Relation-Rich Visual Document Generator for Visual Information Extraction

Zi-Han Jiang[1], Chien-Wei Lin[1], Wei-Hua Li[1], Hsuan-Tung Liu[2], Yi-Ren Yeh[3], Chu-Song Chen[1]
[1]National Taiwan University, [2]E.SUN Financial Holding Co., Ltd., [3]National Kaohsiung Normal University
{r12922082,r11922212,d12922009,chusong}@csie.ntu.edu.tw,
ahare-18342@esunbank.com.tw, yryeh@nknu.edu.tw

Abstract

Despite advances in Large Language Models (LLMs) and Multimodal LLMs (MLLMs) for visual document understanding (VDU), visual information extraction (VIE) from relation-rich documents remains challenging due to the layout diversity and limited training data. While existing synthetic document generators attempt to address data scarcity, they either rely on manually designed layouts and templates, or adopt rule-based approaches that limit layout diversity. Besides, current layout generation methods focus solely on topological patterns without considering textual content, making them impractical for generating documents with complex associations between the contents and layouts. In this paper, we propose a ***R****elation-r****I****ch visual* ***D****ocument* ***GE****nerator (RIDGE) that addresses these limitations through a two-stage approach: (1) Content Generation, which leverages LLMs to generate document content using a carefully designed Hierarchical Structure Text format which captures entity categories and relationships, and (2) Content-driven Layout Generation, which learns to create diverse, plausible document layouts solely from easily available Optical Character Recognition (OCR) results, requiring no human labeling or annotations efforts. Experimental results have demonstrated that our method significantly enhances the performance of document understanding models on various VIE benchmarks.*

1. Introduction

VDU has attracted significant attention in knowledge-centric applications. While conventional document pre-trained models [13, 16, 37, 41, 46, 47] have demonstrated outstanding performance on specific downstream tasks, their practical applicability is often constrained by the necessity of task- or dataset-specific fine-tuning. This has prompted a shift in research focus towards the utilization of LLMs and MLLMs.

To perform document analysis, segmented text entities are the basic elements in a visual document. Entity Linking (links between entities) and Entity Category (attributes or semantic classification labels of entities) are the two main types of information that are critical to VIE. To learn a document-analysis model, a major difficulty is the lack of labeled training data, where the labels can be of unary relation (containing n entities) or binary relation (containing $O(n^2)$ links between entity pairs). However, the effort for labeling $O(n^2)$ labels is huge, which hinders the development of learning effective models for a comprehensive understanding of the diverse forms of visual documents.

LLMs and MLLMs possess generalization capability and emergent properties resulting from training on extensive corpora, allowing them to perform effectively on document-related tasks. Recent models have shown promising results in document-oriented visual question answering, such as Qwen2-VL [44] approaching human-level performance on DocVQA [34]. However, significant challenges persist in extracting information from relation-rich (*e.g.*, forms of various formats) or semi-structured documents (*e.g.*, invoices and receipts). The variability complicates the process of accurately identifying and linking key-value pairs. A general extraction rule cannot be easily established, affecting model accuracy and requiring extensive training data to enhance the generalization capabilities.

In this paper, we introduce a method to synthesize relation-rich visual documents. The generated documents exhibit highly diverse layouts, arise not only from the document's content but also from the variability across different document sources. Our content-rich generative approach leverages LLMs in two steps. First, we use ChatGPT [38] to generate text with entity-category and entity-link information in a hierarchy. Second, we adopt self-supervised learning to build a model that can place the generated text into appropriate bounding box locations with reasonable layouts. Unlike traditional layout generation solutions focusing mainly on producing bounding boxes without relationally meaningful textual content [19, 23, 28, 40], our generated content-and-relation-rich documents can boost the performance of VDU. We demonstrate an overview of existing datasets and our approach as follows.

	Open-set		Closed-set				
Dataset	FUNSD [20]	XFUND [48]	CORD [35]	EPHOIE [43]	POIE [26]	SROIE [17]	DocILE [39]
#Img.	199	199 per lang.	1000	1494	3000	1000	real: 6680 syn:100K*
#Keys	Free	Free	30	10	21	4	55

Table 1. Dataset statistics of VIE tasks. * represents the synthetic data are generated by 100 templates.

VIE Datasets. Existing datasets for VIE falls into two categories: open-set and closed-set. Open-set datasets provide annotations for entity-level categories and entity linking. For example, FUNSD [20], an English form-type document dataset, classifies entities into four categories (header, question, answer, and other), and also provides links between questions and answers. In contrast, closed-set datasets define a set of specific keys and extract corresponding values. For example, CORD [35], a consolidated receipt dataset, includes 30 categories, such as menu name and total price. Compared with closed-set datasets, the number of images in open-set datasets is quite limited, as shown in Tab. 1. For instance, FUNSD contains only 199 document images, while XFUND [48], a form-type document dataset covering 7 languages, also has only 199 document images per language. Additionally, data collection poses another challenge, as these types of documents may contain personal information, making them difficult to access publicly. This shortage further highlights the pressing need for a tool that can automatically generate large volumes of this type of data, complete with annotations and document images.

Overview of Our Approach. We propose RIDGE, a relation-rich visual document generator capable of producing meaningful document content that inherits complex hierarchical relationships between entities, while maintaining a highly diverse layout. To achieve this, we introduce a two-stage generation pipeline. **I. Content Generation.** In this work, we primarily focus on synthesizing open-set VIE data to address its scarcity. We leverage LLMs to generate document content in our meticulously designed *Hierarchical Structure Text (HST)* format, where the generated output can be directly parsed into segmented text entities with their category and linking annotations. With content and annotations now prepared, we advance to the next stage. **II. Content-driven Layout Generation.** Research [6, 9, 31, 40] has shown that LLMs demonstrate layout planning abilities attributed to their training on layout-specific sources like HTML code of mobile app interfaces or websites. Inspired by this, we leverage LLMs to generate document layouts through self-supervised learning. Our training data requires only OCR results (segmented text entities and their bounding box coordinates) from documents that are readily obtainable through off-the-shelf OCR tools. Although we do not provide entity category or linking annotations to the model, it can independently learn content-layout relationships through self-supervision, generating a reasonable layout that aligns with the annotations produced in the previous stage. Additionally, we introduce a training paradigm called *Hierarchical Structure Learning* that further leverages HST for reinforcing document understanding models' comprehension. Experimental results have shown that RIDGE significantly enhances the performance of document understanding models on various VIE benchmarks.

2. Related Work

Synthetic Document Generators. Due to the high cost of manual annotation and the limited availability of sensitive documents, recent research has focused on automatically synthesizing documents to meet the substantial demands of machine learning applications. DocSynth [2] uses a Generative Adversarial Network (GAN) to generate synthetic document images based on manually-designed layouts. Due to the low resolution (128×128) of the generated images, the method can generate only blurry shadows for text and fails to capture detailed document content, resulting the generated synthetic documents only suitable for document layout analysis. SynthDoG [25] aims to generate documents for OCR training by selecting text content from Wikipedia at random and rendering it onto a natural or document-textured image. Its layouts are generated with a rule-based algorithm that merely places random grids on the image, causing content to split across grids in ways that may appear unnatural. DocILE [39] generates realistic content with key information annotations; however, its synthetic method is rule-based, relying on only 100 annotated layout templates. It primarily alters content or applies style changes to font and borders, which limits its diversity and suitability for general training. Existing methods are limited in their ability to produce realistic content and diverse layouts, both of which are essential for robust VIE training.

Document Layout Generation. Automatic layout generation has emerged as a crucial task across diverse application domains, spanning documents, natural images, user interfaces, and posters. Early studies [1, 22, 24, 28, 36] employ GANs and VAEs to learn layout generation through reconstruction, while Jiang *et al*. [23] reformulate this as a sequence-to-sequence problem using transformer architectures. Through advances in diffusion techniques, some studies [12, 18, 19] demonstrate the effectiveness of treating layout generation as a diffusion process. More recently, researchers [6, 9, 30, 31, 40] have begun to utilize LLMs for this task, harnessing their inherent layout understanding. While most methods focus on generating layout (bounding box and category) independent of content, some [4, 21, 27] have incorporated text content with layout planning, primarily focusing on visual design contexts such as posters and graphic designs. These approaches typically handle sparse text elements (keywords or short phrases) arranged around

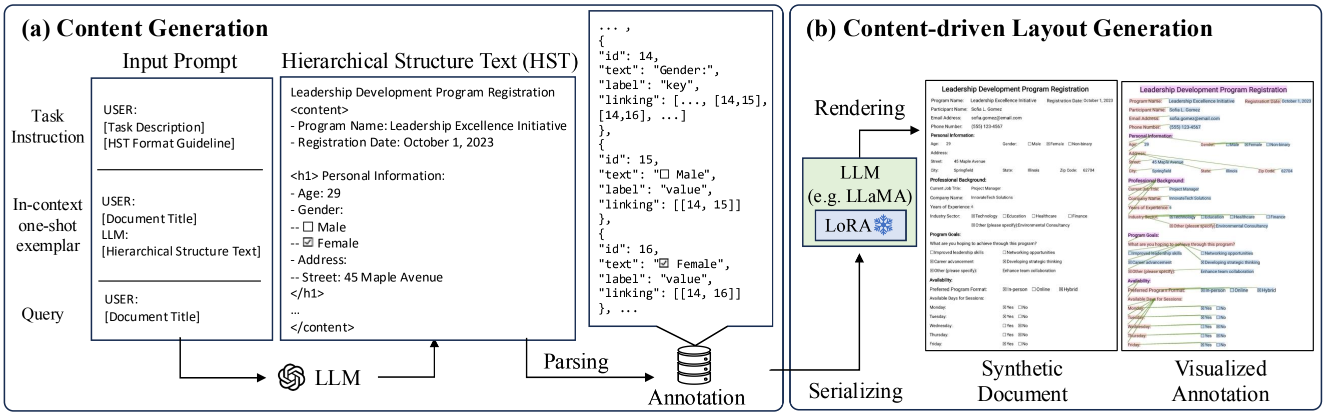

Figure 1. Overview of RIDGE, including (a) Content Generation and (b) Content-driven Layout Generation. In the visualized annotation, purple represents the header, red represents the key, blue represents the value, and green lines represent entity linking.

visual components, rather than documents with dense textual content and complex structural relationships. Among document-layout-related research, most studies predominantly concentrate on academic paper formats due to the accessibility of training data. These methods address solely layout generation, without considering the specific textual content to populate these layouts. Given that font sizes in documents typically follow certain alignment principles, applying such pre-generated layouts would restrict users to fixed-length text within each layout bounding box—a constraint that makes these methods impractical for generating documents with meaningful textual content.

Consequently, existing studies can only be applied to training document layout analysis models, rather than directly supporting VIE tasks, as the latter requires models to develop robust comprehension of complex content-layout relationships. Moreover, previous methods rely heavily on supervised training with layout element categories, which is impractical for relation-rich documents due to limited publicly available annotations. In contrast, our proposed self-supervised method requires only readily available OCR results for training, making it more practical and scalable for real-world applications.

3. Method

This section elaborates on RIDGE, a novel method for synthesizing documents with realistic content and diverse layouts. It employs a two-stage pipeline: **content generation** for text generation and annotation parsing, and **content-driven layout generation** for spatial arrangement of document entities, as shown in Fig. 1.

In this work, we aim to generate open-set VIE dataset consisting of document images and their corresponding annotations, including entity categories and linking, similar to FUNSD. Formally, a document $D = \{e_i\}_{i=1}^{N}$ is composed of entities, where N is the number of entities in the document. Each entity $e = (id, t, b, c, l)$, where id is the entity's unique identifier, t denotes its textual content, $b = (x_{\text{left}}, y_{\text{top}}, x_{\text{right}}, y_{\text{bottom}})$ specifies its bounding box coordinates, $c \in \{\text{header}, \text{key}, \text{value}, \text{other}\}$ indicates its semantic category; and l contains a list of links with other entities, where a link is represented as $[id_{\text{key}}, id_{\text{value}}]$.

3.1. Content Generation

As shown in Fig. 1(a), given a document title, LLMs generate corresponding detailed content of the document in the format of Hierarchical Structure Text (HST). The format is structured as follows: (1) the entire document is wrapped within `<content>` tags after the document title, (2) paragraphs are organized using header tags (`<h1>`, `<h2>`, etc.), where text immediately following each header tag is treated as the paragraph's title, and (3) the content is organized in a tree structure through two patterns—using colon (':') for direct key-value pairs in the same line (e.g., "Name: John"), and using different numbers of hyphens ('-') to indicate hierarchical nesting relationships.

This structured format enables automatic extraction of entity text, categories, and linking relationships. To help LLMs get a more comprehensive understanding of HST, we give a detailed task instruction and an in-context one-shot exemplar for LLMs to reference. For detailed prompts, please refer to the supplementary material.

3.2. Content-driven Layout Generation (CLGM)

In this subsection, we introduce CLGM (see Fig. 2), which is capable of arranging text segments to proper bounding box positions. Instead of traditional category-driven layout generation methods [19, 40], our model is content-driven to tackle nuanced text content in diverse documents. CLGM consists of three parts: *serializing the document layout*, *self-supervised layout learning*, and *document rendering*.

3.2.1. Document Layout Serialization

Since LLMs accept only text input, we need to represent a document layout as a sequence. Previous studies [9, 31, 40]

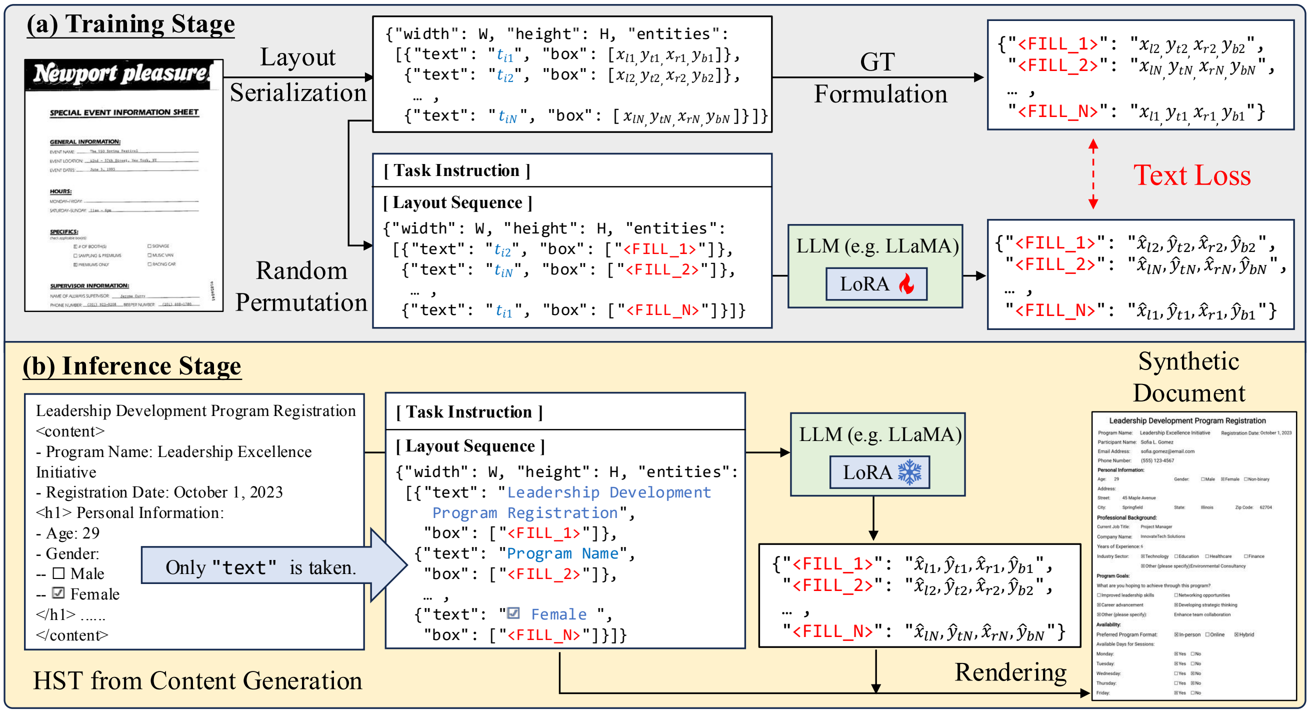

Figure 2. Content-driven Layout Generation Model (CLGM).

have demonstrated that LLMs exhibit strong comprehension in programming languages, due to the inclusion of code-based data in their training corpora. Thus, the layout of a document D can be represented in JSON format as:

$$\{\texttt{"width":W,"height":H,"entities":[}\ \{\texttt{"text":}t_1\texttt{,"box":[}x_{\text{left}_1}, y_{\text{top}_1}, x_{\text{right}_1}, y_{\text{bottom}_1}\texttt{]}\}, ..., \{\texttt{"text":}t_N\texttt{,"box":[}x_{\text{left}_N}, y_{\text{top}_N}, x_{\text{right}_N}, y_{\text{bottom}_N}\texttt{]}\}\texttt{]}\}$$

where W and H indicate the width and height of the layout canvas, and *"entities"* consists of all entities e_i that comprise document D. Unlike previous methods [9, 31, 40] that use HTML or CSS for layout representation, we adopt JSON to reduce input sequence length by eliminating redundant attribute text. In HTML and CSS, an element's bounding box requires repeating attribute text. Our representation consolidates attributes into a single "box" field, which particularly reduces sequence length for complex document layouts containing hundreds of entities.

3.2.2. Layout Self-Supervised Learning

To generate layouts for documents with varying content, we formulate this task as predicting bounding box coordinates for each given text entity. Our approach is inspired by LayoutNUWA [40], a method purely for layout generation without text contents. Our method is content-driven and performs self-supervised learning in a more general way for placing the texts into suitable positions. We use a masking strategy where bounding box coordinates are replaced with special mask tokens <FILL_i> for each e_i. Specifically, let $S(D_{\setminus M})$ denote the input masked sequence to model f_θ:

$$\{\texttt{"width":W,"height":H,"entities":[}\ \{\texttt{"text":}t_1\texttt{,"box":["<FILL\_1>"]}\}, ..., \{\texttt{"text":}t_N\texttt{,"box":["<FILL\_N>"]}\}\texttt{]}\}$$

where $S(\cdot)$ is a serialization function and M denotes the masked bounding boxes. Model f_θ then output a JSON dictionary sequence $S(M)$ that directly maps each mask token to its corresponding coordinates:

$$\{\texttt{"<FILL\_1>":}\ \text{“}x_{\text{left}_1}, y_{\text{top}_1}, x_{\text{right}_1}, y_{\text{bottom}_1}\text{”}, ..., \texttt{"<FILL\_N>":}\ \text{”}x_{\text{left}_N}, y_{\text{top}_N}, x_{\text{right}_N}, y_{\text{bottom}_N}\text{”}\}$$

Content-based layout generation can thus be written as

$$S(M) = f_\theta(S(D_{\setminus M})). \tag{1}$$

Unlike LayoutNUWA which requires LLMs to complete a code template by repeating the entire input with predicted coordinates, our method streamlines this process by focusing solely on predicting bounding box coordinates. This reduces the output sequence length and enhances the efficiency for layout generation. Following the standard autoregressive formulation, we optimize our model using the cross-entropy loss:

$$\mathcal{L} = -\sum_{k=1}^{K} \log P(S(M)^k | S(M)^{<k}, S(D_{\setminus M}), \theta), \tag{2}$$

with K the number of tokens in the output sequence $S(M)$, $S(M)^k$ the k^{th} token in $S(M)$, and θ the model parameters.

Considering the auto-regressive nature of LLMs, we randomly permute the order of input entities during training to prevent the model from learning a fixed layout order, as the document layout inherently has no natural sequence. This randomization strategy brings a crucial benefit—by exposing LLMs to highly unordered text content, which is even challenging for humans to reconstruct, the model learns to infer reasonable relationships between entities solely from their text. This allows our method to generate plausible layouts without requiring category and linking annotations, thus eliminating the need for human annotation effort. During inference, we maintain the order of entities as they appear in HST (*i.e.*, human reading order) to facilitate the model's understanding of document content.

Additionally, we train our model with non-normalized canvas sizes, enabling it to handle diverse layout dimensions directly. It enhances the model's adaptability in practice where documents can vary significantly in size.

3.2.3. Document Rendering

After obtaining the layout coordinates from CLGM, the document rendering module transforms the layout sequence into a visually coherent document image. First, we validate the LLM output to ensure proper text rendering within bounding boxes. Second, we enhance the visual appearance by various styling operations to create realistic documents.

LLM output processing and Styling. When using LLMs to generate structured output, format validation is inevitable. We verify that CLGM's output conforms to valid JSON format and contains all expected mask tokens, excluding samples that fail these criteria. Subsequently, we render text from the left boundary of each bounding box, with font size determined by the box height generated by CLGM. Since different fonts may yield varying text widths, we adaptively extend the box width and shift subsequent entities in the same row when text exceeds its allocated bounds, preserving readability and CLGM's overall layout design. In addition, we enhance the visual appearance through changing fonts, applying bold formatting to text labeled as header, adding different background materials, and incorporating grid lines around entities.

3.3. Hierarchical Structure Learning

Our generated documents with annotations can be directly applied to any document understanding method for training. W.L.O.G., we use MLLMs [32, 44] for training due to their promising performance recently. To exploit our generated HST better, we propose a training paradigm that extensively utilizes HST to enhance models' comprehension. Existing LLM/MLLM-based methods employ text-layout reconstruction learning to enhance models' understanding of document structure. For example, DocOwl-1.5 [14] requires

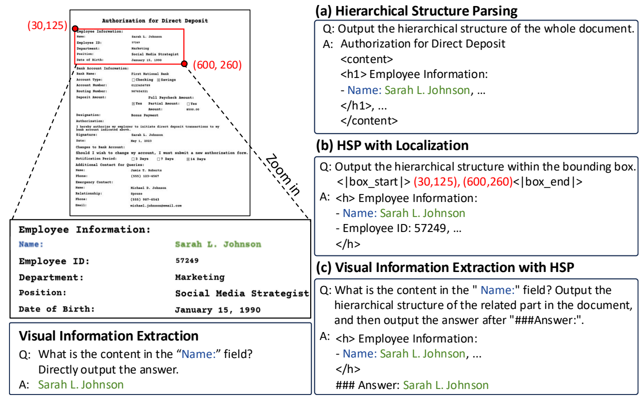

Figure 3. Hierarchical Structure Learning

models to output the document text with newline characters ('\n') and white spaces inserted to mimic the layout of the document image. Additionally, LayoutLLM [33] requires models to reconstruct the document by generating a sequence of entities in the format of "<{box}, {text}>". While these tasks help models learn to associate text with its spatial positions, they may not capture the complex hierarchical relationships inherent in document layouts. To address this issue, we introduce Hierarchical Structure Learning using HST generated by RIDGE to strengthen models' comprehension of hierarchical relationships within document text and layout.

Hierarchical Structure Parsing (HSP) requires models to parse the entire document content into our HST format. As shown in Fig. 3(a), given a document image, models need to identify paragraph boundaries, understand the hierarchical nesting relationships within its content, and output HST with proper tags and indentation levels. This task helps models understand both the overall structure and complex relationships of the document.

HSP with Localization is built upon the basic HSP task. We introduce spatial awareness by requiring models to parse hierarchical structures within specific regions. As shown in Fig. 3(b), models are given a region represented by bounding box coordinates (normalized to 0-999), and are required to output HST for content within that region. This task maintains the localization ability of pre-trained models during downstream training while facilitating hierarchical structure learning by reducing the task to a smaller region.

Visual Information Extraction with HSP combines HSP with VIE task. As shown in Fig. 3(c), when querying specific information (e.g., "Name:"), models first output the hierarchical structure of the relevant document section, then provide the answer after "###Answer:". Inspired by the idea of Chain-of-Thought [45], this approach encourages models to understand document context through its hierarchical structure before extracting information, potentially leading to more robust and interpretable results.

Methods	Open-set		Closed-set						
	FUNSD	**XFUND-ZH**	**CORD**	**CORD$^-$**	**EPHOIE**		**POIE**		**SROIE$^-$**
	F1 %	*F1 %*	*F1 %*	*ANLS %*	*Acc %*	*ANLS %*	*Acc %*	*ANLS %*	*ANLS%*
LayoutLLM [33] *(CVPR'24)*	-	-	-	63.10	-	-	-	-	72.12
Monkey [29] *(CVPR'24)*	34.27	28.02	54.54	69.48	22.44	30.96	30.64	43.95	64.95
DocOwl-1.5-Chat [14] *(EMNLP'24)*	50.88	10.14	64.01	63.15	4.85	8.25	51.57	61.35	61.09
LLaVA-NeXT-Mistral-7B [32]	31.18	6.02	46.58	63.81	4.31	8.02	49.88	60.05	58.59
LLaVA-NeXT-Mistral-7B+	33.41	10.33	52.32	63.93	5.39	12.07	51.46	62.13	58.58
$\Delta\uparrow$	**2.23**	**4.31**	**5.74**	**0.12**	**1.08**	**4.05**	**1.58**	**2.08**	**-0.01**
Qwen2-VL-7B [44]	59.89	62.08	82.71	80.40	76.91	86.52	93.51	96.01	97.50
Qwen2-VL-7B+	66.48	69.84	84.47	85.53	77.89	87.79	94.04	96.71	97.74
$\Delta\uparrow$	**6.59**	**7.76**	**1.76**	**5.13**	**0.98**	**1.27**	**0.53**	**0.70**	**0.24**

Table 2. Performance improvements in zero-shot VIE for MLLMs fine-tuned with RIDGE. "+" denotes that the model is extensively trained with RIDGE. $\Delta\uparrow$ represents performance gain.

4. Experiments

We present the experimental setups for RIDGE, including implementation details, evaluation setup, and results.

4.1. Implementation Details

Models. For the LLM in the content generation stage, we adapt GPT-4o-mini [38] to generate various document-related content with themes spanning business, education, government, and medical use in both English and Chinese. As for CLGM, we use LLaMA-3.1-8B [8] as our backbone and fine-tune it with LoRA [15]. The max sequence length is set as 8000 to accommodate the long layout sequence formed by hundreds of layout entities. Content generation prompts and more training setup are in the supplementary.
Datasets. To develop CLGM's expertise in form-like document layout design, we collect approximately 100K document images with OCR annotations from publicly available VDU datasets. The majority of our training data originates from RVL-CDIP [11], a document classification dataset comprising 16 classes with 25K images per class. We specifically select the "form", "specification", "resume", and "memo" classes for training due to their highly structured layouts. Additionally, we incorporate form-type VIE datasets, including FUNSD [20], XFUND [48] with 7 languages. We also utilize HUST-CELL [50] and manually select approximately 600 structurally complex document images from it. For FUNSD, XFUND, and HUST-CELL, only images from the official training sets are used. Regarding OCR annotations, we use the results provided by the datasets themselves; while RVL-CDIP does not provide OCR annotations, we use Google Vision API to obtain OCR results. Given that FUNSD and XFUND consist entirely of form documents, we treat them as high-quality data and apply 4 times up-sampling during training.

4.2. Evaluation Setup

Since RIDGE is the first work that can automatically generate relation-rich visual documents with annotations, and

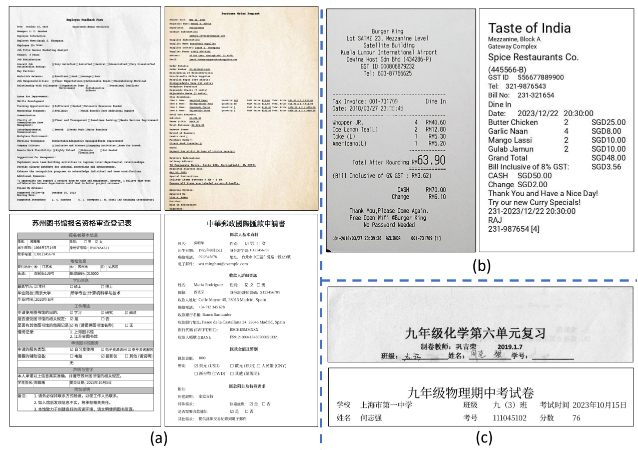

Figure 4. Example of generated documents. (a) General form-like images. (b) Right: SROIE-styled image; Left: real SROIE image. (c) Bottom: EPHOIE-styled image; Top: real EPHOIE image.

the layout evaluation criteria used in previous studies [19, 23, 40] do not consider text content in the document, it is difficult to apply those metrics (such as FID) to evaluate our work. Thus, we conduct comprehensive experiments applying RIDGE on MLLMs as well as the widely used document pre-trained model, LayoutLMv3 [16]. We assess the performance gains brought by RIDGE on various VIE benchmarks, including FUNSD, XFUND-ZH, CORD, CORD$^-$, EPHOIE, POIE, and SROIE$^-$. Benchmarks$^-$ are the cleaned version used in LayoutLLM [33], where keys with multiple values are filtered out. The synthetic documents (referred to as RIDGE) used in the experiments comprise approximately 3K English and 3K Chinese documents, yielding 444K instruction samples, including 194K classical VIE and additional Hierarchical Structure Learning samples. Examples of generated images are shown in Fig. 4(a), with more in the supplementary material.

4.3. Fine-tuning MLLMs with RIDGE

In this subsection, we evaluate the performance gains of MLLMs after being extensively fine-tuned with RIDGE.

Dataset Combinations	**FUNSD**	**CORD**	**CORD⁻**	**POIE**		**SROIE⁻**
	F1 %	*F1 %*	*ANLS %*	*Acc %*	*ANLS %*	*ANLS %*
Qwen2-VL-7B [44]	59.89	82.71	80.40	93.51	96.01	97.50
+RIDGE	65.72 (+5.83)	84.20 (+1.49)	83.71 (+3.31)	**94.45** (+0.94)	96.71 (+0.70)	97.64 (+0.14)
+RIDGE+DocVQA	66.21 (+6.32)	84.06 (+1.35)	**85.66** (+5.26)	94.08 (+0.57)	96.67 (+0.66)	**97.72** (+0.22)
+RIDGE+DocVQA+FUNSD	**67.54** (+7.65)	**84.25** (+1.54)	85.57 (+5.17)	94.11 (+0.60)	**96.72** (+0.71)	97.71 (+0.21)

Table 3. Comparison of fine-tuning MLLMs with RIDGE and real-world datasets. The values in parentheses show the performance gains compared to the baseline Qwen2-VL-7B model.

To strengthen the hierarchical structure understanding of relation-rich documents, we apply Hierarchical Structure Learning during training. Experiments are conducted on LLaVA-NeXT-Mistral-7B (referred to as LLaVA-NeXT) and Qwen2-VL-7B (referred to as Qwen2-VL). More details on training are provided in the supplementary material.

Fine-tuning with RIDGE alone. As shown in Tab. 2, RIDGE consistently enhances model performance across all evaluation benchmarks, with the only exception being a marginal decrease of 0.01% for LLaVA-NeXT on SROIE⁻. While LLaVA-NeXT initially scores behind DocOwl-1.5 [14] on some benchmarks, which is an MLLM mainly trained on VDU tasks, extensive training with RIDGE enables it to surpass DocOwl-1.5. Regarding Qwen2-VL, RIDGE successfully pushes its performance to a new state-of-the-art level compared with previous studies.

Further analyzing the results across different benchmark types, we observe that the performance gains are more substantial in open-set VIE benchmarks (exceeding more than 6%) compared to closed-set benchmarks, particularly for Qwen2-VL. This aligns with our expectations, as RIDGE is mainly designed to generate documents that emphasize open-set scenarios, which promotes better generalization. Beyond its primary strength in open-set data, training on these generalized key-value pairs also yields performance improvements on closed-set data. This is significant because closed-set data often contains inherent biases due to domain-specific key definitions that typically require extensive context to interpret correctly.

Fine-tuning with existing real-world datasets. To further validate the efficacy of RIDGE in diverse training scenarios, we conduct experiments using RIDGE independently and in combination with real-world datasets. For a comprehensive evaluation, we incorporate DocVQA, a widely used dataset containing human-annotated QA pairs for general VDU training, and FUNSD, an open-set VIE dataset with annotated form-type documents. We use RIDGE-generated English documents in this experiment and evaluate the performance gains on English benchmarks. The results of various dataset combinations—(1) RIDGE alone, (2) RIDGE + DocVQA, and (3) RIDGE + DocVQA + FUNSD—are presented in Tab. 3. Each experimental configuration independently initializes from the base Qwen2-VL-7B model.

Methods	**SROIE⁻**	**EPHOIE**	
	ANLS%	*Acc%*	*ANLS%*
Qwen2-VL-7B [44]	97.50	76.91	86.52
+RIDGE	97.74 (+0.24)	77.89 (+0.98)	87.79 (+1.27)
+RIDGE+RIDGE-DS	**98.05 (+0.55)**	**80.91 (+4.00)**	**89.82 (+3.30)**

Table 4. Performance improvements in closed-set VIE after applying domain-specific document generation. ”*RIDGE*” denotes the general open-set documents generated by RIDGE. ”*RIDGE-DS*” denotes domain-specific documents generated by RIDGE.

Training with RIDGE alone achieves substantial improvements across all benchmarks. While incorporating DocVQA and FUNSD brings additional gains in some cases, these improvements are relatively modest. The effectiveness of RIDGE is evident from its contribution: on FUNSD, RIDGE alone contributes to a 5.83% improvement, and on CORD, it yields a 1.49% improvement. Even with the addition of real-world datasets, the maximum gains only reach 7.65% on FUNSD and 1.54% on CORD. These results demonstrate that RIDGE can provide a strong foundation for VIE tasks.

4.4. Domain-Specific Document Generation

To address the challenges in closed-set VIE discussed in Sec. 4.3, we generate additional synthetic data that directly align with specific benchmark's pre-defined key sets. In this experiment, we focus on SROIE and EPHOIE. For these benchmarks, we prompt GPT-4o [38] to generate text content that mimics their respective domains: receipts and exam cover pages. More details can be found in the supplemental. Subsequently, our CLGM tackles the layout design. Examples of generated images are shown in Fig. 4(b)(c).

We conduct experiments using Qwen2-VL, initializing from an enhanced version trained with RIDGE alone (as discussed in Sec. 4.3) for better comprehension of document structure. We perform separate training for each benchmark using approximately 700 SROIE-styled documents and 800 EPHOIE-styled documents respectively. The performance gains after training on these domain-specific documents are presented in Tab. 4. The results show significant performance improvements on both benchmarks. Notably, on EPHOIE, we observe substantial improvements with accuracy increasing by 4.00% and ANLS rising by 3.30%. These improvements demonstrate the ef-

Pre-training Data	Fine-tuning Data	0-shot	1-shot	5-shot	10-shot
-	FUNSD	-	45.42	72.45	71.03
RIDGE		**62.77**	**71.07**	**73.25**	**74.79**
-	XFUND-ZH	-	52.02	76.39	81.18
RIDGE		**69.75**	**70.91**	**78.57**	**83.02**

Table 5. Few-shot learning with LayoutLMv3 for SER task. *x-shot* specifies the number of fine-tuning images.

fectiveness of our domain-specific document generation and validate the reliability of data generated by RIDGE.

4.5. Applied RIDGE on LayoutLMv3

In addition to analyzing entity relations, Semantic Entity Recognition (SER), which focuses on classifying entities into semantic categories, is also a common subtask of VIE. In this experiment, we utilize LayoutLMv3$_{BASE}$ [16], which has been widely adopted for VIE tasks, with few-shot learning to verify the quality of the semantic category annotations in RIDGE and its effectiveness on SER task. We discuss whether introducing RIDGE for pre-training prior to few-shot fine-tuning on real images can boost performance on real datasets. Following LayoutLMv3, a linear classifier is added to adapt the model for SER downstream task.

We evaluate on FUNSD and XFUND-ZH, as RIDGE adheres to the same semantic category labels. We conduct experiments with and without RIDGE pre-training, followed by few-shot fine-tuning on the training set and evaluation on the testing set of each dataset. As shown in Tab. 5, the use of RIDGE consistently enhances model performance across all few-shot settings. Notably, RIDGE demonstrates effective zero-shot adaptation, suggesting the reliability of its semantic category labels for real-world document simulation.

4.6. Ablation Study

To assess the effectiveness of Hierarchical Structure Learning, we conduct an ablation study comparing model performance between training with classical VIE tasks alone and training with both VIE and Hierarchical Structure Learning. To ensure a fair comparison, we train both settings for the same number of iterations (800 steps) to evaluate the efficiency and effectiveness of performance gains. We conduct this study on FUNSD, XFUND-ZH, and CORD, with F1 scores reported in Tab. 6. The first row (*r1*) presents the baseline performance of Qwen2-VL-7B without any additional training. While the majority of performance gains come from VIE data generated by RIDGE (as shown by the larger improvements from *r1* to *r2* than those from *r2* to *r3*), incorporating Hierarchical Structure Learning further improves performance, yielding additional gains of 1.61% on FUNSD, 2.52% on XFUND-ZH, and 0.70% on CORD.

4.7. Interpretability

Since we apply the VIE with HSP task during training, the fine-tuned Qwen2-VL acquires the ability to construct related information from documents in a hierarchical format before responding with the final answer. As shown in Fig. 5(a), before extracting the "subtotal price," the model first outputs each item's quantity and price. Also, in Fig. 5(b), before extracting the "total count of quantit," the model lists each item's information first. The process of hierarchical structure parsing not only helps the model think clearly but also provides users with detailed information related to the target query in a highly structured format.

	Hierarchical Structure Learning	FUNSD	XFUND-ZH	CORD
r1	Qwen2-VL-7B [44]	59.89	62.08	82.71
r2		64.87	67.32	83.77
r3	✓	**66.48**	**69.84**	**84.47**

Table 6. Ablation study on Hierarchical Structure Learning. *r1* represents the baseline model without additional training, *r2* is trained solely on VIE, and *r3* adopts both VIE and Hierarchical Structure Learning.

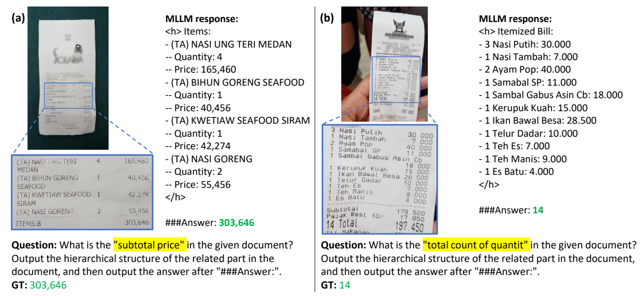

Figure 5. Interpretability brought by VIE with Hierarchical Structure Parsing.

5. Limitation

In this work, we train RIDGE mainly on form-like images and demonstrate its adaptability. Nevertheless, the generalization of RIDGE to significantly distinct document types may require additional training to achieve optimal performance, which is also our future work.

6. Conclusion

We propose RIDGE, a novel relation-rich visual document generator that leverages the powerful content-layout understanding of LLMs to generate meaningful synthetic document images. RIDGE is the first work that can automatically generate annotated relation-rich visual documents, which facilitates the training of visual document analysis models. Additionally, we introduce Hierarchical Structure Learning to enhance models' comprehension of the inherent hierarchical structure in document layouts. Through experiments on various VIE benchmarks, RIDGE can boost the performance of existing document understanding models and increase interpretability for practical applications.

Acknowledgements

This work was supported in part by the National Science and Technology Council, Taiwan under Grants NSTC 111-2634-F-002-023, 112-2221-E-002-182-MY3 & 112-2221-E-002-132-MY3, NTU under grants 113L900902 & 114L893104, and E.SUN Bank. We thank to National Center for High-performance Computing (NCHC) of National Applied Research Laboratories (NARLabs) in Taiwan for providing computational and storage resources.

References

[1] Diego Martin Arroyo, Janis Postels, and Federico Tombari. Variational transformer networks for layout generation. In *CVPR*, pages 13642–13652, 2021. 2

[2] Sanket Biswas, Pau Riba, Josep Lladós, and Umapada Pal. Docsynth: A layout guided approach for controllable document image synthesis. In *International Conference on Document Analysis and Recognition (ICDAR)*, 2021. 2

[3] Manuel Carbonell, Pau Riba, Mauricio Villegas, Alicia Fornés, and Josep Lladós. Named entity recognition and relation extraction with graph neural networks in semi structured documents. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9622–9627, 2021. 3

[4] Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser-2: Unleashing thenbsp;power ofnbsp;language models fornbsp;text rendering. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part V*, page 386–402, Berlin, Heidelberg, 2024. Springer-Verlag. 2

[5] 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. *Science China Information Sciences*, 67(12):220101, 2024. 3

[6] Jaemin Cho, Abhay Zala, and Mohit Bansal. Visual programming for step-by-step text-to-image generation and evaluation. *NeurIPS*, 36, 2024. 2

[7] Brian Davis, Bryan Morse, Brian Price, Chris Tensmeyer, Curtis Wigington, and Vlad Morariu. End-to-end document recognition and understanding with dessurt. In *Computer Vision – ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, page 280–296, Berlin, Heidelberg, 2022. Springer-Verlag. 3

[8] 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. 6, 1

[9] Weixi Feng, Wanrong Zhu, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Xuehai He, S Basu, Xin Eric Wang, and William Yang Wang. LayoutGPT: Compositional visual planning and generation with large language models. In *NeurIPS*, 2023. 2, 3, 4

[10] Andrea Gemelli, Sanket Biswas, Enrico Civitelli, Josep Lladós, and Simone Marinai. Doc2graph: A task agnostic document understanding framework based onnbsp;graph neural networks. In *Computer Vision – ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, page 329–344, Berlin, Heidelberg, 2022. Springer-Verlag. 3

[11] Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 991–995. IEEE, 2015. 6

[12] Liu He, Yijuan Lu, John Corring, Dinei Florencio, and Cha Zhang. Diffusion-based document layout generation. In *Document Analysis and Recognition - ICDAR 2023*, pages 361–378. Springer Nature Switzerland, 2023. 2

[13] Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents. In *AAAI*, pages 10767–10775, 2022. 1, 3

[14] Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. mPLUG-DocOwl 1.5: Unified structure learning for OCR-free document understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3096–3120, Miami, Florida, USA, 2024. Association for Computational Linguistics. 5, 6, 7, 3

[15] 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. In *ICLR*, 2022. 6, 1

[16] 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 MM*, pages 4083–4091, 2022. 1, 6, 8, 3

[17] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520. IEEE, 2019. 2, 1

[18] Mude Hui, Zhizheng Zhang, Xiaoyi Zhang, Wenxuan Xie, Yuwang Wang, and Yan Lu. Unifying layout generation with a decoupled diffusion model. In *CVPR*, pages 1942–1951, 2023. 2

[19] Naoto Inoue, Kotaro Kikuchi, Edgar Simo-Serra, Mayu Otani, and Kota Yamaguchi. LayoutDM: Discrete Diffusion Model for Controllable Layout Generation. In *CVPR*, pages 10167–10176, 2023. 1, 2, 3, 6

[20] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. IEEE, 2019. 2, 6

[21] Peidong Jia, Chenxuan Li, Yuhui Yuan, Zeyu Liu, Yichao Shen, Bohan Chen, Xingru Chen, Yinglin Zheng, Dong Chen, Ji Li, Xiaodong Xie, Shanghang Zhang, and Baining

Guo. Cole: A hierarchical generation framework for multi-layered and editable graphic design, 2024. 2

[22] Zhaoyun Jiang, Shizhao Sun, Jihua Zhu, Jian-Guang Lou, and Dongmei Zhang. Coarse-to-fine generative modeling for graphic layouts. In *AAAI*, pages 1096–1103, 2022. 2

[23] Zhaoyun Jiang, Jiaqi Guo, Shizhao Sun, Huayu Deng, Zhongkai Wu, Vuksan Mijovic, Zijiang James Yang, Jian-Guang Lou, and Dongmei Zhang. Layoutformer++: Conditional graphic layout generation via constraint serialization and decoding space restriction. In *CVPR*, pages 18403–18412, 2023. 1, 2, 6

[24] Akash Abdu Jyothi, Thibaut Durand, Jiawei He, Leonid Sigal, and Greg Mori. Layoutvae: Stochastic scene layout generation from a label set. In *ICCV*, pages 9895–9904, 2019. 2

[25] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *ECCV*, 2022. 2, 3

[26] Jianfeng Kuang, Wei Hua, Dingkang Liang, Mingkun Yang, Deqiang Jiang, Bo Ren, and Xiang Bai. Visual information extraction in the wild: practical dataset and end-to-end solution. In *International Conference on Document Analysis and Recognition*, pages 36–53. Springer, 2023. 2

[27] Fengheng Li, An Liu, Wei Feng, Honghe Zhu, Yaoyu Li, Zheng Zhang, Jingjing Lv, Xin Zhu, Junjie Shen, Zhangang Lin, and Jingping Shao. Relation-aware diffusion model for controllable poster layout generation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, page 1249–1258, 2023. 2

[28] Jianan Li, Tingfa Xu, Jianming Zhang, Aaron Hertzmann, and Jimei Yang. LayoutGAN: Generating graphic layouts with wireframe discriminator. In *ICLR*, 2019. 1, 2

[29] Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. In *CVPR*, pages 26763–26773, 2024. 6

[30] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. LLM-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Transactions on Machine Learning Research*, 2024. Featured Certification. 2

[31] Jiawei Lin, Jiaqi Guo, Shizhao Sun, Zijiang James Yang, Jian-Guang Lou, and Dongmei Zhang. Layoutprompter: Awaken the design ability of large language models. In *NeurIPS*, 2023. 2, 3, 4

[32] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 5, 6, 1, 3

[33] 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 *CVPR*, pages 15630–15640, 2024. 5, 6, 1, 3

[34] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *WACV*, 2021. 1

[35] 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. 2

[36] Akshay Gadi Patil, Omri Ben-Eliezer, Or Perel, and Hadar Averbuch-Elor. Read: Recursive autoencoders for document layout generation. In *CVPRW*, pages 544–545, 2020. 2

[37] Qiming Peng, Yinxu Pan, Wenjin Wang, Bin Luo, Zhenyu Zhang, Zhengjie Huang, Yuhui Cao, Weichong Yin, Yongfeng Chen, Yin Zhang, Shikun Feng, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. ERNIE-layout: Layout knowledge enhanced pre-training for visually-rich document understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3744–3756, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics. 1, 3

[38] Partha Pratim Ray. Chatgpt: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*, 3:121–154, 2023. 1, 6, 7

[39] Štěpán Šimsa, Milan Šulc, Michal Uřičář, Yash Patel, Ahmed Hamdi, Matěj Kocián, Matyáš Skalickỳ, Jiří Matas, Antoine Doucet, Mickaël Coustaty, et al. Docile benchmark for document information localization and extraction. In *International Conference on Document Analysis and Recognition*, pages 147–166. Springer, 2023. 2

[40] Zecheng Tang, Chenfei Wu, Juntao Li, and Nan Duan. LayoutNUWA: Revealing the hidden layout expertise of large language models. In *ICLR*, 2024. 1, 2, 3, 4, 6

[41] Yi Tu, Ya Guo, Huan Chen, and Jinyang Tang. LayoutMask: Enhance text-layout interaction in multi-modal pre-training for document understanding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15200–15212, Toronto, Canada, 2023. Association for Computational Linguistics. 1, 3

[42] 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. 3

[43] Jiapeng Wang, Chongyu Liu, Lianwen Jin, Guozhi Tang, Jiaxin Zhang, Shuaitao Zhang, Qianying Wang, Yaqiang Wu, and Mingxiang Cai. Towards robust visual information extraction in real world: new dataset and novel solution. In *AAAI*, pages 2738–2745, 2021. 2, 1

[44] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 1, 5, 6, 7, 8, 3

[45] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny

Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, pages 24824–24837. Curran Associates, Inc., 2022. 5

[46] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1192–1200, 2020. 1, 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 *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2579–2591, Online, 2021. Association for Computational Linguistics. 1, 3

[48] Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. XFUND: A benchmark dataset for multilingual visually rich form understanding. In *Findings of the Association for Computational Linguistics: ACL 2022*, Dublin, Ireland, 2022. Association for Computational Linguistics. 2, 6

[49] Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, Qin Jin, Liang He, Xin Lin, and Fei Huang. UReader: Universal OCR-free visually-situated language understanding with multimodal large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2841–2858, Singapore, 2023. Association for Computational Linguistics. 3

[50] Wenwen Yu, Chengquan Zhang, Haoyu Cao, Wei Hua, Bohan Li, Huang Chen, Mingyu Liu, Mingrui Chen, Jianfeng Kuang, Mengjun Cheng, et al. Icdar 2023 competition on structured text extraction from visually-rich document images. In *International Conference on Document Analysis and Recognition*, pages 536–552. Springer, 2023. 6