ERNIE-Layout: Layout Knowledge Enhanced Pre-training for Visually-rich Document Understanding

Qiming Peng[1*], **Yinxu Pan**[1*], **Wenjin Wang**[2*†], **Bin Luo**[1‡], **Zhenyu Zhang**[1], **Zhengjie Huang**[1], **Teng Hu**[1], **Weichong Yin**[1], **Yongfeng Chen**[1], **Yin Zhang**[2], **Shikun Feng**[1], **Yu Sun**[1], **Hao Tian**[1], **Hua Wu**[1], **Haifeng Wang**[1]

[1]Baidu Inc., Beijing, China

[2]Zhejiang University, Hangzhou, China

{pengqiming, panyinxu, luobin06, zhangzhenyu07}@baidu.com

{wangwenjin, zhangyin98}@zju.edu.cn {yinweichong, fengshikun, sunyu02}@baidu.com

Abstract

Recent years have witnessed the rise and success of pre-training techniques in visually-rich document understanding. However, most existing methods lack the systematic mining and utilization of layout-centered knowledge, leading to sub-optimal performances. In this paper, we propose ERNIE-Layout, a novel document pre-training solution with layout knowledge enhancement in the whole workflow, to learn better representations that combine the features from text, layout, and image. Specifically, we first rearrange input sequences in the serialization stage, and then present a correlative pre-training task, reading order prediction, to learn the proper reading order of documents. To improve the layout awareness of the model, we integrate a spatial-aware disentangled attention into the multi-modal transformer and a replaced regions prediction task into the pre-training phase. Experimental results show that ERNIE-Layout achieves superior performance on various downstream tasks, setting new state-of-the-art on key information extraction, document image classification, and document question answering datasets. The code and models are publicly available at PaddleNLP[1].

1 Introduction

Visually-rich Document Understanding (VrDU) is an important research field aiming to handle various types of scanned or digital-born business documents (e.g., forms, invoices), which has attracted great attention from the industry and academia due to its various applications. Distinct from conventional natural language understanding (NLU) tasks that use only plain text, VrDU models have the opportunity to access the most primitive data features. Herein, the diversity and complexity of document formats pose new challenges to the task, an ideal model needs to make full use of the textual, layout, and even visual information to fully understand visually-rich document like humans.

The preliminary works for VrDU (Yang et al., 2016, 2017; Katti et al., 2018; Sarkhel and Nandi, 2019; Cheng et al., 2020) usually adopt uni-modal or shallow multi-modal fusion approaches, which are task-specific and require massive data annotations. Recently, pre-training language models have swept the field, LayoutLM (Xu et al., 2020), LayoutLMv2 (Xu et al., 2021), and some advanced document pre-training approaches (Li et al., 2021a; Appalaraju et al., 2021; Gu et al., 2022) have been proposed successively and achieved great successes in various VrDU tasks. Unlike popular uni-modal or vision-language frameworks (Devlin et al., 2019; Liu et al., 2019; Lu et al., 2019; Yu et al., 2021), the uniqueness of document understanding models lies in how to exploit the layout knowledge.

However, existing document pre-training solutions typically fall into the trap of simply taking 2D coordinates as an extension of 1D positions to endow the model layout awareness. Considering the characteristics of VrDU, we believe that the layout-centered knowledge should be systematically mined and utilized from two aspects: (1) On the one hand, *layout implicitly reflects the proper reading order of documents*, while previous methods are used to perform the serialization by multiplexing the results of Optical Character Recognition (OCR), which roughly arrange tokens in the top-to-bottom and left-to-right manner (Wang et al., 2021c; Gu et al., 2022). Inevitably, it is inconsistent with human reading habits for documents with complex layouts (e.g., tables, forms, multi-column templates) and leads to sub-optimal performances for downstream tasks. (2) On the other hand, *layout is actually the third modality besides language and vision*, while current models are used to take lay-

*Equal contribution.

†Work done during internship at Baidu Inc.

‡Corresponding author: Bin Luo.

[1]https://github.com/PaddlePaddle/PaddleNLP/tree/develop/model_zoo/ernie-layout

out as a special position feature, such as the layout embedding in input layer (Xu et al., 2020) or the bias item in attention layer (Xu et al., 2021). The lack of cross-modal interaction between layout and text/image might restrict the model from learning the role of layout in semantic expression.

To achieve these goals, we propose a systematic layout knowledge enhanced pre-training approach, ERNIE-Layout[2], to improve the performances of document understanding tasks. First of all, we employ an off-the-shelf layout-based document parser in the serialization stage to generate an appropriate reading order for each input document, so that the input sequences received by the model are more in line with human reading habits than using the rough raster-scanning order. Then, each textual/visual token is equipped with its position embedding and layout embedding, and sent to the stacked multi-modal transformer layers. To enhance cross-modal interaction, we present a spatial-aware disentangled attention mechanism, inspired by the disentangled attention of DeBERTa (He et al., 2021), in which the attention weights between tokens are computed using disentangled matrices based on their hidden states and relative positions. In the end, layout not only acts as the 2D position attribute of input tokens, but also contributes a spatial perspective to the calculation of semantic similarity.

With satisfactory serialization results, we propose the pre-training task, reading order prediction, to predict the next token for each position, which facilitates the consistency within the same arranged text segment and the discrimination between different segments. Furthermore, when pre-training, we also adopt the classic masked visual-language modeling and text-image alignment tasks (Xu et al., 2021), and present a fine-grained multi-modal task, replaced regions prediction, to learn the correlation among language, vision and layout.

We construct broad experiments on three representative VrDU downstream tasks with six publicly available datasets to evaluate the performance of the pre-trained model, i.e., the key information extraction task with the FUNSD (Jain and Wigington, 2019), CORD (Park et al., 2019), SROIE (Huang et al., 2019), Kleister-NDA (Graliński et al., 2021) datasets, the document question answering task with the DocVQA (Mathew et al., 2021) dataset, and the document image classification task with the RVL-CDIP (Harley et al., 2015) dataset. The results show that ERNIE-Layout significantly outperforms strong baselines on almost all tasks, proving the effectiveness of our two-part layout knowledge enhancement philosophy.

The contributions are summarized as follows:

- ERNIE-Layout proposes to rearrange the order of input tokens in serialization and adopt a reading order prediction task in pre-training. To the best of our knowledge, ERNIE-Layout is the first attempt to consider the proper reading order in document pre-training.
- ERNIE-Layout incorporates the spatial-aware disentangled attention mechanism in the multi-modal transformer, and designs a replaced regions prediction pre-training task, to facilitate the fine-grained interaction across textual, visual, and layout modalities.
- ERNIE-Layout refreshes the state-of-the-art of various VrDU tasks, and extensive experiments demonstrate the effectiveness of exploiting layout-centered knowledge.

2 Related Work

Layout-aware Pre-trained Model. Humans understand visually rich documents through many perspectives, such as language, vision, and layout. Based on the powerful modeling ability of Transformer (Vaswani et al., 2017), LayoutLM (Xu et al., 2020) initially embeds the 2D coordinates as layout embeddings for each token and extends the famous masked language modeling pre-training task (Devlin et al., 2019) to masked visual-language modeling, which opens the prologue of layout-aware pre-trained models. Afterwards, LayoutLMv2 (Xu et al., 2021) concatenates document image patches with textual tokens, and two pre-training tasks, text-image matching and text-image alignment, are proposed to realize the cross-modal interaction. StructuralLM (Li et al., 2021a) leverages segment-level, instead of word-level, layout features to make the model aware of which words come from the same cell. DocFormer (Appalaraju et al., 2021) shares the learned spatial embeddings across modalities, making it easy for the model to correlate text to visual tokens and vice versa. TILT (Powalski et al., 2021) proposes an encoder-decoder model to generate results that are not explicitly included in the input sequence to solve the limitations of sequence

[2]It is named after the knowledge enhanced pre-training model, ERNIE (Sun et al., 2019), as a layout enhanced version.

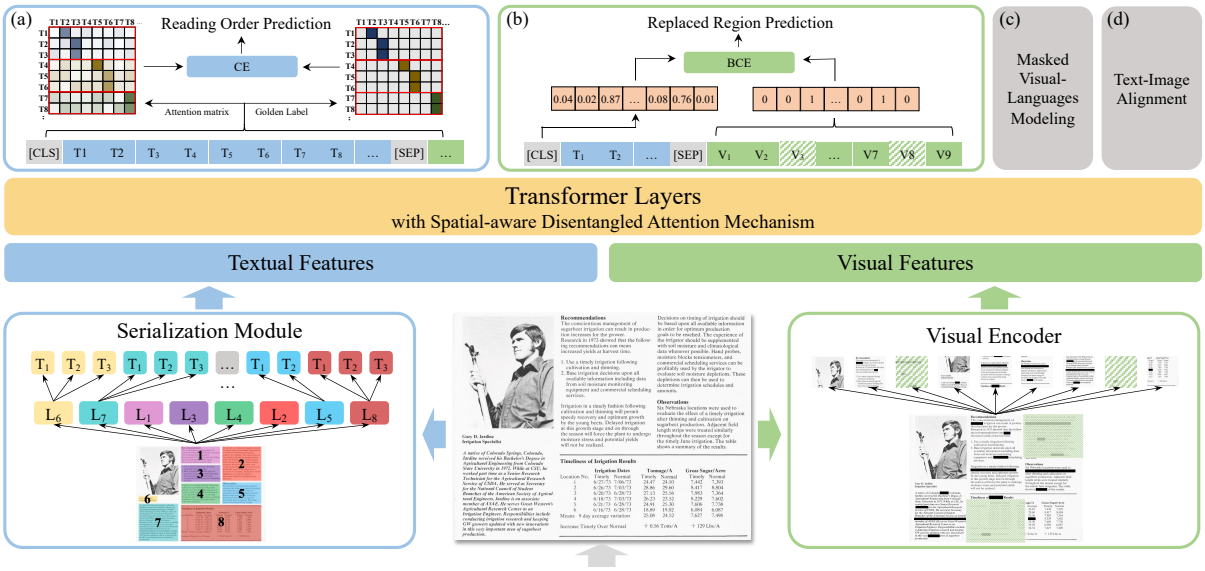


Figure 1: The architecture and pre-training objectives of ERNIE-Layout. The serialization module is introduced to correct the order of raster-scan, and the visual encoder extracts corresponding image features. With the spatial-aware disentangled attention mechanism, ERNIE-Layout is pre-trained with four tasks.

labeling. However, these methods ignore the potential value of layout in-depth and directly rely on a raster-scanning serialization, which is contrary to human reading habits. To solve this problem, LayoutReader (Wang et al., 2021c) designs a sequence-to-sequence framework to generate an appropriate reading order for each document. Unfortunately, it is carefully designed for reading order detection and cannot directly empower various document understanding tasks. Besides, the above methods are used to regard layout as a subsidiary feature of text along with the idea of LayoutLM, but the same text with different layouts may also express different semantics. Therefore, we believe that layout should be regarded as the third modality independent of language and vision.

Knowledge-enhanced Representation. Following the BERT (Devlin et al., 2019) architecture, many efforts are devoted to pre-trained language models for learning informative representations. There are some studies show that extra knowledge, such as facts in WikiData and WordNet, can further benefit the pre-trained models (Zhang et al., 2019; Liu et al., 2020; He et al., 2020; Wang et al., 2021b), but the embeddings of words in the text and entities in the knowledge graphs are not in the same vector space, so a cumbersome adaptation module is required (He et al., 2020; Wang et al., 2021a). Another research line is to excavate the potential human cognitive laws of the text itself: ERNIE (Sun et al., 2019) creativity proposes entity-level mask in pre-training to incorporate the human knowledge into language models. Similarly, SpanBERT (Joshi et al., 2020) modifies the making schema and training objectives to better represent and predict text spans. BERT-wwm (Cui et al., 2021) introduces a whole word masking strategy for Chinese language models. Outside the field of plain text, ERNIE-ViL (Yu et al., 2021) incorporates structured knowledge obtained from scene graphs to learn joint representations of vision-language. Inspired by the above work, we leverage the implicit knowledge related to layout, e.g., reading order, for the understanding of visually rich documents.

3 Methodology

Figure 1 shows an overview of the ERNIE-Layout. Given a document, ERNIE-Layout rearranges the token sequence with the layout knowledge and extracts visual features from the visual encoder. The textual and layout embeddings are combined into textual features through a linear projection, and similar operations are executed for visual embeddings. Next, these features are concatenated and fed into the stacked multi-modal transformer layers, which are equipped with the proposed spatial-aware disentangled attention mechanism. For pre-training, ERNIE-Layout adopts four pre-training tasks, including the new proposed reading order prediction, replaced region prediction tasks, and the traditional masked visual-language modeling, text-image alignment tasks.

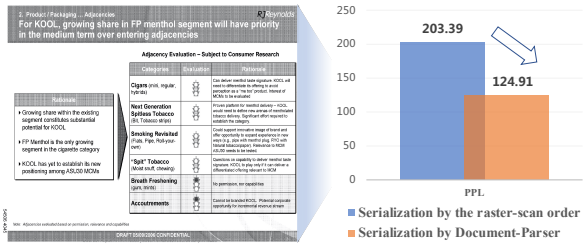


Figure 2: The effect of layout knowledge enhanced serialization compared with vanilla raster-scanning order. By using Document-Parser, the perplexity of the document with a complex layout is significantly reduced.

3.1 Serialization Module

Before feeding a visually rich document to neural networks, serialization, that is, recognizing the text and arranging them in proper order, is a necessary step. First, an OCR tool is used to obtain the words and their coordinates in documents. Then the traditional method arranges the identified elements from left to right and top to bottom by raster-scan to generate the input sequence. Although this method is easy to implement, it cannot correctly handle documents with complex layouts. Look at the example in Figure 2, there are two tables, and the cells in these tables also have some newline text. Suppose we want to extract some information from it. In that case, the expected results may not be obtained according to the raster-scanning order, because the words in the same cell are scattered.

Inspired by the human reading habits, we adopt Document-Parser, an advanced document layout analysis toolkit based on Layout-Parser[3], to serialize these documents. As shown in Figure 1, based on the words and their boxes recognized by OCR, it first detects document elements (e.g., paragraphs, lists, tables, figures), and then uses specific algorithms to obtain the logical relationship between words based on the characteristics of different elements, to obtain the proper reading order.

To quantitatively analyze the benefits of layout knowledge enhanced serialization, we take perplexity (PPL), calculated by GPT-2 (Radford et al., 2019), as the evaluation metric. PPL is widely used for measuring the performance of language models. From Figure 2, we find that the input sequence serialized by Document-Parser has a lower PPL than the raster-scanning order. More implementation details and cases are detailed in Appendix A.1.

[3] https://github.com/Layout-Parser/layout-parser

3.2 Input Representation

The input sequence of ERNIE-Layout includes a textual part and a visual part, and the representation of each part is a combination of its modal features and layout embeddings (Xu et al., 2021).

Text Embedding. The document tokens after the serialization module are used as the text sequence. Following the pre-processing of BERT-Style models (Devlin et al., 2019), two special tokens `[CLS]` and `[SEP]` are appended at the beginning and end of the text sequence, respectively. Finally, the text embedding of token sequence T is expressed as:

$$\mathbf{T} = \mathbf{E}_{tk}(T) + \mathbf{E}_{1p}(T) + \mathbf{E}_{tp}(T), \quad (1)$$

where $\mathbf{E}_{tk}$, $\mathbf{E}_{1p}$, $\mathbf{E}_{tp}$ respectively denote the token embedding, 1D position embedding, and token type embedding layer.

Visual Embedding. To extract the visual features of documents, we employ Faster-RCNN (Ren et al., 2015) as the backbone of visual encoder. In particular, the document image is resized to 224×224 and fed into visual backbone, an adaptive pooling layer is introduced to convert the output into a feature map with a fixed width W and height H (here, we set them to 7). Next, we flatten the feature map into a visual sequence V, and project each visual token to the same dimension as text embedding with a linear layer $F_{vs}(\cdot)$. Similarity, the 1D position and token type `[V]` are taken into consideration for the generation of visual embedding:

$$\mathbf{V} = F_{vs}(V) + \mathbf{E}_{1p}(V) + \mathbf{E}_{tp}(\texttt{[V]}). \quad (2)$$

Layout Embedding. For each textual token, the OCR tool provides its 2D coordinates with the width and height of the bounding box $(x_0, y_0, x_1, y_1, w, h)$, where (x_0, y_0) denote coordinates of the upper left corner of the bounding box, (x_1, y_1) denote the bottom right corner, $w = x_1 - x_0$, $h = y_1 - y_0$, and all the coordinate values are normalized in the range $[0, 1000]$. For the visual token, similar calculation processes can also be performed. To look up the layout embeddings of textual/visual token, we construct separate embedding layers in the horizontal and vertical directions:

$$\mathbf{L} = \mathbf{E}_{2x}(x_0, x_1, w) + \mathbf{E}_{2y}(y_0, y_1, h), \quad (3)$$

where $\mathbf{E}_{2x}$ is the x-axis embedding layer, $\mathbf{E}_{2y}$ denotes the y-axis embedding layer.

To achieve the ultimate input representation $\mathbf{H}$ of ERNIE-Layout, we integrate the embedding of each textual and visual token with its corresponding layout embeddings. Finally, the textual and visual embeddings are combined together to obtain a long sequence with the length being $N + HW$, where N is the max length of the textual part:

$$\mathbf{H} = [\mathbf{T} + \mathbf{L}; \mathbf{V} + \mathbf{L}]. \tag{4}$$

3.3 Multi-modal Transformer

In the final input representation, textual and visual tokens are spliced together, and the self-attention mechanism in the transformer supports their layer-aware cross-modal interaction. However, as a unique modality, layout features should be involved in the calculation of attention weight, and the tightness between them and contents (collectively refers to text and image) should also be taken into account explicitly. Inspired by the disentangled attention of DeBERTa (He et al., 2021), in which the attention weights among tokens are computed using disentangled matrices on their contents and relative positions, we propose spatial-aware disentangled attention for the multi-modal transformer to enable the participation of layout features.

Firstly, we take 1D position as an example to define the relative distance $\delta_{1p}(\cdot)$ between token i and j, and the definition in the x-axis and y-axis directions of 2D layout is the same:

$$\delta_{1p}(i,j) = \begin{cases} 0 & \text{for } i-j \leqslant -k \\ 2k-1 & \text{for } i-j \geqslant k \\ i-j+k & \text{others,} \end{cases} \tag{5}$$

Next, to construct relative position vectors consistent with the input dimension, we introduce three relative position embedding tables $\mathbf{E}'_{1p}$, $\mathbf{E}'_{2x}$, $\mathbf{E}'_{2y}$ for 1D position, 2D x-axis and 2D y-axis. After looking up the embedding table, a series of projection matrices map these relative position vectors as well as the content vectors into $\mathbf{Q}^\star$, $\mathbf{K}^\star$, $\mathbf{V}^\star$ in the attention mechanism, where $\star \in \{ct, 1p, 2x, 2y\}$. In the process of attention calculation, we decouple the raw score into four parts to realize the in-depth exchange of 1D/2D features and contents:

$$A_{ij}^{ct,ct} = \mathbf{Q}_i^{ct} {\mathbf{K}_j^{ct}}^\top, \tag{6}$$

$$A_{ij}^{ct,1p} = \mathbf{Q}_i^{ct} {\mathbf{K}_{\delta_{1p}(i,j)}^{1p}}^\top + \mathbf{K}_j^{ct} {\mathbf{Q}_{\delta_{1p}(j,i)}^{1p}}^\top, \tag{7}$$

$$A_{ij}^{ct,2x} = \mathbf{Q}_i^{ct} {\mathbf{K}_{\delta_{2x}(i,j)}^{2x}}^\top + \mathbf{K}_j^{ct} {\mathbf{Q}_{\delta_{2x}(j,i)}^{2x}}^\top, \tag{8}$$

$$A_{ij}^{ct,2y} = \mathbf{Q}_i^{ct} {\mathbf{K}_{\delta_{2y}(i,j)}^{2y}}^\top + \mathbf{K}_j^{ct} {\mathbf{Q}_{\delta_{2y}(j,i)}^{2y}}^\top. \tag{9}$$

Finally, all these attention scores are summed up to get the attention matrix $\hat{\mathbf{A}}$. With the scaling and normalization operations, the output of spatial-aware disentangled attention is[4]:

$$\hat{A}_{ij} = A_{ij}^{ct,ct} + A_{ij}^{ct,1p} + A_{ij}^{ct,2x} + A_{ij}^{ct,2y}, \tag{10}$$

$$\mathbf{H}_{out} = \text{softmax}(\frac{\hat{\mathbf{A}}}{\sqrt{3d}})\mathbf{V}^{ct}. \tag{11}$$

3.4 Pre-training Tasks

There are four pre-training tasks in ERNIE-Layout. We design reading order prediction and replaced region prediction, as well as borrow masked visual-language modeling and text-image alignment from LayoutLMv2 (Xu et al., 2021), so that the model has the ability to learn layout knowledge and fuse various multi-modal information.

Reading Order Prediction. The serialization result consists of several text segments, including a series of words and 2D coordinates. Based on the knowledge, we organize the input words in proper reading order. However, there is no explicit boundary between text segments in the input sequence received by the transformer. To make the model understand the relationship between layout knowledge and reading order and still work well when receiving input in inappropriate order, we propose Reading Order Prediction (ROP) and hope the attention matrix $\hat{\mathbf{A}}$ carries the knowledge about reading order. In this way, we give $\hat{A}_{ij}$ an additional meaning, i.e., the probability that the j-th token is the next token of the i-th token. Besides, the ground truth is a 0-1 matrix G, where 1 indicates that there is a reading order relationship between the two tokens and vice versa. For the end position, the next token is itself. In pre-training, we calculate the loss with Cross-Entropy:

$$\mathcal{L}_{\text{ROP}} = -\sum_{0 \leq i < N} \sum_{0 \leq j < N} G_{ij} \log(\hat{A}_{ij}). \tag{12}$$

Replaced Region Prediction. In visual encoder, each document image is processed into a sequence with a fixed length HW. To enable the model perceive fine-grained correspondence between image patches and text, with the help of layout knowledge, we propose Replaced Region Prediction (RRP). Specifically, 10% of the patches are randomly selected and replaced with a patch from another image, the processed image is encoded by the visual encoder and input into the multi-modal transformer.

[4]The schematic workflow is shown in Appendix A.2

Then, the [CLS] vector output by the transformer is used to predict which patches are replaced. So the loss of this task is:

$$\mathcal{L}_{\text{RRP}} = -\sum_{0 \le i < HW} [G_i \log(P_i) + (1 - G_i) \log(1 - P_i)], \quad (13)$$

where G_i is the golden label of replaced patches, P_i is the normalized probability of prediction.

Masked Visual-Language Modeling. Similar to masked language modeling (MLM), the objective of masked visual-language modeling (MVLM) is to recover the masked text token based on its text context and the whole multi-modal clues.

Text-Image Alignment. Besides the image-side cross-modal task RRP, we also adopt Text-Image Alignment (TIA), as a text-side task, to help the model learn the spatial correspondence between image regions and coordinates of bounding box. Here, some text lines are randomly selected, and their corresponding regions are covered on the document image. Then, a classification layer is introduced to predict whether each text token is covered.

To sum up, the final pre-training objective is:

$$\mathcal{L} = \mathcal{L}_{\text{ROP}} + \mathcal{L}_{\text{RRP}} + \mathcal{L}_{\text{MVLM}} + \mathcal{L}_{\text{TIA}} \quad (14)$$

4 Experiments

4.1 Datasets

For the fairness of experiments, we only use *layout knowledge enhanced serialization to rearrange the reading order of pre-training data*, which means that ERNIE-Layout receives the same input as the compared methods in the fine-tuning phase.

Pre-training. Following popular choice in VrDU, we crawl the homologous data of IIT-CDIP Test Collection (Lewis et al., 2006) from Tabacco website, which contains over 30 million scanned document pages, and randomly select 10 million pages from them as the pre-training data.

Fine-tuning. We carry out broad experiments on various downstream VrDU tasks and datasets. For the *key information extraction* task, we select FUNSD (Jain and Wigington, 2019), CORD (Park et al., 2019), SROIE (Huang et al., 2019), and Kleister-NDA (Graliński et al., 2021) as the evaluation datasets. For the *document question answering* task, the DocVQA (Mathew et al., 2021) dataset is selected. For the *document image classification* task, we select the RVL-CDIP (Harley et al., 2015) dataset. Table 1 shows the brief statistics of them and more details are included in Appendix A.3.

Dataset	#Field	#Train	#Dev	#Test
FUNSD	4	149	-	50
CORD	30	800	100	100
SROIE	4	626	-	347
Kleister-NDA	4	254	83	203
RVL-CDIP	16	320K	40K	40K
DocVQA	-	39K	5K	5K

Table 1: Statistics of datasets for downstream tasks

Dataset	Epoch	Weight Decay	Batch
FUNSD	100	-	2
CORD	30	0.05	16
SROIE	100	0.05	16
Kleister-NDA	30	0.05	16
RVL-CDIP	20	0.05	16
DocVQA	6	0.05	16

Table 2: Hyper-parameters for downstream tasks

4.2 Settings

Pre-training. ERNIE-Layout has 24 transformer layers with 1024 hidden units and 16 attention heads. The maximum sequence length of textual tokens is 512 the sequence length of visual tokens is 49. The transformer is initialized from RoBERTa-large (Liu et al., 2019), and the visual encoder takes Faster-RCNN (Ren et al., 2015) as the initialized model. The rest parameters are randomly initialized. We use Adam (Kingma and Ba, 2014) as the optimizer, with a learning rate of 1e-4 and a weight decay of 0.01. The learning rate is linearly warmed up over the first 10% steps, then linearly decayed to 0. ERNIE-Layout is trained on 24 Tesla A100 GPUs for 20 epochs with a batch size of 576.

Fine-tuning. We solve the *key information extraction* tasks (FUNSD, CORD, SROIE, Kleister-NDA) with a sequence labeling framework and introduce a token-level classification layer to predict the BIO labels. For the *document question answering* task (DocVQA), we follow the extractive question-answering paradigm and build a token-level classifier after the ERNIE-Layout output representation to predict the start and end position of the answer. For the *document image classification* task (RVL-CDIP), the representation of [CLS] is processed by a fully-connected network to predict the document label. ERNIE-Layout is fine-tuned for all the downstream tasks using Adam optimizer, with a learning rate of 2e-5, weight decay of 0.01. Similar to pre-training, the learning rate is linearly warmed up and then linearly decayed. Other hyper-parameters are shown in Table 2.

#	Methods	**FUNSD** (F1)	**CORD** (F1)	**SROIE** (F1)	**Kleister-NDA** (F1)
1	$\text{BERT}_{\text{large}}$ (Liu et al., 2019)	0.6563	0.9025	0.9200	0.7910
2	$\text{RoBERTa}_{\text{large}}$ (Liu et al., 2019)	0.7072	-	0.9280	-
3	$\text{UniLMv2}_{\text{large}}$ (Bao et al., 2020)	0.7257	0.9205	0.9488	0.8180
4	$\text{LayoutLM}_{\text{large}}$ (Xu et al., 2020)	0.7895	0.9493	0.9524	0.8340
5	$\text{TILT}_{\text{large}}$ (Powalski et al., 2021)	-	0.9633	**0.9810**	-
6	$\text{LayoutLMv2}_{\text{large}}$ (Xu et al., 2021)	0.8420	0.9601	<u>0.9781</u>	0.8520
7	$\text{StructuralLM}_{\text{large}}$ (Li et al., 2021a)	<u>0.8514</u>	-	-	-
8	$\text{DocFormer}_{\text{large}}$ (Appalaraju et al., 2021)	0.8455	<u>0.9699</u>	-	<u>0.8580</u>
9	$\text{ERNIE-Layout}_{\text{large}}$ (ours)	**0.9312**	**0.9721**	0.9755	**0.8810**

Table 3: Results (Entity-level F1 score) of ERNIE-Layout and previous methods on the *Key Information Extraction* task (*FUNSD*, *CORD*, *SROIE*, *Kleister-NDA*). The highest and second-highest scores are bolded and underlined.

#	Methods	Fine-tuning set	ANLS	△ANLS
1	$\text{BERT}_{\text{large}}$ (Liu et al., 2019)	train	0.6768	
2	$\text{RoBERTa}_{\text{large}}$ (Liu et al., 2019)	train	0.6952	
3	$\text{UniLMv2}_{\text{large}}$ (Bao et al., 2020)	train	0.7709	
4	$\text{LayoutLM}_{\text{large}}$ (Xu et al., 2020)	train	0.7259	
5	$\text{TILT}_{\text{large}}$ (Powalski et al., 2021)	-	**0.8705**	
6	$\text{StructuralLM}_{\text{large}}$ (Li et al., 2021a)	-	0.8349	
7a	$\text{LayoutLMv2}_{\text{large}}$ (Xu et al., 2021)	train	0.8348	+ 0.0639 (#3)
7b	$\text{LayoutLMv2}_{\text{large}}$ (Xu et al., 2021)	train + dev	<u>0.8529</u>	+ 0.0820 (#3)
8a	$\text{ERNIE-Layout}_{\text{large}}$ (ours)	train	0.8321	+ <u>0.1369</u> (#2)
8b	$\text{ERNIE-Layout}_{\text{large}}$ (ours)	train+dev	0.8486	+ **0.1534** (#2)
9	$\text{ERNIE-Layout}_{\text{large}}$ (leaderboard)	train+dev	**0.8841**	

Table 4: Results (Average Normalized Levenshtein Similarity, ANLS) of ERNIE-Layout and previous methods on the *Document Question Answering* task (*DocVQA*). "-" means the fine-tuning set is not clearly described in the original paper. △ANLS means ANLS difference between the multi-modal model and its corresponding text-only model, where ERNIE-Layout is initialized from RoBERTa and LayoutLMv2 is initialized from UniLMv2.

4.3 Results

Key Information Extraction. Table 3 shows the results on four datasets, in which we utilize entity-level F1 score to evaluate these sequence labeling tasks. ERNIE-Layout achieves new state-of-the-art on FUNSD, CORD, Kleister-NDA, and competitive performance on SROIE. It is worth mentioning that, in the FUNSD, ERNIE-Layout obtains a significant and stable improvement of 7.98% (with a standard deviation 0.0011), compared to the previous best results. The above phenomena are enough to verify the effectiveness of our design philosophy that mining and utilizing layout knowledge in document pre-training models.

Document Question Answering. Table 4 lists the Average Normalized Levenshtein Similarity (ANLS) score on DocVQA. Compared with all of the text-only baselines and best-performing multi-modal models, our method achieves competitive results and maximum performance improvement. Note that LayoutLMv2(#7) is developed based on UniLMv2(#3), a model with powerful question-answering ability and even beat the multi-model model LayoutLM (#4) on the task. Unfortunately, UniLMv2 does not open any pre-training code or pre-trained model, and we can only use the parameters of RoBERTa to initialize our ERNIE-Layout. Nevertheless, we are surprised that ERNIE-Layout

#	Methods	Accuracy
1	$\text{BERT}_{\text{large}}$ (Liu et al., 2019)	0.8992
2	$\text{RoBERTa}_{\text{large}}$ (Liu et al., 2019)	0.9011
3	$\text{UniLMv2}_{\text{large}}$ (Bao et al., 2020)	0.9020
4	$\text{LayoutLM}_{\text{large}}$ (Xu et al., 2020)	0.9443
5	$\text{TILT}_{\text{large}}$ (Powalski et al., 2021)	0.9552
6	$\text{LayoutLMv2}_{\text{large}}$ (Xu et al., 2021)	0.9564
7	$\text{StructuralLM}_{\text{large}}$ (Li et al., 2021a)	<u>0.9608</u>
8	$\text{DocFormer}_{\text{large}}$ (Appalaraju et al., 2021)	0.9550
9	$\text{ERNIE-Layout}_{\text{large}}$ (ours)	**0.9627**

Table 5: Results (Accuracy) of ERNIE-Layout and previous methods on the *Document Image Classification* task (*RVL-CDIP*).

#	MVLM	TIA	RRP	ROP	SADA	SASA	FUNSD	CORD
1	√						0.8712	0.9513
2	√	√					0.8753	0.9555
3†	√	√	√				0.8848	0.9565
4†	√	√	√	√			0.8978	0.9603
5†	√	√	√	√	√		0.9241	0.9673
6	√	√	√	√		√	0.9128	0.9658

Table 6: Performance analysis with different pre-training tasks and attention mechanisms, in which SADA refers to the spatial-aware disentangled attention in ERNIE-Layout, SASA refers to the spatial-aware self-attention proposed by LayoutLMv2. † indicates the added module is proposed in this paper.

#	Serialization Module	FUNSD	CORD
1	w/ Raster-Scan	0.9128	0.9658
2	w/ Layout-Parser	0.9143	0.9671
3	w/ Document-Parser	0.9171	0.9678

Table 7: Performance analysis with different serialization modules, in which Raster-Scan means serialization with vanilla OCR results, while Layout-Parser and Document-Parser arrange the recognized words with the help of layout knowledge.

brings an exciting performance improvement to the backbone (almost double the increase of LayoutLMv2). Furthermore, we achieve top-1 on the DocVQA leaderboard with ensemble.

Document Image Classification. Table 5 shows the classification accuracy on RVL-CDIP, which again confirms the effectiveness of ERNIE-Layout in general document understanding. Unlike these key information extraction or document question answering tasks focusing on multi-modal semantic understanding, document image classification requires a macro perception of text content and document layout. Although our pre-training tasks pay attention to the fine-grained cross-modal matching, ERNIE-Layout still refreshes the best performance of the cross-grained task.

4.4 Analysis

We further conduct analysis experiments to study the effectiveness of the proposed pre-training tasks, attention mechanisms, and the serialization modules. We select FUNSD and CORD as the evaluation datasets, keep all ablations sharing the same hyper-parameter settings, and report the average number of five runs with different random seeds.

Effectiveness of Pre-training Tasks. In this experiment, we start with the basic MVLM task to implement baseline models (#1), and integrate new tasks step by step until the final model contains all four pre-training tasks (#5). From Table 6, we observe that RRP brings an improvement of 0.95% on FUNSD, demonstrating the benefit of fine-grained cross-modal interaction. When incorporating ROP, the performance of FUNSD is further increased by 1.3%. We consider that ROP facilitates the model to learn a better representation that contains the reading order knowledge.

Effectiveness of Attention Mechanisms. LayoutLMv2 (Xu et al., 2021) initially proposes spatial-aware self-attention to consider layout features in attention calculation, and many subsequent methods follow this idea. From Table 6, we find that adopting such a mechanism can boost the performance of downstream tasks (#4 v.s. #6). Meanwhile, disentangling attention into the position and content parts is another efficient solution to earn further performance gains (#5 v.s. #6).

Effectiveness of Serialization Modules. Here we explore the impact of using different serialization modules on the downstream VrDU tasks. As shown in Table 7, with the layout-knowledge based serialization modules (#2, #3), the model could achieve better performances (even without the disentangled attention). We attribute the improvement to the fact that, although the advanced serialization is not used for fine-tuning datasets, the model has the ability to understand the proper reading order of documents after pre-training.

5 Conclusion

In this paper, we propose ERNIE-Layout, to integrate layout knowledge into document pre-training models from two aspects: serialization and attention. ERNIE-Layout attempts to rearrange the recognized words of documents, which achieves considerable improvement on downstream tasks over the original raster-scanning order. Besides, we also design a novel attention mechanism to help ERNIE-

Layout build better interaction between text/image and layout features. Extensive experiments demonstrate the effectiveness of ERNIE-Layout, and various analyses show the impact of different utilization of layout knowledge on VrDU tasks.

Acknowledgements

This work was supported by Baidu Inc., the NSFC projects (No. 62072399), Chinese Knowledge Center for Engineering Sciences and Technology, and Artificial Intelligence Research Foundation of Baidu Inc., MoE Engineering Research Center of Digital Library.

References

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

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, et al. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pages 642–652.

Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marçal Rusinol, Minesh Mathew, CV Jawahar, Ernest Valveny, and Dimosthenis Karatzas. 2019. Icdar 2019 competition on scene text visual question answering. In *Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1563–1570.

Mengli Cheng, Minghui Qiu, Xing Shi, Jun Huang, and Wei Lin. 2020. One-shot text field labeling using attention and belief propagation for structure information extraction. In *Proceedings of the 28th ACM International Conference on Multimedia (MM)*, pages 340–348.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, 29:3504–3514.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 4171–4186.

Filip Graliński, Tomasz Stanisławek, Anna Wróblewska, Dawid Lipiński, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemysław Biecek. 2021. Kleister: A novel task for information extraction involving long documents with complex layout. In *Proceedings of the 2021 International Conference on Document Analysis and Recognition (ICDAR)*.

Zhangxuan Gu, Changhua Meng, Ke Wang, Jun Lan, Weiqiang Wang, Ming Gu, and Liqing Zhang. 2022. Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4583–4592.

Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. 2015. Evaluation of deep convolutional nets for document image classification and retrieval. In *Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 991–995.

Bin He, Di Zhou, Jinghui Xiao, Xin Jiang, Qun Liu, Nicholas Jing Yuan, and Tong Xu. 2020. Bert-mk: Integrating graph contextualized knowledge into pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2281–2290.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *Proceedings of the 2021 International Conference on Learning Representations (ICLR)*, pages 1–21.

Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. 2019. Icdar2019 competition on scanned receipt ocr and information extraction. In *Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520.

Rajiv Jain and Curtis Wigington. 2019. Multimodal document image classification. In *Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 71–77.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics (TACL)*, 8:64–77.

Anoop R Katti, Christian Reisswig, Cordula Guder, Sebastian Brarda, Steffen Bickel, Johannes Höhne, and Jean Baptiste Faddoul. 2018. Chargrid: Towards understanding 2d documents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4459–4469.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

David Lewis, Gady Agam, Shlomo Argamon, Ophir Frieder, David Grossman, and Jefferson Heard. 2006. Building a test collection for complex document information processing. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR)*, pages 665–666.

Chenliang Li, Bin Bi, Ming Yan, Wei Wang, Songfang Huang, Fei Huang, and Luo Si. 2021a. Structurallm: Structural pre-training for form understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6309–6318.

Yulin Li, Yuxi Qian, Yuechen Yu, Xiameng Qin, Chengquan Zhang, Yan Liu, Kun Yao, Junyu Han, Jingtuo Liu, and Errui Ding. 2021b. Structext: Structured text understanding with multi-modal transformers. In *Proceedings of the 29th ACM International Conference on Multimedia (MM)*, pages 1912–1920.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-bert: Enabling language representation with knowledge graph. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2901–2908.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Processing of the 33rd Conference on Neural Information Processing Systems (NeurIPS)*, pages 1–11.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2200–2209.

Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. 2019. Cord: A consolidated receipt dataset for post-ocr parsing. In *Workshop on Document Intelligence at NeurIPS 2019*.

Rafał Powalski, Łukasz Borchmann, Dawid Jurkiewicz, Tomasz Dwojak, Michał Pietruszka, and Gabriela Pałka. 2021. Going full-tilt boogie on document understanding with text-image-layout transformer. In *Proceedings of the 2021 International Conference on Document Analysis and Recognition (ICDAR)*, pages 732–747.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, pages 1–9.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Processing of the 29th Conference on Neural Information Processing Systems (NeurIPS)*, pages 1–9.

Ritesh Sarkhel and Arnab Nandi. 2019. Deterministic routing between layout abstractions for multi-scale classification of visually rich documents. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. Ernie: Enhanced representation through knowledge integration. *arXiv preprint arXiv:1904.09223*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Processing of the 31st Conference on Neural Information Processing Systems (NeurIPS)*, pages 5998–6008.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuan-Jing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. 2021a. K-adapter: Infusing knowledge into pre-trained models with adapters. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1405–1418.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021b. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics (TACL)*, 9:176–194.

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

Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. 2021. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2579–2591.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1192–1200.

Xiao Yang, Ersin Yumer, Paul Asente, Mike Kraley, Daniel Kifer, and C Lee Giles. 2017. Learning to extract semantic structure from documents using mul-

timodal fully convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5315–5324.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics (NAACL)*, pages 1480–1489.

Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2021. Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 3208–3216.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. Ernie: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1441–1451.

A Appendix

A.1 More Details about Document-Parser

The Document-Parser assembles multiple modules such as document-specific OCR, Layout-Parser, and Table-Parser, in which the Layout Parser and Table Parser modules are crucial for incorporating layout knowledge in ERNIE-Layout.

An important preprocessing step for document understanding is serializing the extracted document tokens. The popular method for this serialization is performed directly on the output results of OCR in raster-scanning order and is sub-optimal though simple to implement. With the Layout-Parser and Table-Parser in the Document-Parser toolkit, the order of the tokens will be further rearranged according to the layout knowledge. During the parsing processing, the tables and figures are detected as spatial layouts, and the free texts are processed by paragraph analysis, combining heuristics and detection models to get the paragraph layout information and the upper-lower boundary relationship.

To validate the effectiveness of our method, we use an open-sourced language model GPT-2 (Radford et al., 2019), to calculate the PPL of the serialized token sequence by the raster-scanning order and Document-Parser respectively. Since documents with complex layouts only account for a small proportion of the total documents, in a test of 10,000 documents, the average PPL only drops

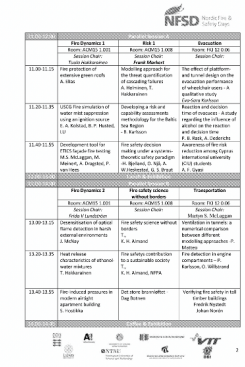
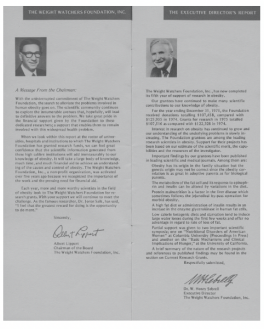
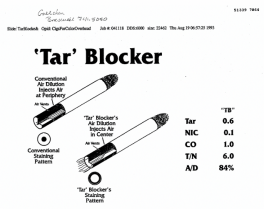
Document Page	RS	DP
	100.39	67.98
	98.99	42.02
	146.66	76.87
	70.12	25.61
	219.47	170.54

Table 8: The PPL of serialized token sequence with different methods. RS refers to the Raster-scanning order and DP refers to the order with Document-Parser.

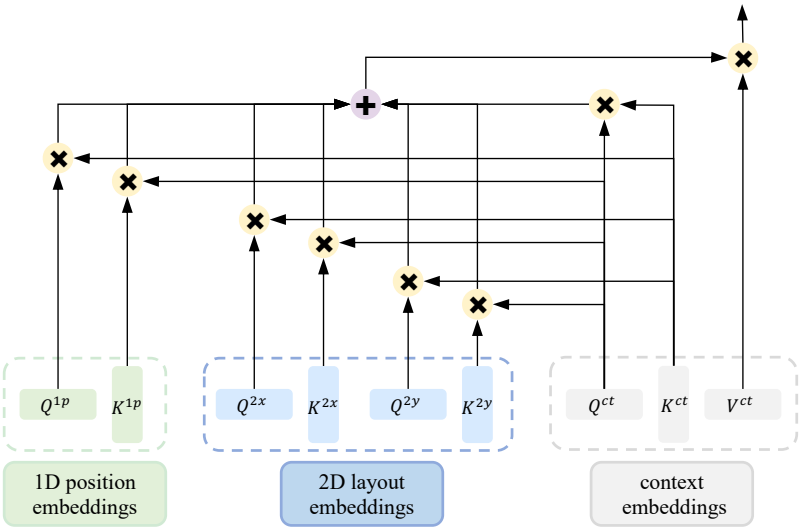


Figure 3: The internal working principle of spatial-aware disentangled attention.

Parallel Session A		
Fire Dynamics 1	**Risk 1**	**Evacuation**
Room: ACM15 1.001	Room: ACM15 1.008	Room: FKJ 12 0.06
Session Chair: *Tuula Hakkarainen*	*Session Chair:* ***Frank Markert***	*Session Chair:*

Figure 4: The example of a document with a complex layout. The serialization result with the raster-scanning order is "... *Session Chair: Session Chair: Session Chair: Tuula Hakkarainen ...*", while serialization with Document-Parser is "... *Session Chair: Tuula wz Session Chair: Frank Markert ...*", which is more consistent with human reading habits.

about 1 point. However, for these documents with complex layouts, as shown in Table 8, Document-Parser shows great advantages. An example is shown in Figure 4, which is extracted from the third image in Table 8. to show the sequence serialized by Raster-Scan and Document-Parser.

A.2 More Details about Multi-modal Transformer

Section 3.3 describes the proposed spatial-aware disentangled attention for the multi-modal transformer through formulas. To facilitate intuitive understanding, we also supplement the flow chart of calculation in Figure 3.

A.3 More Details about Experiments

A.3.1 Finetuning Datasets

FUNSD (Jain and Wigington, 2019) is a dataset for form understanding on noisy scanned documents that aims at extracting values from forms, which comprises 199 real, fully annotated, scanned forms. The training set contains 149 samples, and the test set contains 50 samples. We use the official OCR annotations. Following previous methods, we adopt entity-level F1 as the evaluation metric. Like StructralLM (Li et al., 2021a), we use the cell-level layout information when fine-tuning.

CORD (Park et al., 2019) is a consolidated dataset for receipt parsing as the first step towards post-OCR parsing tasks. CORD consists of thousands of Indonesian receipts, including images, box/text annotations for OCR, and multi-level semantic labels for parsing. The training, validation, and test sets contain 800, 100, and 100 receipts, respectively. We use the official OCR annotations and entity-level F1 as the evaluation metric.

SROIE (Huang et al., 2019) is a scanned receipts OCR and key information extraction dataset, which covers important aspects related to the analysis of scanned receipts. The training and test set contain 626 and 347 samples, respectively. This task requires the model to extract values from each receipt of four predefined keys: company, date, address, and total. We use the official OCR annotations and entity-level F1 as the evaluation metric.

Kleister-NDA (Graliński et al., 2021) is provided for key information extraction task, which involves a mix of scanned and born-digital long formal documents. The training, valid, and test sets contain 254, 83, and 203 samples, respectively. Due to the test set is not publicly available, we report the entity-level F1 score on the validation set, which is computed by the official evaluation tools[5]. The task aims to extract values of four predefined

[5]https://gitlab.com/filipg/geval

Methods	**FUNSD** (F1)	**CORD** (F1)	**SROIE** (F1)	**Kleister-NDA** (F1)	**DocVQA** (ANLS)	RVL-CDIP (Acc)
$\text{LayoutLM}_{\text{base}}$ (2020)	0.7866	0.9472	0.9438	0.8270	0.6979	0.9442
$\text{TILT}_{\text{base}}$ (2021)	-	0.9511	**0.9765**	-	**0.8392**†	0.9525
$\text{LayoutLMv2}_{\text{base}}$ (2021)	0.8276	0.9495	0.9625	<u>0.8330</u>	0.7808	0.9525
$\text{DocFormer}_{\text{base}}$ (2021)	<u>0.8334</u>	<u>0.9633</u>	-	-	<u>0.7878</u>	**0.9617**
$\text{ERNIE-Layout}_{\text{base}}$ (ours)	**0.9028**	**0.9661**	<u>0.9719</u>	**0.8740**	0.7758	<u>0.9581</u>

Table 9: Results of ERNIE-Layout (base-level model) and previous methods on various downstream VrDU tasks. † marks the results without any description of fine-tuning set (train or train+dev), The bold and underlined scores indicate the best and second results, respectively.

keys: date, jurisdiction, party, and term.

RVL-CDIP (Harley et al., 2015) is a document classification dataset consisting of grayscale document images. The training, validation, and test sets contain 320000, 40000, and 40000 document images, respectively. The document images are categorized into 16 classes, with 25000 images per class. We use Microsoft OCR tools to extract text and layout information from document images, and the evaluation metric is classification accuracy.

DocVQA (Mathew et al., 2021) is a dataset for Visual Question Answering (VQA) on document images. The dataset consists of 50000 questions defined on 12767 document images. The document images are split into the training set, validation set, and test set with the ratio of 8:1:1. We use the Microsoft OCR tools to extract the texts and layouts from document images. The task aims to predict the start and end position of the answer span. Average Normalized Levenshtein Similarity (Biten et al., 2019) is used as the evaluation metric.

A.3.2 Baselines

In the experiment, we compare ERNIE-Layout with two groups of recent models: **text-only models** (BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), UniLMv2 (Bao et al., 2020)) and **multi-modal models** (LayoutLM (Xu et al., 2020), LayoutLMv2 (Xu et al., 2021), TILT (Powalski et al., 2021), StructuralLM (Li et al., 2021b), DocFormer (Appalaraju et al., 2021)). Note that LayoutLM is initialized from BERT, LayoutLMv2 is initialized from UniLMv2, TILT is initialized from T5, StructuralLM and our ERNIE-Layout are initialized from RoBERTa.

A.3.3 Results with RoBERTa-base

In the main content, we leverage RoBERTa-large to initialize ERNIE-Layout and compare it with the previous same-level model. Here, we also compare the base-level model with 12 transformer layers (768 hidden state and 12 attention heads), that is, initializing ERNIE-Layout with RoBERTa-base. We omit StructuralLM since it does not release the parameters and performances of its base model. From the results in Table 9, it is easy to observe a similar phenomenon with $\text{ERNIE-Layout}_{\text{large}}$: $\text{ERNIE-Layout}_{\text{base}}$ also achieves significant performance improvement on various VrDU tasks, especially in FUNSD and Kleister-NDA, but slightly poor in DocVQA (detailed analysis and further exploration have been given in Section 4.3). By the way, We are also pleasantly surprised to find that $\text{ERNIE-Layout}_{\text{base}}$ even beats some large-level model in kinds of datasets (e.g., FUNSD, CORD, Kleister-NDA, RVL-CDIP).