingly pronounced as documents grow longer.

4.3 Ablation Studies

To isolate the contribution of each component to parsing accuracy and sparse visual access, we conduct controlled ablation studies on MPDocBench using approximately 200K multi-page training samples under a 32K context window. This reduced setup is used only for ablation studies, while the final model is trained with the full training configuration described in Appendix A. All variants are trained under the same setting for a fair comparison.

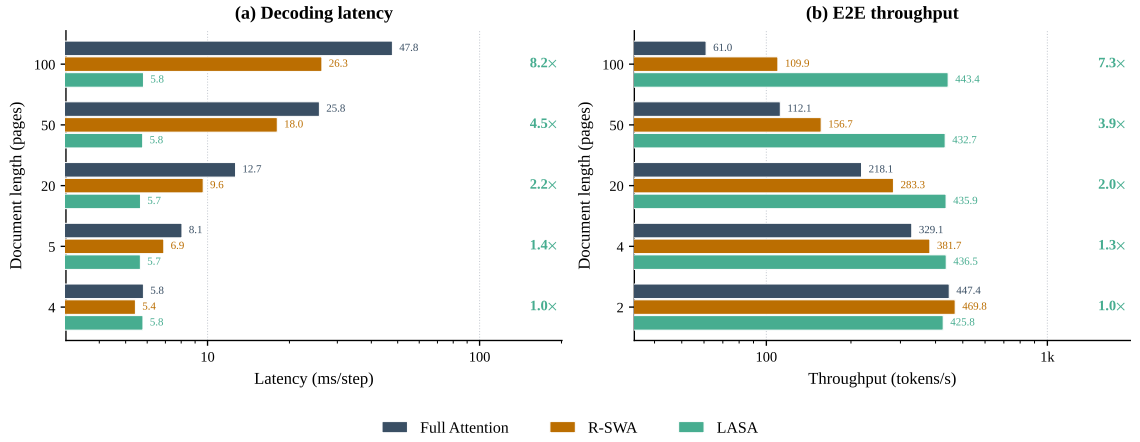


Figure 5: Length-agnostic efficiency scaling. LASA maintains nearly constant decoding latency and throughput as document length increases.

Table 3: Ablation studies of the core components of LASA. The gray row denotes the original open-source UOCR-1.0 checkpoint, while all subsequent variants are trained on the same 200K training set. Visual Context (%) denotes the fraction of visual tokens accessed during decoding.

Page-Constrained Visual Access	Page-RoPE	Token-Bounded Router	Score ↑	Δ	Visual Context (%) ↓
–	–	–	61.89	–	100
–	–	–	64.00	+2.11	100
–	–	MSA	57.28	-6.72	8
✓	–	MSA	59.17	+1.89	8
✓	✓	MSA	65.39	+6.22	8
✓	✓	Ours	**69.00**	+3.61	**8**

These results show that simply introducing sparse attention leads to unsatisfactory performance in multi-page parsing. By reducing cross-page ambiguity and optimizing the router end-to-end, rather than merely fitting the full-attention distribution, our design substantially improves parsing accuracy while maintaining the same level of sparsity. This provides a practical foundation for scaling document parsing to substantially longer inputs.

5 Related Work

5.1 Document Parsing and Multi-page Document VLMs

Document parsing has increasingly been reshaped by advances in vision-language models. Existing methods broadly follow either pipeline-based designs Cui et al. (2025); Zhang et al. (2026); Cui et al. (2026); Niu et al. (2025); Wang et al. (2026); Li et al. (2025c); Duan et al. (2026) that combine specialized page analysis modules with VLM-based decoding, or unified VLM-based parsers Li et al. (2025b); Dong et al. (2026); Wei et al. (2026); Lu et al. (2026); Wu et al. (2026); Li et al. (2026) that directly generate structured representations from page images. Despite substantial progress, these methods remain limited to within-page modeling and cannot effectively capture structures and semantic dependencies that span multiple pages. Recent work such as MinerU-Popo Xu et al. (2026c) performs cross-page reasoning over page-level OCR outputs to re-

cover document-level structures. However, this two-stage design can propagate upstream parsing errors and loses visual evidence that may be necessary to resolve ambiguous cross-page relationships.

To overcome this limitation, recent end-to-end multi-page VLMs jointly model visual content from multiple pages within a shared context, allowing cross-page evidence to be directly accessed during parsing. UnlimitedOCR Yin et al. (2026b) improves long-document efficiency with R-SWA, but still retains dense access to remote visual tokens, leaving the remote attention scope dependent on document length. However, as document length increases, the visual context accessible to each decoding query continues to expand, creating a fundamental scalability bottleneck for hundred-page parsing. Our work differs by explicitly bounding remote visual access, targeting attention scalability as document length grows.

5.2 Efficient Long-Context Modeling

Efficient long-context modeling has been explored through both attention replacement and sparse attention. Linear or recurrent alternatives Gu & Dao (2023); Yang et al. (2023); Katharopoulos et al. (2020); Yang et al. (2024); Li et al. (2025a) reduce long-context cost by summarizing historical information into compact states. While effective for general sequence modeling, such recurrent compression may be less desirable for document parsing, where preserving fine-grained visual features is important for accurate recognition and cross-page retrieval.

Sparse attention retains direct access to selected historical tokens or KV blocks while avoiding dense all-to-all computation. Early methods use predefined patterns such as local windows and global tokens Beltagy et al. (2020); Zaheer et al. (2020), whereas recent approaches adopt query-dependent retrieval. Hard-selection methods, such as Compressed Sparse Attention(CSA) Xu et al. (2026b) and MiniMax Sparse Attention(MSA) Lai et al. (2026), rely on discrete top-k routing and typically require auxiliary supervision or warmup. In contrast, DashAttention Huang et al. (2026) and HiLS Hu et al. (2026) incorporate differentiable or forward-integrated routing scores, enabling more direct end-to-end optimization.

6 Conclusion

We study ultra-long document parsing at the scale of tens to hundreds of pages, where preserving cross-page dependencies must be reconciled with the rapidly growing cost and search space of end-to-end modeling. Conventional sparse attention retains a large remote search space, making KL-distilled routing less reliable for relevant block selection. LASA addresses this issue by exploiting the hierarchical structure of multi-page documents and directly optimizing fixed-budget selection for the final prediction objective. Together with MarathonDocBench, this enables systematic evaluation of both parsing quality and scalability in the hundred-page regime. Across MPDocBench and MarathonDocBench, LASA achieves strong document-level parsing performance while keeping per-step decoding cost nearly insensitive to document length. These results demonstrate that hierarchical sparse attention provides a practical path toward scalable end-to-end parsing of increasingly long documents.

7 Limitations and Future Work

We develop LASA into an end-to-end long-document parser that combines efficient sparse visual access with strong parsing performance. However, the current training corpus remains relatively limited in both scale and diversity compared with the wide variety of real-world multi-page documents. Future work will focus on constructing a larger and higher-quality training corpus with broader coverage of document domains, layouts, lengths, and cross-page structures. Such improvements may further enhance robustness and generalization to substantially longer and more structurally complex documents.

AI USE STATEMENT

Generative AI tools were used to assist with language polishing, improving readability, and limited code editing during the preparation of this manuscript. They were not used to generate the core research ideas, methodology, experimental design, or scientific conclusions. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide the methodological, training, data-processing, and implementation details required for reproduction across the main paper and appendix, together with additional hyperparameters and extended results in the appendix. The source code and configurations will be released upon acceptance.

ETHICS STATEMENT

This work does not involve human-subject experiments or the collection of sensitive personal information. The training data are constructed from web-crawled and in-house document sources and are processed according to applicable data-use and privacy requirements. We use the data solely for document parsing research and take care to avoid exposing private or personally identifiable information.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Peng Cai, Zhaofan Zou, Shifa Liu, Yikun Wang, Jiawei Tang, Kaicheng Yang, Meng Tong, MingKun Jiang, Zhongjiang He, and Hao Sun. Teleocr: Navigating document parsing across digital and camera-captured documents. *arXiv preprint arXiv:2608.12898*, 2026. URL https://arxiv.org/abs/2608.12898.

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 0.9 b ultra-compact vision-language model. *arXiv preprint arXiv:2510.14528*, 2025.

Cheng Cui, Ting Sun, Suyin Liang, Tingquan Gao, Zelun Zhang, Jiaxuan Liu, Xueqing Wang, Changda Zhou, Hongen Liu, Manhui Lin, et al. Paddleocr-vl-1.5: Towards a multi-task 0.9 b vlm for robust in-the-wild document parsing. *arXiv preprint arXiv:2601.21957*, 2026.

Daxiang Dong, Mingming Zheng, Dong Xu, Chunhua Luo, Bairong Zhuang, Yuxuan Li, Ruoyun He, Haoran Wang, Wenyu Zhang, Wenbo Wang, et al. Qianfan-ocr: A unified end-to-end model for document intelligence. *arXiv preprint arXiv:2603.13398*, 2026.

Shuaiqi Duan, Yadong Xue, Weihan Wang, Zhe Su, Huan Liu, Sheng Yang, Guobing Gan, Guo Wang, Zihan Wang, Shengdong Yan, et al. Glm-ocr technical report. *arXiv preprint arXiv:2603.10910*, 2026.

Hao Feng, Wei Shi, Ke Zhang, Xiang Fei, Lei Liao, Dingkang Yang, Yongkun Du, Xuecheng Wu, Jingqun Tang, Yang Liu, et al. Dolphin-v2: Universal document parsing via scalable anchor prompting. *arXiv preprint arXiv:2602.05384*, 2026.

Google DeepMind. Gemini 3.1. `https://deepmind.google/models/gemini/pro/`, 2026a.

Google DeepMind. Gemini 3.8. `https://deepmind.google/models/gemini/flash/`, 2026b.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Xiang Hu, Xinyu Wei, Hao Gu, Minshen Zhang, Tian Liang, Huayang Li, Lei Zhu, Yan Wang, Sirui Han, Yushi Bai, et al. Hierarchical sparse attention done right: Toward infinite context modeling. *arXiv preprint arXiv:2607.02980*, 2026.

Yuxiang Huang, Nuno MT Gonçalves, Federico Alvetreti, Lei Li, Xu Han, Edoardo M Ponti, André FT Martins, and Marcos V Treviso. Dashattention: Differentiable and adaptive sparse hierarchical attention. *arXiv preprint arXiv:2605.18753*, 2026.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pp. 5156–5165. PMLR, 2020.

Xunhao Lai, Weiqi Xu, Yufeng Yang, Qiaorui Chen, Yang Xu, Lunbin Zeng, Xiaolong Li, Haohai Sun, Haichao Zhu, Vito Zhang, et al. Minimax sparse attention. *arXiv preprint arXiv:2606.13392*, 2026.

Aonian Li, Bangwei Gong, Bo Yang, Boji Shan, Chang Liu, Cheng Zhu, Chunhao Zhang, Congchao Guo, Da Chen, Dong Li, et al. Minimax-01: Scaling foundation models with lightning attention. *arXiv preprint arXiv:2501.08313*, 2025a.

Gengluo Li, Xingyu Wan, Shangpin Peng, Weinong Wang, Hao Feng, Yongkun Du, Binghong Wu, Zheng Ruan, Zhiqiong Lu, Liang Wu, et al. Hunyuanocr-1.5: Making lightweight ocr vlms faster and better. *arXiv preprint arXiv:2607.04884*, 2026.

Yumeng Li, Guang Yang, Hao Liu, Bowen Wang, and Colin Zhang. dots. ocr: Multilingual document layout parsing in a single vision-language model. *arXiv preprint arXiv:2512.02498*, 2025b.

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*, 2025c.

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

Shiyin Lu, Yinglun Li, Yu Xia, Yuhui Chen, An-Yang Ji, Jun-Peng Jiang, Qing-Guo Chen, Jianshan Zhao, En Lin, Haijun Li, et al. Ovisocr2 technical report. *arXiv preprint arXiv:2607.13639*, 2026.

Junbo Niu, Zheng Liu, Zhuangcheng Gu, Bin Wang, Linke Ouyang, Zhiyuan Zhao, Tao Chu, Tianyao He, Fan Wu, Qintong Zhang, et al. MinerU2.5: A decoupled vision-language model for efficient high-resolution document parsing. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics–Industry Track*, 2025.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, SH Cai, Yuan Cao, Y Charles, HS Che, Cheng Chen, Guanduo Chen, et al. Kimi k2. 5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*, 2026.

TongYi. TongYi. https://www.aliyun.com/product/tongyi, 2025.

Bin Wang, Tianyao He, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Tao Chu, Yuan Qu, Zhenjiang Jin, Weijun Zeng, Ziyang Miao, et al. Mineru2. 5-pro: Pushing the limits of data-centric document parsing at scale. *arXiv preprint arXiv:2604.04771*, 2026.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Shu Wei, Jingjing Wu, Lingshu Zhang, Qunyi Xie, Hao Zou, Le Xiang, Xu Fan, Yangliu Xu, Manhui Lin, Xiaolong Ma, et al. Hpd-parsing: Hierarchical parallel document parsing. *arXiv preprint arXiv:2607.18839*, 2026.

Hao Wu, Haoran Lou, Xinyue Li, Zuodong Zhong, Zhaojun Sun, Phellon Chen, Xuanhe Zhou, Kai Zuo, Yibo Chen, Xu Tang, et al. Firered-ocr technical report. *arXiv preprint arXiv:2603.01840*, 2026.

Anyi Xu, B Li, Bangcai Lin, Bing Xue, BingCheng Xian, Bingzheng Xu, Bochao Wu, Bowei Zhang, Boyi Deng, CC Yu, et al. Deepseek-v4. 1-flash: Pushing the limits of kv cache compression. *arXiv preprint arXiv:2609.19969*, 2026a.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026b.

Bangrui Xu, Ziyang Miao, Xuanhe Zhou, Yiming Lin, Zirui Tang, Xiaomeng Zhao, Fan Wu, Cheng Tan, Bin Wang, and Conghui He. Mineru-popo: Universal post-processing model for structured document parsing. *arXiv preprint arXiv:2605.24973*, 2026c.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2023.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. *Advances in neural information processing systems*, 37:115491–115522, 2024.

Kun Yin, Yunfei Wu, Bing Liu, Zhongpeng Cai, Xiaotian Li, Huang Chen, Xin Li, Haoyu Cao, Yinsong Liu, Deqiang Jiang, et al. Youtu-parsing: Perception, structuring and recognition via high-parallelism decoding. *arXiv preprint arXiv:2601.20430*, 2026a.

Youyang Yin, Huanhuan Liu, Qunyi Xie, Chaorun Liu, Shiqi Yang, Shaohua Wang, Zhanlong Liu, Hao Zou, Jinyue Chen, Shu Wei, et al. Unlimited ocr works. *arXiv preprint arXiv:2606.23050*, 2026b.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020.

Z.AI. GLM-5.3-Flash. https://huggingface.co/zai-org/GLM-5.3-Flash. Model card, accessed 2026-09-25.

Zelun Zhang, Hongen Liu, Suyin Liang, Yubo Zhang, Yiqing Xiang, Jiaxuan Liu, Ting Sun, Manhui Lin, Yue Zhang, Changda Zhou, et al. Paddleocr-vl-1.6: Expanding the frontier of document parsing with under-optimized region refinement and progressive post-training. *arXiv preprint arXiv:2606.03264*, 2026.

Bangbang Zhou, Hangdi Xing, Yifan Chen, Jianjun Xu, Qi Zheng, Feiyu Gao, Zhibo Yang, Shuai Bai, Ming Yan, Jieping Ye, et al. Mpdocbench-parse: Benchmarking practical multi-page document parsing. *arXiv preprint arXiv:2605.22100*, 2026.

A Training and Implementation Details

A.1 Training Data Construction

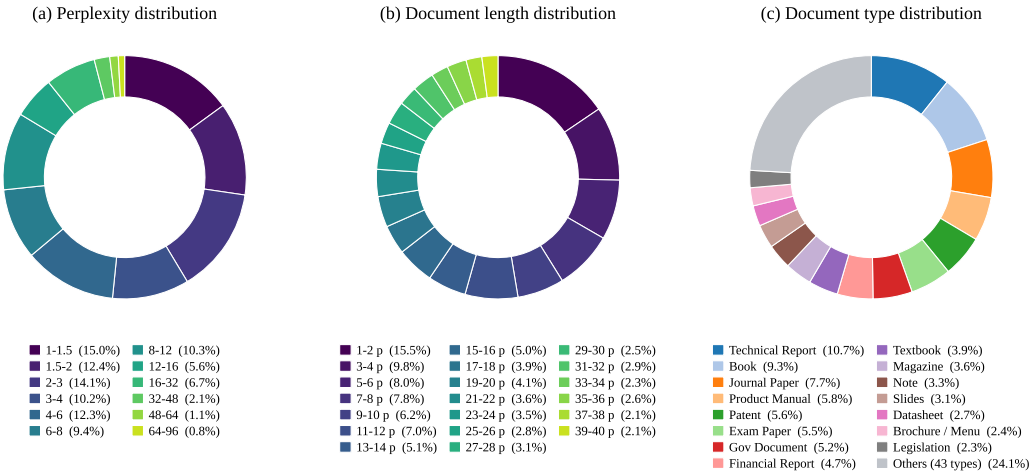


Figure 6: Distribution of the retained multi-page training corpus across perplexity, document length, and document type.

We construct the training corpus from a large collection of multi-page documents obtained from web-crawled and in-house sources. Starting from an initial pool of approximately 1.4 million documents, we generate pseudo-labels through a multi-stage annotation, filtering, and sampling pipeline.

Pseudo-label Generation. We first apply PaddleOCR-VL-1.6 to obtain page-level parsing results for each document. The page-wise outputs are then assembled into document-level annotations using the cross-page and cross-column merging procedure of MinerU Popo. We additionally apply supplementary merging strategies to refine incomplete or inconsistent structures across adjacent pages. The resulting annotations provide scalable supervision for both page-local content and cross-page structures.

Source Annotation. To characterize the heterogeneous composition of the initial data pool, we use a VLM to assign source and scenario labels to each document. These annotations are subsequently used to control the composition of the final training set and maintain broad coverage across document domains.

Quality Filtering. We use the open-source UnlimitedOCR model to compute perplexity over the pseudo-labeled sequences as a model-based signal of sample difficulty and learnability. We filter out abnormally high-perplexity samples and downsample overly easy, redundant examples. Additional heuristic rules remove malformed cases such as repetitive or corrupted outputs.

Distribution-Balanced Sampling. After filtering, we perform balanced sampling over document type, language, perplexity, and page-count distributions. As illustrated in Figure 6, the resulting corpus covers a broad range of perplexity levels, document lengths, and document categories, avoiding excessive concentration in dominant sources or easy examples. This procedure yields approximately 400K multi-page training samples.

The pseudo-labels are used only as scalable training supervision and are independent of the benchmark annotations used for evaluation.

A.2 Training Configuration

Our model is initialized from the open-source UnlimitedOCR checkpoint. Unlike the original UnlimitedOCR multi-page setting, which uses the base representation, our implementation enables Gundam mode for both single-page and multi-page inputs.

We perform supervised fine-tuning for one epoch using Adam with an initial learning rate of 1×10^{-4} and cosine decay. The maximum training context length is 64K tokens. All training experiments are conducted on NVIDIA A800 80GB GPUs.

A.3 Sparse Attention Configuration

Following the configuration used throughout the main experiments, we set the block size to 32. For each decoding query, Token Bounding selects $k = 12$ remote visual blocks, while 4 additional blocks are reserved for the local attention window. The local window therefore contains 128 tokens, matching the local context size used by UnlimitedOCR.

For Page-RoPE, we use the page-level positional encoding described in Section 3.1. Unless otherwise specified, the same sparse-attention configuration is used across all accuracy and efficiency experiments.

A.4 Inference and Evaluation Setup

LASA is implemented in the vLLM inference framework. At inference time, we use a maximum context length of 128K tokens for MPDocBench and 768K tokens for MarathonDocBench, reflecting their substantially different document-length regimes. Unless otherwise specified, all compared methods are evaluated under the same inference framework and hardware setting.

We evaluate our LASA-based parser on MPDocBench and MarathonDocBench using the same metric definitions. We report edit distance for text and reading order, TEDS for tables and headings, CDM for formulas, F1 for figures, and the corresponding overall score. All inference experiments are conducted on NVIDIA A800 80GB GPUs.

B Bounded KV-Cache via Sliding-Window Streaming

Although LASA bounds the attention scope of each decoding query, extremely long documents may still exceed the available context window during prefill. We therefore adopt a sliding-window streaming strategy. Given a window width W and stride S, each step retains the most recent $W - S$ pages together with their parsed text, introduces only the next S pages, and continues decoding from the retained prefix. This bounds the active context while preserving local cross-window continuity. Algorithm 1 summarizes the procedure.

The memory benefit follows directly from bounding the active page window. Full-context processing retains visual tokens from all N pages, causing the KV-cache footprint to grow approximately linearly with document length. In contrast, sliding-window streaming keeps only a fixed window of W pages active, so the peak KV-cache scales with W rather than N. As illustrated in Figure 7, this keeps memory usage bounded even as document length increases. Table 4 further shows that this reduction in peak memory does not degrade parsing quality.

Algorithm 1 Sliding-Window Streaming for Unlimited Document Parsing

Require: Document pages $\{I_1, \ldots, I_D\}$, window width W, stride S
Ensure: Concatenated document transcription

$\mathcal{O} \leftarrow [\,]$
$e \leftarrow \min(W, D)$
$\mathcal{P} \leftarrow [I_1, \ldots, I_e]$
$\mathcal{T} \leftarrow \textsc{Decode}(\textsc{Prefill}(\mathcal{P}))$
$\mathcal{O}.\mathrm{append}(\mathcal{T})$
while $e < D$ **do**
  $(\mathcal{P}, \mathcal{T}) \leftarrow \textsc{RetainLastPages}(\mathcal{P}, \mathcal{T}, W - S)$
  $s \leftarrow e + 1$
  $e_{\text{new}} \leftarrow \min(e + S, D)$
  $\mathcal{P}_{\text{new}} \leftarrow [I_s, \ldots, I_{e_{\text{new}}}]$
  $\mathcal{T}_{\text{new}} \leftarrow \textsc{Decode}\Big(\textsc{Prefill}(\mathcal{P}_{\text{new}}; \mathcal{P}, \mathcal{T})\Big)$
  $\mathcal{P} \leftarrow \mathcal{P} \,\|\, \mathcal{P}_{\text{new}}$
  $\mathcal{T} \leftarrow \mathcal{T} \,\|\, \mathcal{T}_{\text{new}}$
  $\mathcal{O}.\mathrm{append}(\mathcal{T}_{\text{new}})$
  $e \leftarrow e_{\text{new}}$
end while
return $\textsc{Concat}(\mathcal{O})$

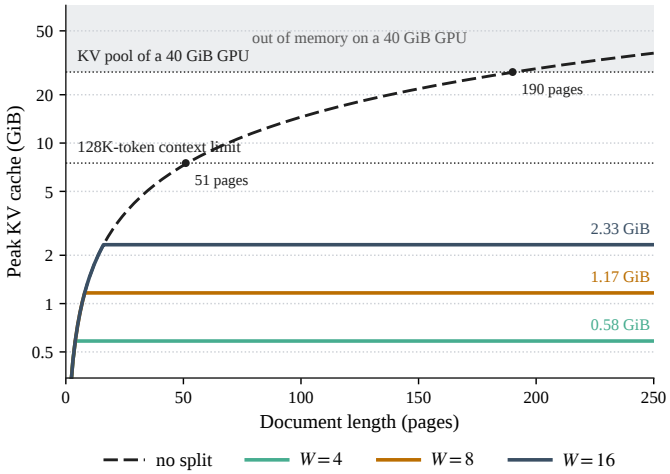


Figure 7: Peak KV-cache usage under full-context processing and bounded sliding-window streaming with different window sizes.

Table 4: Effect of bounded KV-cache streaming on parsing quality and peak KV-cache usage.

Setting	Overall ↑	Peak Visual KV Cache ↓	Scaling
w/o Streaming	83.16	$\mathcal{O}(D)$	Document length
w/ Streaming	83.18	$\mathcal{O}(W)$	Window width