A Simple yet Effective Layout Token in Large Language Models for Document Understanding

Zhaoqing Zhu[1*], Chuwei Luo[1*†], Zirui Shao[2*], Feiyu Gao[1†], Hangdi Xing[2], Qi Zheng[1]
Ji Zhang[1]
[1]Alibaba Group, [2]Zhejiang University
{zzhaoqing.z,luochuwei,zhengqisjtu}@gmail.com
{shaozirui,xinghd}@zju.edu.cn, {feiyu.gfy,zj122146}@alibaba-inc.com

Abstract

Recent methods that integrate spatial layouts with text for document understanding in large language models (LLMs) have shown promising results. A commonly used method is to represent layout information as text tokens and interleave them with text content as inputs to the LLMs. However, such a method still demonstrates limitations, as it requires additional position IDs for tokens that are used to represent layout information. Due to the constraint on max position IDs, assigning them to layout information reduces those available for text content, reducing the capacity for the model to learn from the text during training, while also introducing a large number of potentially untrained position IDs during long-context inference, which can hinder performance on document understanding tasks. To address these issues, we propose LayTokenLLM, a simple yet effective method for document understanding. LayTokenLLM represents layout information as a single token per text segment and uses a specialized positional encoding scheme. It shares position IDs between text and layout tokens, eliminating the need for additional position IDs. This design maintains the model's capacity to learn from text while mitigating long-context issues during inference. Furthermore, a novel pre-training objective called Next Interleaved Text and Layout Token Prediction (NTLP) is devised to enhance cross-modality learning between text and layout tokens. Extensive experiments show that LayTokenLLM outperforms existing layout-integrated LLMs and MLLMs of similar scales on multi-page document understanding tasks, as well as most single-page tasks.

1. Introduction

Document understanding [9] is currently an important area in both industry and academic research, driven by the critical need to efficiently process and understand complex documents. In recent years, large language models (LLMs) and multimodal large language models (MLLMs) have made remarkable progress in this field. Especially in tasks involving rich textual content, such as the document-oriented Visual Question Answering (DocVQA) [32] task and the visually-rich document information extraction (VIE) [19, 33, 48] task. Some works [20, 28, 44] suggest that the most valuable information for document understanding can be derived from both the text and its layout, treating spatial layouts as a form of lightweight visual information. Building on this idea, these approaches [20, 28, 44] that integrate such spatial layouts as visual information with text for LLMs have shown promising results, and sometimes superior performance compared to MLLMs.

The integration of layout information can be broadly cat-

*Equal contribution.† Corresponding author.

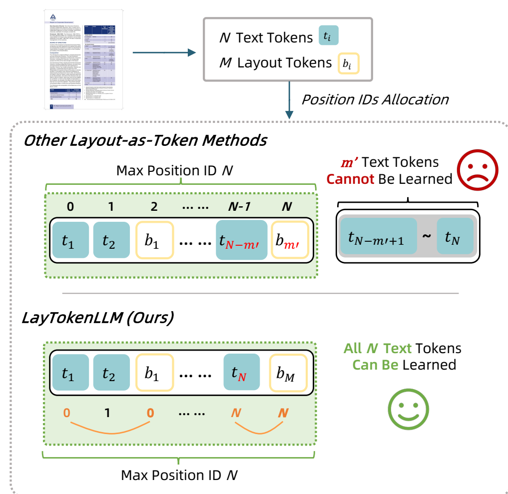

Figure 1. Comparison with other Layout-as-Token methods. Previous Layout-as-Token methods require additional position IDs for layout information which squeeze the learning space for text content, while LayTokenLLM eliminates the need for additional position IDs of layout information by sharing the first position ID of corresponding text content.

Text and Layout Format	Number of Extra Position IDs for Representing Layout		T-Ratio (N_t / N)
	In a Segment	Avg. on MP-DocVQA	
Plain Text (w/o Layout)	0	0	100%
{text:"text",Box:[123, 456, 133, 500]} [13]	27	8015	27.02%
<ref>text</ref><box>(123,456),(133,500)</box> [3, 27]	21	7959	32.26%
<ref>text</ref><box>[[253, 231, 733, 787]]</box> [6]	18	6894	35.71%
text+one box hidden representation [28]	1	350	90.91%
text+layout_token (**Ours**)	**0**	**0**	**100%**

Table 1. Comparison of different paradigm to integrate layout information with text content. T-Ratio is defined as the ratio of the position utilization for text tokens (N_t) to the maximum trained position length (N). In the table, N is set to 2048.

egorized into two types: layout-as-modality methods [31, 44] and layout-as-tokens methods [13, 20, 28, 35]. Layout-as-modality methods treat layout information as an additional modality [31, 44], modeling it alongside text within LLMs and training specialized LLMs. Although layout-as-modality methods have shown good performance, they require substantial modifications to the LLM architecture to incorporate layout information, making them less lightweight. On the other hand, layout-as-tokens methods represent the layout information as text tokens, and interleave them with the corresponding text content as inputs into the LLMs, which provide a more lightweight and commonly used approach [20, 28] for document understanding.

However, existing layout-as-token methods still encounter a significant limitation. Due to the constraint on max position IDs, assigning them to layout information reduces those available for text content, reducing the capacity for the model to learn from the text during training. As illustrated in Fig. 1, the context window during training is constrained by the maximum position ID N. When tokens that are used for representing layout information are integrated into an LLM, the allocation of additional position IDs (m') for layout information reduces the number of position IDs available for the text content ($N - m'$), leading to less capacity for LLMs to learn from the text during training. To better quantify the impact of incorporating additional layout information, a rough measure called T-Ratio which is used to represent the ratio of the position utilization for text tokens (N_t) to the maximum trained position length (N) is shown in Tab. 1. As can be seen, assigning position IDs to tokens that are used for representing layout information significantly affects the T-ratio, even when only a single position ID is used to represent the layout in a segment. Although the T-ratio is a rough measure, it reflects the impact of introducing layout information, as assigning position IDs to layout tokens reduces the number of position IDs available for text content, ultimately limiting the model's capacity to learn from the text during training. And it can be also seen that existing methods allocate hundreds or even thousands of additional position IDs for layout information on the MP-DocVQA dataset and these additional position IDs are potentially unlearned during training, which may exacerbate the long-context inference problem [34, 37].

To address these issues, we propose a simple yet effective framework in this paper, called LayTokenLLM, for document understanding. LayTokenLLM is a lightweight framework that represents the layout information of each text segment as a single layout token and employs a specially designed positional encoding scheme for the layout tokens. Notably, as shown in Tab. 1, LayTokenLLM incorporates layout information as a single layout token but without allocating any additional position ID, ensuring comprehensive learning of text content (100% max position IDs for text content) during training, while alleviating long-context issues introduced by layout information during inference. Additionally, a novel pre-training objective called Next Interleaved Text and Layout Token Prediction (NTLP) is proposed to improve the comprehension of interleaved format and deepen the connection between these distinct types of information in LayTokenLLM. Different from previous methods that focus solely on either text or layout content for subsequent predictions [28, 44], NTLP leverages the autoregressive traits of LLMs and additionally facilitates cross-prediction between text and layout. Extensive experiments across widely used benchmarks for both single-page and multi-page document understanding demonstrate the effectiveness of the proposed LayTokenLLM.

Our contributions are summarized as follows:

1) This paper introduces LayTokenLLM, a simple yet effective method to integrate layout information into LLMs for document understanding. It represents layout information as a single token and uses a specially designed positional encoding scheme, avoiding the issues caused by allocating additional position IDs for the layout information.
2) A novel pre-training objective called Next Interleaved Text and Layout Token Prediction is introduced to enhance cross-modal prediction and relational learning between text and layout modalities.
3) Experimental results show that the proposed LayTokenLLM significantly outperforms existing methods utilizing LLMs/MLLMs for multi-page document understanding, while also achieving superior performance in most subtasks of single-page document comprehension.

2. Related Work

Recently, leveraging large language models (LLMs) and multimodal large language models (MLLMs) for document understanding have shown significant progress. Although existing MLLMs show promising results in document understanding, they still struggle with issues associated with high-resolution input, particularly in cases of dense or difficult-to-recognize text. Considering the layout information is vital for document understanding [2, 10, 12, 17, 22, 25, 29, 30, 36, 45, 47, 50], existing an alternative approach, integrating spatial layouts with text as lightweight visual information for LLMs has shown promising results, and sometimes even superior performance compared to MLLMs. These approaches can be categorized into two types: layout-as-modality methods and layout-as-tokens methods.

2.1. Multi-modal Large Language Models

Existing MLLMs [1, 4, 8, 26, 39, 49, 52] show exceptional performance for document understanding. Models [4, 7, 8, 23, 51] like InternVL, Qwen-VL have augmented MLLMs with advanced visual capabilities by introducing high-resolution visual input to better handle documents containing dense or difficult-to-recognize text. However, the methods require an excessive number of image tokens, adversely affecting inference speed [7, 23, 51]. In response to this challenge, a series of MLLMs [14, 15, 24] propose to reduce the token count by compressing image patches, but this may lead to the loss of critical textual information.

2.2. Layout-as-Modality Methods

Layout-as-Modality methods treat layout information as an additional modality, modeling it alongside text within LLMs [5, 40, 42] and training specialized LLMs for document understanding. Luo et al. [31] make pioneering attempts to combine layout information with text into LLMs. In order to fully exploit the document layout information, it employs pre-trained document encoders, which represent the spatial layout of text as an additional modality similar to previous pre-trained text-layout models [18, 46]. Recently, Wang et al. [44] propose to further disentangle the text and layout modalities by considering the inter-dependency between them. However, these methods require modifying the model architecture and necessitate an additional pre-training stage, making them less lightweight.

2.3. Layout-as-Token Methods

Layout-as-Tokens methods represent layout information as text sequences, embedding the sequences interleaved with the corresponding text as inputs into the LLMs as shown in Tab. 1, providing a more natural and commonly used approach. Specifically, He et al. [13] introduce an in-context learning format like "{*text:"text",Box:[123, 456, 133, 500]*}" which incorporate layout information (See Tab. 1, line 2) in the demonstration to enable LLMs to understand positional relationships. And Lamott et al. [20] design a novel document verbalizer to effectively encode the layout information in the prompt. Perot et al. [35] generate LLM prompts containing both the text content and coordinate tokens, which communicate the layout modality and act as unique identifiers of the text segments for information extraction and localization, while Lu et al. [28] use one hidden token to represent layout information. Despite their convenience and effectiveness, these methods introduce an excessive number of interleaved position spaces to represent the layout, leading to a dilution of the textual content (see Tab. 1). The extra interleaved position occupied by models not only hampers the comprehension learning and increases the burden of comprehension of the text content.

3. Method

In this section, our LayTokenLLM is presented, which is an LLM-based method that incorporates text and spatial layout information which can be viewed as lightweight visual information. To incorporate layout information while avoiding issues arising from extra position ID allocations, and to enhance the connection between text and layout within the same segment, two primary components are proposed: a simple yet effective Layout Token, and a pre-training objective designed for interleaved text and layout format.

3.1. Model Architecture

The overall architecture of LayTokenLLM is shown in Fig. 2. Once the text segments with corresponding layout information are parsed from the document (e.g., by OCR), the bounding box coordinates of each text segment are first compressed into one single layout token with a layout tokenizer. Then the text tokens and their corresponding layout tokens are interleaved and input to LLM. A simple yet effective layout positional encoding scheme is designed to address the issues of additional position IDs. Furthermore, a novel pretraining objective is proposed to enhance cross-modal connections within the same segment.

3.1.1 Details of Layout Token

As shown in the upper left part of Fig. 2, a learnable embedding $t \in \mathbb{R}^d$ is employed as a query, mapping each text segment's bounding box Box into only one single layout token $b \in \mathbb{R}^d$:

$$b = F_{Attn}(t, F_B(Box)), \tag{1}$$

where F_B represents a projector that encodes the bounding box defined by four-dimensional coordinates $[x1, y1, x2, y2]$ into a high-dimensional embedding, and

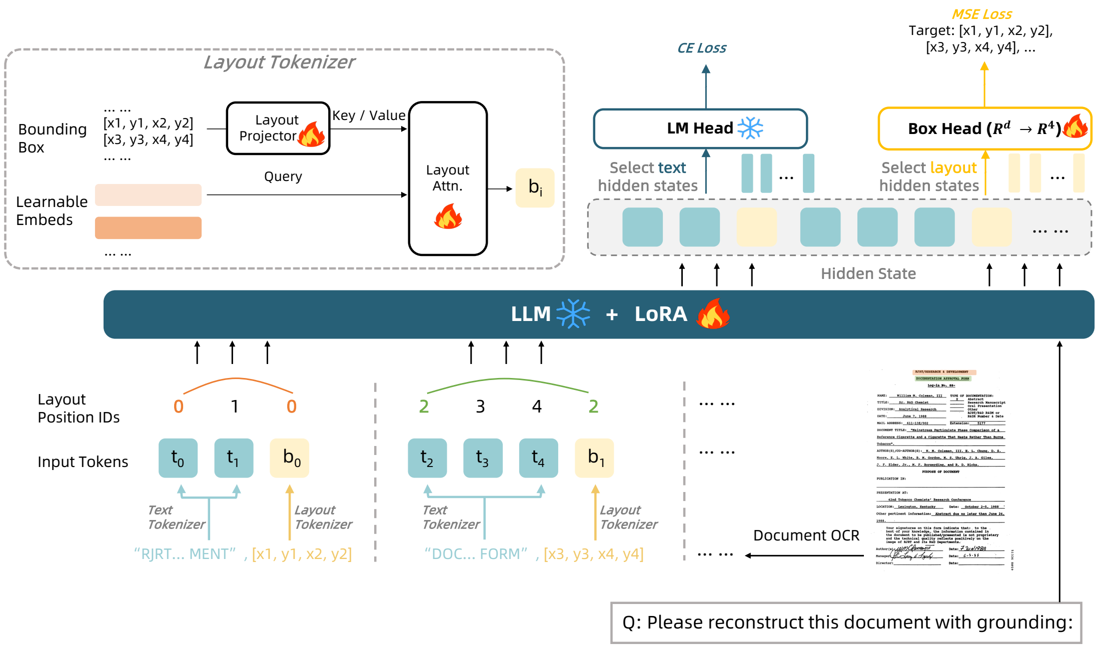

Figure 2. The overall architecture of LayTokenLLM. Given the text segments with layouts parsed from document (e.g., by OCR), LayTokenLLM first tokenizes the layout information (bounding box) of each text segment into a single layout token by leveraging a trainable projector and an attention module with learnable query. Subsequently, the text tokens and layout tokens are interleaved and the position IDs are assigned by sharing the first position ID of each text segment with the corresponding layout token, preserving the entire learning space for textual content. Finally, distinct training objectives are employed for the text and layout information, respectively.

F_{Attn} represents an attention encoder which takes the learnable embedding as query and the high-dimensional embedding of bounding box as key and value. Through the layout tokenizer, the layout information is significantly compressed, thereby alleviating the burden of longer tokens while enhancing inference speed.

3.1.2 Positional Encoding Scheme for Layout Token

The most prevalent positional encoding method for LLMs is the Rotary Positional Encoding (RoPE) [38]. Let T and L represent the length of tokens used for text and layout information in an OCR segment, previous methods will allocate additional position IDs for the interleaved layout information and set the position IDs of a segment P as:

$$P = [0, 1, \ldots, T-1, T, \ldots, T+L-1]. \tag{2}$$

However, even compressing the layout information to a single layout token for each text segment, an additional position ID must still be allocated. Moreover, the positional distance between adjacent text segments will be stretched due to the inserted layout tokens.

To address the issues of additional position IDs and the comprehension burden of stretched positional distance introduced by layout information, a straightforward and efficient positional encoding scheme is proposed that reuses the position IDs already utilized in the text tokens for layout tokens. Considering the cross-modality alignment within the same text segment, each single layout token is assigned with the position ID of the first text token in its corresponding text content (as illustrated in the lower left part of Fig. 2). Then the position IDs of a text segment P is expressed as:

$$P = [0, 1, \ldots, T-1, 0]. \tag{3}$$

Consequently, LayTokenLLM needs no additional position IDs for layout information, enabling the trained position IDs to be entirely dedicated to text content, and achieving a 100% T-Ratio. At the same time, the positional distance between adjacent text segments is preserved.

3.2. Pretraining Objective

Leveraging the autoregressive capabilities of LLMs and inspired by the "Next Token Prediction" in LLMs pretraining, the Next Interleaved Text and Layout Token Prediction (NTLP) is proposed. Previous works, such as LayTextLLM [28], focus solely on the prediction of text tokens under interleaved text and layout format without supervising layout information, even though they integrate layout information for LLMs. Considering the significant role of layout information for document understanding, as illustrated in Fig. 3, NTLP performs the next prediction task to reconstruct the document on all interleaved text and layout tokens, with training on both modalities. Thus, NTLP enables

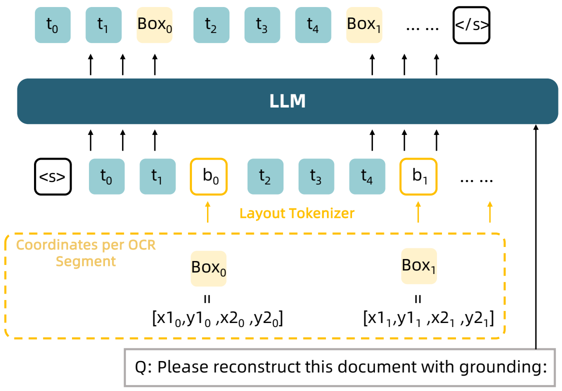

Figure 3. Illustration of the Next Interleaved Text and Layout Token Prediction objective. The supervision is conducted on both text and layout tokens to reconstruct text content and layout information simultaneously.

effective learning of layout information, enhances cross-modal prediction, and improves relational learning between text and layout modalities.

Specifically, NTLP minimizes the loss between the grounding truth of the next token and its prediction, whether the token is a text token or a layout token, and the loss function is defined as:

$$\mathcal{L} = \frac{1}{N-1} \sum_{i=1}^{N-1} \mathcal{L}_i(z^i | z^0, z^1, ..., z^{i-1}), \tag{4}$$

where z^i denotes the i-th token, while $\mathcal{L}_i$ represents the loss associated with predicting the token z^i based on all preceding tokens $z^0, z^1, \ldots, z^{i-1}$. For supervised training involving text, employing the commonly used cross-entropy (CE) loss associated with large language models (LLMs). Notably, given that the layout information has been encoded as a single token alongside the floating-point representation of layout (bounding box) information, NTLP introduces a dedicated layout head f_{lay} to map the layout hidden states to four-dimensional coordinates $[x_1, y_1, x_2, y_2]$, which serve as the predicted layout output for supervised training utilizing Mean Squared Error (MSE) loss. Thus, $\mathcal{L}_i$ can be expressed as:

$$\mathcal{L}_i = \begin{cases} \mathcal{L}_{CE}(f_{text}(z^i), y^i_{text}), z^i \in \mathcal{C}_{lay}, \\ \mathcal{L}_{MSE}(f_{lay}(z^i), Box^i), z^i \in \mathcal{C}_{text}, \end{cases} \tag{5}$$

where f_{text} denotes the text head, while y^i_{text} represents the one-hot encoding of the true label corresponding to text token $z^i \in \mathcal{C}_{text}$. Additionally, Box^i signifies the true four-dimensional coordinates for layout token $z^i \in \mathcal{C}_{lay}$.

4. Experiments

4.1. Training Dataset Collection

Pre-training data of LayTokenLLM utilizes the open-source document dataset called Layout-aware SFT data from LayoutLLM [31], which comprises an ensemble of diverse and high-quality data relevant to document understanding and information extraction tasks. For pre-training efficiently, filtering out too long documents with token lengths of more than 2k for effective pre-training.

SFT data of LayTokenLLM employs the datasets extensively used in single-page and multi-page document understanding tasks to ensure high-quality SFT. For the single-page document understanding task, the combined training sets of DocVQA [32] and SIBR [48] constitute the SFT dataset. DocVQA includes 50k question-answer pairs grounded on 12k document images. Meanwhile, SIBR is a real-world dataset for Visual Information Extraction tasks, covering challenging scenarios with difficult-to-recognize text like blur, partial occlusions, and printing shifts. Regarding multi-page document understanding SFT, leverages an ensemble of datasets that incorporates the training sets from MP-DocVQA [41] and DUDE [43].

4.2. Training Setup

In the experiments, two widely used models, Qwen1.5-7B [40] and LLama3-8B [11], are employed as the main LLM components of LayTokenLLM, referred to as LayTokenLLM-7B and LayTokenLLM-8B, respectively. Moreover, for a more comprehensive comparison with other Layout-as-Token methods, we also consider comparisons using the same training data and LLM backbone as ours, but employing different commonly used text and layout formats as input proposed by existing methods [4, 8, 13], such as "{text:"text",Box:[123, 456, 133, 500]}", as shown in Tab. 1. During both pre-training and SFT phases, as illustrated in Fig. 2, the LLM is frozen, while the parameters of the LoRA [16], layout tokenizer, and layout head are randomly initialized and updated to support lightweight training. The pretraining stage and single-page document SFT are trained for 3 epochs with a batch size of 64, a learning rate of 3e-4, and a maximum position ID set to 2048. To handle full training of long-context content under computational constraints, multi-page document SFT employs a 2-stage strategy: first, processing documents up to 4k tokens (maximum position ID) with a batch size of 32; second, handling those exceeding 4k up to 16k tokens and batch size is 8. The training is performed on 8 Nvidia A100 GPUs.

4.3. Evaluation Setup

For the single-page document understanding task, widely used benchmarks such as Document Visual Question Answering (Document VQA) and Visual Information Extraction (VIE) are employed, with only the test sets being utilized across all benchmarks. The Document VQA datasets specifically utilize the DocVQA test set, consisting of 5,188 questions. For the VIE task, which includes the SIBR [48], FUNSD [19], and CORD [33] benchmarks, the cleaned test

Setting	Single-page Document VQA				Multi-page Document VQA	
	SIBR	**FUNSD**	**CORD**	**DocVQA**	**MP-DocVQA**	**DUDE**
Plain Text						
Qwen1.5-7B-Chat [40]	38.81	52.52	29.71	64.27	47.15	28.98
Llama3-8B-Instruct [11]	51.77	57.47	40.00	74.22	50.75	24.89
Text + Layout-as-Modality						
DocLLM-7B$\diamond$ [44]	-	(51.80)	(67.40)	69.50	-	-
LayoutLLM-7B$\diamond$ [31]	-	79.98	63.10	74.27	-	-
Text + Layout-as-Token						
LayTextLLM-7B$\diamond$ [28]	-	72.00	45.50	77.20	-	-
text, [123, 456, 133, 500]$\star$	91.44	79.89	67.77	81.16	59.17	41.01
{text:"text",Box:[123, 456, 133, 500]}$\star$ [13]	91.45	79.98	68.57	81.98	55.96	37.96
<ref>text</ref><box>(123,456),(133,500)</box>$\star$ [3]	91.43	79.56	69.62	81.37	57.81	39.67
<ref>text</ref><box>[[253, 231, 733, 787]]</box>$\star$ [6]	88.24	78.17	56.32	80.18	56.16	40.82
LayTokenLLM-llama2-7B$\diamond$ (Ours)	90.13	76.10 (67.39)	67.60 (73.39)	79.98	56.30	36.59
LayTokenLLM-7B$\star$ (Ours)	92.03	78.72 (69.47)	73.79 (71.03)	81.50	72.81	49.72
LayTokenLLM-8B$\triangle$ (Ours)	**92.20**	**81.62 (70.96)**	**78.30 (75.35)**	**85.11**	**74.31**	**52.00**

Table 2. Comparison with the LLMs integrating layout information. Symbols $\diamond$, $\star$ and $\triangle$ represent the LLM backbones used: Llama2-7B, Qwen1.5-7B and Llama3-8B. Methods marked with $\star$ are trained identically to LayTokenLLM. $(\cdot)$ shows F1-scores on uncleaned FUNSD and CORD, as used in DocLLM [44]. '**Bold**' means the best in our series, while 'Underline' marks the best among all compared methods.

Models	SIBR	FUNSD	CORD	DocVQA
QwenVL-7B [3]	21.65	47.09	30.00	65.10
InternVL2-8B [7]	68.39	75.84	79.88	91.66
TextMonkey-7B [27]	51.30	65.49	67.54	66.70
LayTokenLLM-7B	92.03	78.72	73.79	81.50
LayTokenLLM-8B	**92.20**	**81.62**	**78.30**	**85.11**

Table 3. Comparison with MLLMs on single-page document datasets. '**Bold**' means the best in our series, while 'Underline' marks the best among all compared methods.

Models	MP-DocVQA	DUDE
LongVA-7B [51]	60.80	38.37
Idefics3-8B [21]	67.15	38.65
LLaVA-next-interleave-7B [23]	44.87	28.03
InternVL2-8B [6]	68.00	37.00
MPLUG-DocOwl2-8B [15]	69.42	46.77
LayTokenLLM-7B	72.81	49.72
LayTokenLLM-8B	**74.31**	**52.00**

Table 4. Comparison with MLLMs on multi-page document datasets.

sets of FUNSD and CORD provided by LayoutLLM [31] are used. SIBR's test set consists of 400 images, annotated with entity instances and links to challenge visual information extraction models. The FUNSD dataset features a test collection of 50 form images, each meticulously labeled with entities such as headers, questions, answers, and others, complemented by annotations for entity linking. Conversely, the CORD dataset encompasses a test suite of 100 receipt images, each enriched with annotations spanning 30 distinct entity categories, including but not limited to tax amounts and total prices. Following LayoutLLM [31], transform the VIE datasets into question-answering format, and the QA for both DocVQA and VIE task is evaluated by ANLS [32]. For the multi-image document understanding task, our experiments test on MP-DocVQA and DUDE, which are widely used for multi-page document understanding. Following the evaluation metric settings of the original datasets, the MP-DocVQA is evaluated by ANLS, while DUDE adopts a version of ANLS that it has modified. The hyperparameters during inference (e.g., top k, beam search, etc.) are set to their default values.

4.4. Main Results

4.4.1 Effectiveness Comparison

Comparison with LLMs combined with Layout Information is illustrated in Tab. 2. It can be seen that variant LLMs that incorporate layout information consistently outperform plain text models in all document comprehension tasks, proving that layout is crucial for document understanding. Moreover, our method achieves competitive results in single-page document VQA (leading in 2 subtasks and with a higher average compared to other methods using the same LLM). Notably, LayTokenLLM outperforms other methods by a large margin in multi-page document VQA (more than 10% improvement among the marked $\star$ approaches). We believe the more significant improvement on multi-page document VQA is due to the fact that in single-page documents most cases do not exceed the trained maximum position ID. Consequently, the

Text and Layout Format	**FLOPs/MACs↓**	**SP Doc VQA ANLS Avg↑**	**MP Doc VQA ANLS Avg↑**
Plain Text (w/o Layout)	7.95/3.98	46.33	38.07
{text:"text",Box:[123, 456, 133, 500]} [13]	28.76/14.57	80.49	46.96
<ref>text</ref><box>(123,456),(133,500)</box> [3, 27]	28.69/14.34	80.50	48.74
<ref>text</ref><box>[[253, 231, 733, 787]]</box> [6]	32.81/16.40	75.73	48.49
text+layout_token (**LayTokenLLM**)	**9.32/5.36**	**81.51**	**61.27**

Table 5. Comparison with Layout-as-Token Methods on the multi-page document (MP Doc) understanding tasks, which all initialized from Qwen1.5-7B-Chat. 'FLOPs/MACs' denotes the Floating Point Operations Per Second (FLOPs) and the Multiply-Accumulate Operations (MACs) on DocVQA which are broadly used to measure the computational complexity and efficiency.

impact of additional layout information on the LLM can be largely alleviated through further fine-tuning. In contrast, in multi-page documents with extensive context that require the allocation of numerous position IDs, the introduction of additional position IDs for layout information may exacerbate the challenges associated with long-context processing. Our LayTokenLLM demonstrates remarkable performance by effectively circumventing the need for extra position IDs dedicated to layout information, thereby emphasizing its efficiency and superiority in handling such complex scenarios. Furthermore, experiments with different LLM backbone initializations consistently achieve superior results across all benchmarks, substantiating that LayTokenLLM can adapt to various LLMs.

Comparison with MLLMs is shown in Tab. 3 and Tab. 4. Considering the distinct advantages of existing MLLMs in both single-page and multi-page document understanding tasks, representative works in each task are selected for comparison. It can be seen that LayTokenLLM achieves the comparable performance of the best model InternVL2-8B across most single-page tasks. Particularly in challenging scenarios like SIBR, which covers difficult-to-recognize text, LayTokenLLM achieves 92.20%, compared to InternVL2-8B's 68.39%, showcasing a significant advantage which is attributed to the enhanced preservation of textual and layout information of the document. Furthermore, in multi-page document understanding, LayTokenLLM exceeds both InternVL2 and MPLUG-DocOwl by over 5% on the DUDE dataset. This superiority may stem from MLLMs often compressing images into fewer tokens for multi-page documents, which results in the loss of textual information. In contrast, LayTokenLLM retains a greater proportion of text, enhancing document representation and discernment.

4.4.2 Efficiency Comparison

Tab. 5 presents a comparative analysis of the Layout-as-Token method in terms of efficiency and performance. Compared to the methods with a comparable number of parameters, LayTokenLLM demonstrates superior performance in both single-page and multi-page document understanding tasks while exhibiting better efficiency. Notably, due to its lightweight design, our LayTokenLLM exhibits a low processing time that is comparable to only Plain Text input, and more than half that of alternative methods integrating layout information. These results affirm that LayTokenLLM is both effective and efficient.

#	Layout Token		NTLP	SP Doc ANLS Avg	MP Doc ANLS Avg
	Layout Tokenizer	LayPosID			
0				80.07	50.09
1	✓			78.89	58.22
2	✓	✓		79.27	60.50
3	✓	✓	✓	**81.51**	**61.27**

Table 6. Ablation study on single-page document (SP Doc) and multi-page document (MP Doc) understanding tasks. LayPosID represents the positional encoding scheme for our Layout Token.

4.5. Ablation Study

To evaluate the effectiveness of the proposed Layout Token and pre-training objective in the document understanding task, an ablation study is conducted (see Tab. 6).

Initial Baseline. The #0 baseline disables both *Layout Token* and *NTLP* objective. It utilizes uncompressed layout information as textual tokens for LLM input, consistent with the fine-tuning data as LayTokenLLM settings. Under this configuration, the baseline achieves a high average performance of 80.07% in single-page document understanding tasks, but performs poorly in multi-page document scenarios due to the layout information occupying critical text learning and understanding space.

Effect of Layout Token. The proposed Layout Token in LayTokenLLM is generated by our Layout Tokenizer with LayPosID. In #1, the layout tokenizer is introduced. Compared to #0, the proposed compression of layout information enables more text information to be learned within a fixed window, leading to significant performance improvements in multi-page tasks. Meanwhile, in single-page document scenarios, there is a slight degradation in performance due to the information loss caused by layout information compression. In #2, the framework extends #1 via LayPosID (our positional encoding scheme), which further eliminates the need for extra layout positional indexing and achieves 100% T-ratio. As a result, #2 demonstrates additional performance gains over #1, with a substantial im-

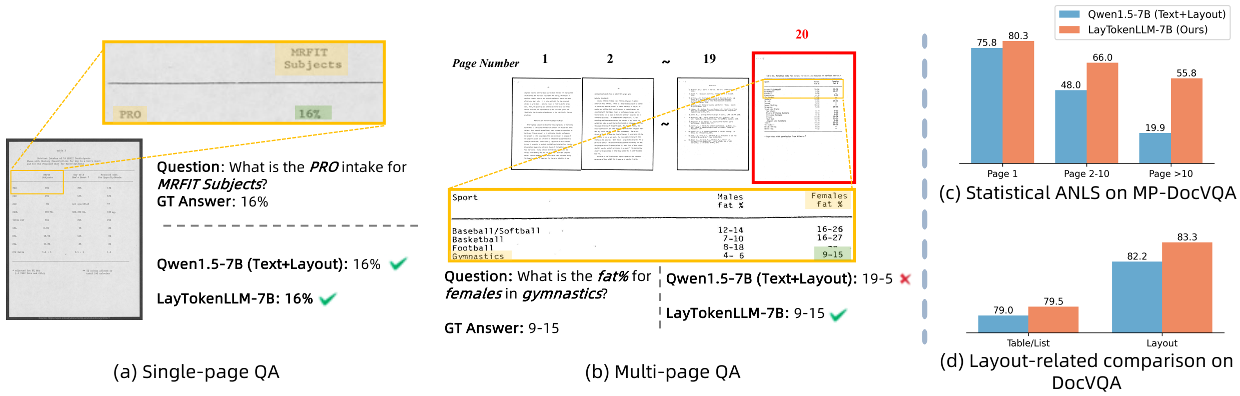

Figure 4. Qualitative results on (a) single-page and (b) multi-page document QA, where "Qwen1.5-7B (Text+Layout)" is trained with the same data and LLM as LayTokenLLM-7B, but employs norm text and layout format ("text, [123, 456, 133, 500]") instead of Layout Token. The **Yellow** highlights denote the relevant areas or keys for QA, while the **Green** highlights indicate the correct answers. (c) Distribution of statistical ANLS in terms of pages along the posed questions on MP-DocVQA. (d) Comparison of layout-related performance using the single-page document dataset, DocVQA.

provement of 2.3% in multi-page document understanding.

Effect of NTLP Objective. Compared with #2, #3 further incorporated the *NTLP*, which employs a next interleaved text and layout prediction task. The objective enhances both text and layout representation learning, as well as their interconnections. Performance improvements are observed in both single-page and multi-page document understanding tasks, with increases of 2.2% and 0.8% respectively.

Overall, the ablation study confirms the effectiveness of the Layout Token and the NTLP pre-training objective.

4.6. Qualitative Results

To further study the effectiveness of our method, two examples from single-page and multi-page document QA scenarios and statistical analysis related to page numbers are presented in Fig. 4. In the context of key-value QAs that rely on spatial layouts, the Qwen1.5-7B model, which integrates standard text and layout formats, can accurately respond on single-page documents (Fig. 4(a)) but exhibits answer confusion on multi-page documents (Fig. 4(b)). In contrast, LayTokenLLM achieves correct reasoning on both single-page and multi-page documents. We think the confusion in multi-page documents is mainly due to the added position IDs overhead caused by incorporating layout information, leading to long-context issues. So we further conduct a statistical analysis on the performance related to the page ordinal number with proposed questions, as depicted in Fig. 4(c). It can be seen that the performance of the Qwen1.5-7B model with the direct integration of layout information declines significantly with an increasing number of pages. In contrast, our LayTokenLLM exhibits a marked performance advantage as pages increase, highlighting its superiority, especially in understanding long-context documents. Moreover, LayTokenLLM's layout representation performance is further evaluated under conditions excluding the impact of position ID overhead (short-context scenario), using the "table/list" and "layout" subset of the DocVQA dataset, see Fig. 4(d). The results show that LayTokenLLM not only avoids negative impacts but also improves results compared with Qwen1.5-7B (Text+Layout), demonstrating its effectiveness in re-expressing layout information. Overall, LayTokenLLM ensures comprehensive text learning while clearly preserving layout information, leading a more complete document understanding.

5. Limitations

Although the proposed Layout Token demonstrates that LayTokenLLM can effectively address text-dense documents with rich layout information, it may overlook certain graphical elements, such as charts and icons. Additionally, although NTLP pre-training has been shown to enhance document understanding, future work could explore more granular tasks, such as fine-grained layout relationship prediction. Further research may focus on equipping LayTokenLLM with these capabilities.

6. Conclusion

We propose LayTokenLLM, which incorporates a simple yet effective Layout Token to ensure comprehensive learning of text content while alleviating long-context issues introduced by layout information. Furthermore, an interleaved text and layout token next prediction pre-training objective is utilized to enhance cross-modal prediction and relational learning between text and layout modalities. Extensive experiments demonstrate the effectiveness of LayTokenLLM across diverse benchmarks for both single-page and multi-page document understanding.

References

[1] Gpt-4v(ision) system card. 2023. 3

[2] Srikar Appalaraju, Bhavan Jasani, and Bhargava Urala Kota. DocFormer: End-to-end transformer for document understanding. In *ICCV*, pages 4171–4186, 2021. 3

[3] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 2, 6, 7

[4] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. 2023. 3, 5

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, pages 1877–1901. Curran Associates, Inc., 2020. 3

[6] 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. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023. 2, 6, 7

[7] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024. 3, 6

[8] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 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, 2024. 3, 5

[9] Lei Cui, Yiheng Xu, Tengchao Lv, and Furu Wei. Document ai: Benchmarks, models and applications. *arXiv preprint arXiv:2111.08609*, 2021. 1

[10] Cheng Da, Chuwei Luo, Qi Zheng, and Cong Yao. Vision grid transformer for document layout analysis. In *ICCV*, pages 19462–19472, 2023. 3

[11] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 5, 6

[12] Zhangxuan Gu, Changhua Meng, Ke Wang, Jun Lan, Weiqiang Wang, Ming Gu, and Liqing Zhang. Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In *CVPR*, pages 4583–4592, 2022. 3

[13] Jiabang He, Lei Wang, Yi Hu, Ning Liu, Hui Liu, Xing Xu, and Heng Tao Shen. Icl-d3ie: In-context learning with diverse demonstrations updating for document information extraction. *ICCV*, 2023. 2, 3, 5, 6, 7

[14] Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Chen Li, Ji Zhang, Qin Jin, Fei Huang, et al. mplug-docowl 1.5: Unified structure learning for ocr-free document understanding. *arXiv preprint arXiv:2403.12895*, 2024. 3

[15] Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. mplug-docowl2: High-resolution compressing for ocr-free multi-page document understanding. *arXiv preprint arXiv:2409.03420*, 2024. 3, 6

[16] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 5

[17] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *ACM Multimedia*, 2022. 3

[18] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4083–4091, New York, NY, USA, 2022. Association for Computing Machinery. 3

[19] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Funsd: A dataset for form understanding in noisy scanned documents, 2019. 1, 5

[20] Marcel Lamott, Yves-Noel Weweler, Adrian Ulges, Faisal Shafait, Dirk Krechel, and Darko Obradovic. Lapdoc: Layout-aware prompting for documents. 2024. 1, 2, 3

[21] Hugo Laurençon, Andrés Marafioti, Victor Sanh, and Léo Tronchon. Building and better understanding vision-language models: insights and future directions. *arXiv preprint arXiv:2408.12637*, 2024. 6

[22] Chenliang Li, Bin Bi, and Ming Yan. StructuralLM: Structural pre-training for form understanding. In *ACL*, 2021. 3

[23] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024. 3, 6

[24] Wentong Li, Yuqian Yuan, Jian Liu, Dongqi Tang, Song Wang, Jianke Zhu, and Lei Zhang. Tokenpacker: Efficient visual projector for multimodal llm. *arXiv preprint arXiv:2407.02392*, 2024. 3

[25] Yulin Li, Yuxi Qian, Yuechen Yu, Xiameng Qin, Chengquan Zhang, Yan Liu, Kun Yao, Junyu Han, Jingtuo Liu, and Errui Ding. Structext: Structured text understanding with multi-modal transformers. In *ACM Multimedia*, pages 1912–1920, 2021. 3

[26] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information

Processing Systems, pages 34892–34916. Curran Associates, Inc., 2023. 3

[27] Yuliang Liu, Biao Yang, Qiang Liu, Zhang Li, Zhiyin Ma, Shuo Zhang, and Xiang Bai. Textmonkey: An ocr-free large multimodal model for understanding document. *arXiv preprint arXiv:2403.04473*, 2024. 2, 6, 7

[28] Jinghui Lu, Haiyang Yu, Yanjie Wang, Yongjie Ye, Jingqun Tang, Ziwei Yang, Binghong Wu, Qi Liu, Hao Feng, Han Wang, Hao Liu, and Can Huang. A bounding box is worth one token: Interleaving layout and text in a large language model for document understanding, 2024. 1, 2, 3, 4, 6

[29] Chuwei Luo, Guozhi Tang, Qi Zheng, Cong Yao, Lianwen Jin, Chenliang Li, Yang Xue, and Luo Si. Bi-vldoc: Bidirectional vision-language modeling for visually-rich document understanding. *arXiv preprint arXiv:2206.13155*, 2022. 3

[30] Chuwei Luo, Changxu Cheng, Qi Zheng, and Cong Yao. Geolayoutlm: Geometric pre-training for visual information extraction. In *CVPR*, pages 7092–7101, 2023. 3

[31] Chuwei Luo, Yufan Shen, Zhaoqing Zhu, Qi Zheng, Zhi Yu, and Cong Yao. Layoutllm: Layout instruction tuning with large language models for document understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15630–15640, 2024. 2, 3, 5, 6

[32] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *WACV*, pages 2200–2209, 2021. 1, 5, 6

[33] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. {CORD}: A consolidated receipt dataset for post-{ocr} parsing. In *Workshop on Document Intelligence at NeurIPS 2019*, 2019. 1, 5

[34] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*, 2023. 2

[35] Vincent Perot, Kai Kang, Florian Luisier, Guolong Su, Xiaoyu Sun, Ramya Sree Boppana, Zilong Wang, Zifeng Wang, Jiaqi Mu, Hao Zhang, Chen-Yu Lee, and Nan Hua. Lmdx: Language model-based document information extraction and localization, 2024. 2, 3

[36] Yufan Shen, Chuwei Luo, Zhaoqing Zhu, Yang Chen, Qi Zheng, Zhi Yu, Jiajun Bu, and Cong Yao. Proctag: Process tagging for assessing the efficacy of document instruction data. *arXiv preprint arXiv:2407.12358*, 2024. 3

[37] Woomin Song, Seunghyuk Oh, Sangwoo Mo, Jaehyung Kim, Sukmin Yun, Jung-Woo Ha, and Jinwoo Shin. Hierarchical context merging: Better long context understanding for pre-trained llms. *arXiv preprint arXiv:2404.10308*, 2024. 2

[38] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 4

[39] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 3

[40] Qwen Team. Introducing qwen1.5, 2024. 3, 5, 6

[41] Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. Hierarchical multimodal transformers for multipage docvqa. *Pattern Recognition*, 144:109834, 2023. 5

[42] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

[43] Jordy Van Landeghem, Rubèn Tito, Łukasz Borchmann, Michał Pietruszka, Pawel Joziak, Rafal Powalski, Dawid Jurkiewicz, Mickaël Coustaty, Bertrand Anckaert, Ernest Valveny, et al. Document understanding dataset and evaluation (dude). In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19528–19540, 2023. 5

[44] Dongsheng Wang, Natraj Raman, Mathieu Sibue, Zhiqiang Ma, Petr Babkin, Simerjot Kaur, Yulong Pei, Armineh Nourbakhsh, and Xiaomo Liu. DocLLM: A layout-aware generative language model for multimodal document understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8529–8548, Bangkok, Thailand, 2024. Association for Computational Linguistics. 1, 2, 3, 6

[45] Yiheng Xu, Minghao Li, Lei Cui, and Shaohan Huang. LayoutLM: Pre-training of text and layout for document image understanding. In *KDD*, pages 1192–1200, 2020. 3

[46] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL) 2021*, 2021. 3

[47] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In *ACL*, 2021. 3

[48] Zhibo Yang, Rujiao Long, Pengfei Wang, Sibo Song, Humen Zhong, Wenqing Cheng, Xiang Bai, and Cong Yao. Modeling entities as semantic points for visual information extraction in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1, 5

[49] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13040–13051, 2024. 3

[50] Yuechen Yu, Yulin Li, Chengquan Zhang, Xiaoqiang Zhang, Zengyuan Guo, Xiameng Qin, Kun Yao, Junyu Han, Errui Ding, and Jingdong Wang. Structextv2: Masked visual-textual prediction for document image pre-training. In *ICLR*, 2023. 3

[51] Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from

language to vision. *arXiv preprint arXiv:2406.16852*, 2024. 3, 6

[52] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations*, 2024. 3