EasyChair Preprint

№ 12130

Embedding Layout in Text for Document Understanding Using Large Language Models

Mohammad Minouei, Mohammad Reza Soheili and Didier Stricker

February 15, 2024

Embedding Layout in Text for Document Understanding Using Large Language Models

Mohammad Minouei[1,2][0000−0001−7476−6533], Mohammad Reza Soheili[2,3][0000−0002−5974−3939], and Didier Stricker[1,2]

[1] Department of Computer Science, RPTU Kaiserslautern-Landau, Germany
[2] German Research Institute for Artificial Intelligence (DFKI), 67663 Kaiserslautern, Germany
{firstname.lastname}@dfki.de
[3] Department of Electrical and Computer Engineering, Kharazmi University, Iran
{lastname}@khu.ac.ir

Abstract. In this paper, we address the challenge of effectively utilizing Large Language Models (LLMs) for Visually Rich Document Understanding (VRDU), a key part of intelligent document processing systems. While LLMs excel in various Natural Language Processing (NLP) tasks, their application for extracting information from complex structured documents like invoices and forms is limited. This limitation arises from the difficulty in contextually understanding these documents, largely due to the lack of layout information. Our research is dedicated to unlocking the full potential of LLMs for VRDU by integrating OCR data into an HTML format, which preserves the essential spatial layout for accurate information extraction. The empirical results show a notable improvement, with a more than 20 percent increase over baseline performances. This research highlights the promising potential of LLMs in VRDU and sets the stage for further innovations in automated document processing.

Keywords: Document Understanding · Large Language Model · Information Extraction

1 Introduction

Document understanding aims at interpreting and extracting meaningful information from documents. Being an active area of research, a wide range of approaches have been studied in the literature that utilize document images, text, or a combination of both [24]. The field has evolved from initial heuristic methods [10], to the modern specialized deep neural network techniques [13]. The complexity of this task lies in the endless variations in document layouts, such as invoices, tax forms, and many more. These structured documents often feature elements such as tables and key-value pairs, needing advanced methods for information extraction.

The advent of LLMs has brought about a significant transformation in the field of natural language processing, surpassing previous state-of-the-art (SOTA)

methods [20]. Models like Chat-GPT [3] and Llama [25] have gained significant recognition for their remarkable text understanding and generation capabilities. These models are trained on large datasets, allowing them to recognize complex patterns and subtleties in natural language. With their substantial size, often comprising hundreds of billions of parameters, LLMs can handle a diverse range of tasks without the need for task specific data, also known as zero-shot ability.

LLMs are not limited to just generating text; they have the potential to transform how we understand documents. Their applications are diverse, including tasks such as extracting information, conducting semantic searches, and summarizing documents [20]. This understanding can be further refined with a limited number of examples to adapt to specific document related tasks. The few-shot learning ability of LLMs is especially useful in processing documents from fields where labeled data is rare or costly to gather [5].

This has inspired researchers to explore new ways to effectively leverage LLMs for document understanding tasks. Despite their impressive capabilities, using LLMs to interpret visually dense, structured documents is still understudied. A major difficulty is the absence of layout information in the text, which is vital for effective information extraction. To make the most of LLMs, it is crucial to prepare the input data carefully. How well these models perform largely depends on how the input prompts are structured. If the preparation is not done properly, it can result in responses that are either irrelevant or incorrect [12].

Recently, researchers introduced the Visually-Rich Document Understanding (VRDU) benchmark dataset [26] to evaluate how well models perform in this area. VRDU includes two kinds of documents of *purchases* and *registration* forms. The VRDU benchmark provides challenging tasks to assess the capabilities of models in various scenarios, including test sets with mixed templates or templates that have not been seen before. It also evaluates the performance of models in situations with limited data (few-shot settings) and their ability to identify nested or repeated entities. This benchmark offers an excellent opportunity to showcase the potential of language models in understanding documents [26].

The significance of the VRDU benchmark is especially apparent when evaluating advanced document AI systems like LayoutLM [28] and FormNet [15]. These models demonstrate substantial progress on document understanding, yet struggle with VRDU's complex dataset. The LayoutLM series, represents a major step forward in understanding document images. They combine language models' capabilities with spatial and visual contexts. The LayoutLM model improved upon BERT by adding 2-D positional and image embedding for tokens. It showed skill in tasks like extracting information and classifying documents. LayoutLMv2 [27] went further, improving how it integrates visual data during pre-training and using a multi-modal transformer architecture. This development included a spatial-aware self-attention mechanism, enhanced through tasks like masked visual-language modeling and text-image matching, improving its understanding of visually complex documents. LayoutLMv3 [13], an improvement over LayoutLMv2, adopts patch embedding along with Vision Transformers [6]

instead of using a CNN backbone. It simplifies the structure and pre-training process, focusing on masked language modeling (MLM), masked image modeling (MIM), and word-patch alignment (WPA), boosting its document understanding capabilities.

FormNet [15], another innovative system, combines sequence and convolutional approaches for impressive results. It introduces rich attention and super-tokens. Rich attention calculates attention scores by considering the spatial relationships between tokens, capturing the document's structural details. Super-tokens are created for each word in a form, incorporating embeddings from neighboring tokens using graph convolutions.

Incorporating layout with text in the network has been studied in various works [18,17,16,7,8]. However, Donut [14] proposed an end-to-end encoder-decoder model that leverages transformer architecture to directly map raw input images to desired outputs, bypassing the need for OCR.

In [22], the authors introduce a novel method called LMDX for extracting information from semi-structured documents using LLMs. The LMDX approach addresses the challenges of information extraction by incorporating text position encoding and a grounding mechanism along with their LLM. It uses a five-stage process: OCR, chunking, generating prompts, LLM inference, and decoding. This pipeline is designed to efficiently identify and locate entities in the documents.

Facing the challenge of using LLMs for understanding documents, we propose using a machine-friendly data representation. HTML stands out for its adaptability and proves to be highly effective in this context. In [9], authors extensively analyze the application of LLMs in tasks related to understanding HTML. The study highlights that, when appropriately fine-tuned, LLMs demonstrate outstanding performance on benchmark tests assessing HTML comprehension. In our specific use-case, the conversion of raw text into HTML allows us to retain both the textual content and spatial layout of the documents, which plays a key role in achieving accurate and precise analysis. This approach is particularly useful for documents with standard layouts, such as forms with key-value pairs.

In summary, our research aims to bridge the gap between the capabilities of LLMs and the practical requirements of visually-rich document understanding. We make the following contributions:

- We present a new method for transforming OCR document outputs into structured HTML representations that preserve spatial relationships and layout context.
- Through extensive experiments, we show that instruction-based prompting, which includes HTML representations, improves LLMs' capacity to comprehend complex visual layouts.

Overall, our research contributes to the ongoing efforts to leverage LLMs in real-world VRDU applications.

2 Approach

The field of large language models is rapidly evolving, with new models being introduced regularly. However, for our specific task, we require a LLM that can handle large contexts, understand HTML and JSON, and follow instructions precisely. We have identified a variant of Llama 2 [25], known as CodeLlama [23], which meets these requirements. CodeLlama is fine-tuned with massive datasets containing code, markup languages, and natural language text related to coding. It is specifically designed to handle extensive input contexts, allowing it to process longer sequences of up to 16,384 tokens. Moreover, CodeLlama excels at following detailed instructions, making it ideal for tasks related to programming and data manipulation.

Our method uses tailored instruction prompts to fine-tune and test the LLM for understanding documents. We begin by preparing the data, converting the document's OCR output into an HTML representation. This HTML format serves as the input for the LLM along with an instruction prompt that contains task-specific details and the desired output. The LLM is then fine-tuned using these instruction prompts. The following sections will explain these steps in more detail.

2.1 HTML Representation

HTML is an ideal format for representing complex layout structures of documents. In the past, OCR engines such as Tesseract [2] have offered a specialized HTML representation in hOCR format [4]. Unlike plain text, HTML elements can capture how textual components spatially relate to one another within a rich formatting structure, which is particularly useful in scenarios that involve key-value inputs and require maintaining the relationships between words. As studied in [9], HTML serves as an interpretable structured medium for LLM.

The conversion process from a document's OCR output to HTML is outlined in Algorithm 1. We use bounding box coordinates to arrange the text elements into a `<table>` layout, with `<tr>` rows and `<td>` cells, based on their relative positions. The algorithm then sorts and organizes these elements to create a coherent HTML structure that retains original spatial positioning relationships. Figure 1 shows a sample document and corresponding HTML encoding generated by this process.

2.2 Prompt Generation

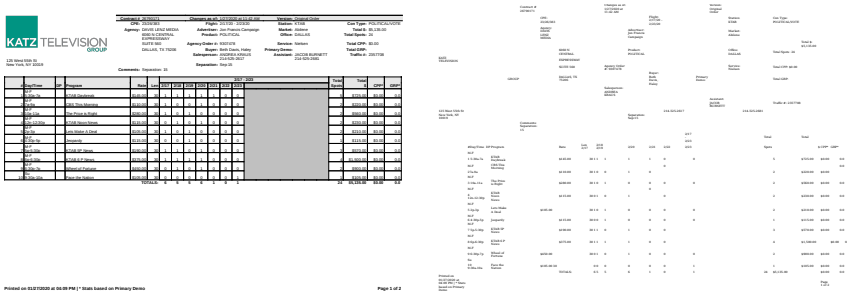
LLMs can be effectively directed to perform specific tasks by providing an instructive prompt that clearly defines the desired behavior. The Llama LLM uses two types of prompts: a system prompt and an instruction prompt. The system prompt sets the general tone and expectations for the interaction and is prepended to the prompt. The instruction prompt, on the other hand, clearly specifies the expected response.

We have designed the system prompt as follows:

Algorithm 1: Convert OCR results to HTML Table

Data: List of texts and bounding boxes
Result: HTML table

```
foreach bounding box do
    Calculate row and column based on bounding box coordinates;
    Append (row, column, text) to data list;
Sort data_list by row and then by column;
Initialize table_html;
foreach (row, column, text) in sorted_data do
    if row ≠ current_row then
        if current_row ≠ 0 then
            Add "</tr>" to table_html;
        Add "<tr>" to table_html;
        Update current_row and reset last_col;
    Calculate colspan based on column and last_col;
    Add <td></td> with text content to table_html;
    Update last_col with current column;
Add "</tr> </table>" to table_html;
```



(a) (b)

Fig. 1. Comparison between the original document (a) and its corresponding HTML representation (b). Sample from VRDU benchmark.

"Below is an instruction that describes a task, paired with an input that provides further context. Your response is a JSON object that appropriately completes the request. The JSON must be between [JSON] and [/JSON] tags."

This prompt sets the format and expectations for the model's response.

Following this, the instruction prompt offers specific directives for the task, guiding the LLM to precisely extract and organize the required information:

"Given the following HTML table, extract key details and organize them into a single JSON object. Please provide values for the fields including 'advertiser,' 'property,' 'agency,' 'tv_address,' 'contract_num,' 'product,' 'gross_amount,' 'flight_from,' 'flight_to,' and 'line_item' is an array (with 'channel,' 'program_desc,' 'program_end_date,' 'program_start_date,' and 'sub_amount'). Ensure that the extracted information accurately reflects the content of the html. Output must be JSON."

Figure 2 illustrates the process of forming the prompt for our application. Initially, an HTML table is generated from the OCR data, capturing the layout and textual content of the document. Subsequently, a JSON object is derived from the ground truth, which includes only the key values relevant to the current page. The combination of these elements results in the creation of a well-formatted prompt.

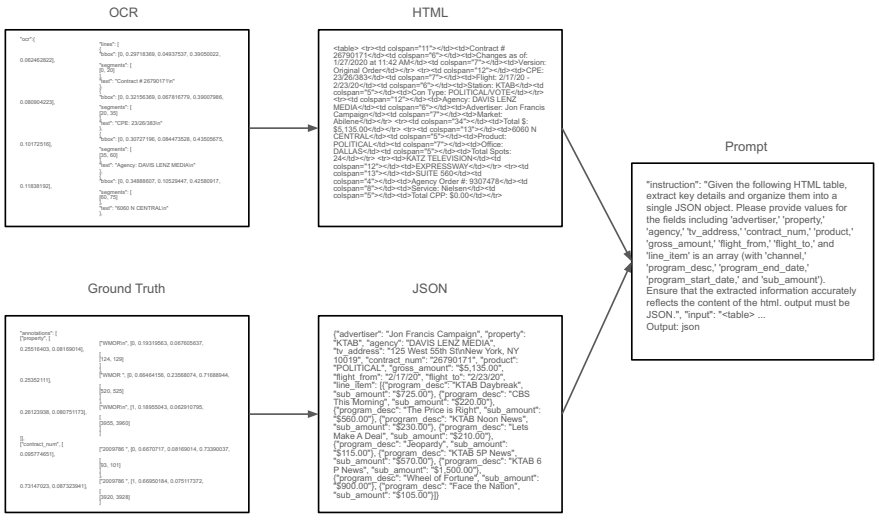


Fig. 2. An HTML table reflecting the document's layout and content is combined with a JSON object containing key-value pairs from the ground truth annotations to construct the final prompt.

2.3 Implementation Details

For fine-tuning we used parameter-efficient fine-tuning (PEFT) [19] method with Low-Rank Adaptation (LoRA) [11] configuration. LoRA provides an efficient method for fine-tuning LLM by introducing smaller low-rank matrices to each layer instead of modifying the original weight matrices. For our implementation, we chose an alpha value of 64, which scales the low-rank updates applied to the model weights. The rank of the low-rank matrices was set to 16, determining the size of the trainable matrices. Additionally, we integrated a dropout rate of 0.05 in our LoRA layers. Training was conducted for 1K iterations on an A100 GPU. We used a baseline learning rate of 1e-4 with a cosine learning rate scheduler. The scheduler decreases the rate following a cosine curve over the training span. During inference for each sample, we combine the predicted outcomes of all pages into a single JSON object. To ensure accurate evaluation, we sort the key values in both the JSON object and the ground truth. Our code is available at: https://github.com/minouei-kl/llm4vrdu.

3 Experiments and Results

In this section, we review the experiments and their results. First, we evaluate the Ad-buy dataset from VRDU under various settings defined by the benchmark. Next, we focus on a specific subset of the Ad-buy dataset, comprising 100 training samples and a test set with unseen templates. This subset serves as challenging testing ground to conduct additional experiments. Using this subset, we assess the model's performance using a different input encoding, evaluate how the LLM performs without any training, and compare the results with another LLM. Lastly, we test the Consolidated Receipt Dataset (CORD) [21] to determine the versatility of our encoding approach.

3.1 Datasets

The VRDU benchmark is composed of two distinct datasets: Ad-buy Forms and Registration Forms. The Ad-buy Forms dataset, which is the more challenging of the two, consists of 641 documents. These documents are mainly invoices or receipts related to political advertisements, featuring details like product names, flight dates, and total prices, which are typically found in invoices. These documents contain complex elements such as tables, multi-column layouts, and key-value pairs. They contain diverse data types, including prices, dates, addresses, and nested entities. The dataset provides high quality OCR extraction results for the text and their corresponding positions in the documents.

The benchmark includes two tasks: Mixed Template Learning (MTL) and Unseen Template Learning (UTL). MTL evaluate the models' ability to handle various templates by incorporating multiple templates across training and testing sets. UTL evaluates the models' capacity to adapt to templates not seen during training. Each task in the VRDU dataset consists of 300 documents in the testing

set, with four different training sets of 10, 50, 100, and 200 samples, respectively. This structure allows for assessing models on their efficiency with data and their performance with limited training data. Additionally, the authors implement a type-aware matching algorithm to accurately assess performance, taking into account different data types and formats.

Additionally, the CORD [21] contains a thousand of Indonesian receipt images receipts. It comes with rich annotations for OCR and multi-level semantic labels for each word. The dataset is divided into training (800 receipts), validation (100 receipts), and test sets (100 receipts).

3.2 Evaluation on VRDU benchmark

Table 1 compares our proposed model with others, including LMDX, FormNet, and different versions of LayoutLM, evaluated on the Ad-buy dataset. It shows the performance of these models with varying data sizes and whether the templates in training/testing was mixed or unseen. The performance metrics include Micro-F1 and Line-Item F1 scores as defined in [26].

Our model shows significant improvement over basic methods such as FormNet and the LayoutLM family in all settings. As the size of the training set increases, there is a consistent improvement in performance. The extraction of line items, which contain nested or itemized information, is particularly challenging because the evaluation is strict, even a single missing item in a group is marked completely as incorrect.

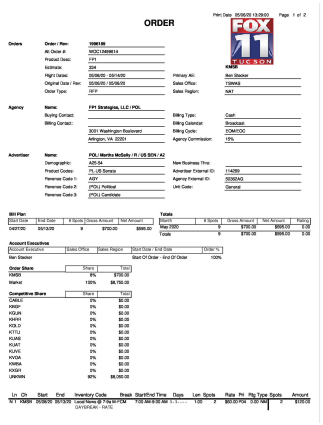
Although the proposed method performs well, it has not reached the top performance achieved by LMDX due to several factors. First, LMDX has a larger architecture with greater processing capabilities. Additionally, LMDX benefits from pre-training on a private dataset, enhancing its performance. LMDX also utilizes multiple inferencing techniques, leading to higher accuracy at a higher computational cost. Lastly, LMDX undergoes more training with 4,000 iterations compared to our 1,000 iterations. These factors, considering the computational cost and limitations in our experiments, explain the superior performance of the LMDX model in this context.

Figure 3 shows a sample document, ground truth, and model predictions. While most details are accurately extracted, there are instances where parts of the program description are missed. Such mistakes lead to a decline in the performance of the line-item.

Table 2 shows the detailed performance of our model on the Ad-buy dataset for different fields, under different template sizes (10, 50, 100, 200). For both *mixed* and *unseen* templates, as the size increases, there is an improvement in F1 scores across most fields. Some fields such as ‘gross amount‘, ‘product,‘ ‘agency‘, and ‘advertiser‘ consistently show higher F1 scores across both template types and all sizes, indicating that the model is particularly effective in these areas. Conversely, fields like ‘tv address‘, ‘line item‘, have lower F1 scores, especially in smaller template sizes, which means the model struggles more with these fields.

Table 1. Overall performance on Ad-buy dataset across various train sizes and template setting in train/test (mixed, unseen).

Size	Model	Mixed Template		Unseen
		Micro-F1	Line Item F1	Micro-F1
10	FormNet	20.47	5.72	20.28
	LayoutLM	20.20	6.95	19.92
	LayoutLMv2	25.36	9.96	25.17
	LayoutLMv3	10.16	5.92	10.01
	LMDX PaLM 2-S	54.35	39.35	54.82
	Proposed	38.06	19.66	37.76
50	FormNet	40.68	19.06	39.52
	LayoutLM	39.76	19.50	38.42
	LayoutLMv2	42.23	20.98	41.59
	LayoutLMv3	39.49	19.53	38.43
	LMDX PaLM 2-S	75.08	65.42	75.70
	Proposed	58.16	42.72	56.87
100	FormNet	40.38	18.80	39.88
	LayoutLM	42.38	21.26	41.46
	LayoutLMv2	44.97	23.52	44.35
	LayoutLMv3	42.63	22.08	41.54
	LMDX PaLM 2-S	78.05	69.77	75.99
	Proposed	65.9	52.51	63.71
200	FormNet	43.23	21.86	42.87
	LayoutLM	44.66	23.90	44.18
	LayoutLMv2	46.54	25.46	46.31
	LayoutLMv3	45.16	24.51	44.43
	LMDX PaLM 2-S	79.82	72.09	78.42
	Proposed	74.74	64.24	71.82



(a)


```
{
  "advertiser": "POL/ Martha McSally / R / US SEN / AZ",
  "property": "KMSB",
  "agency": "FP1 Strategies, LLC/POL",
  "tv_address": null,
  "contract_num": "1996189",
  "product": "FP1",
  "gross_amount": "$700.00",
  "flight_from": "05/08/20",
  "flight_to": "05/14/20",
  "line_item": [
  {
  "channel": "KMSB",
  "program_start_date": "05/08/20",
  "program_end_date": "05/13/20",
  "program_desc": "Local News @ 7-9a M-FCM DAYBREAK - RATE",
  "sub_amount": "$120.00"
  }
  ]
}
```


(b)


```
{
  "advertiser": "POL/ Martha McSally / R / US SEN / AZ",
  "property": "KMSB",
  "agency": "FP1 Strategies, LLC/POL",
  "tv_address": null,
  "contract_num": "1996189",
  "product": "FP1",
  "gross_amount": "$700.00",
  "flight_from": "05/08/20",
  "flight_to": "05/14/20",
  "line_item": [
  {
  "sub_amount": "$120.00",
  "channel": "KMSB",
  "program_start_date": "05/08/20",
  "program_end_date": "05/13/20",
  "program_desc": "Local News @ 7-9a M-FCM"
  }
  ]
}
```


(c)

Fig. 3. (a) Document image. (b) Ground truth. (c) Predicted sequence.

Table 2. F1-Scores per field on the Ad-Buy dataset across various train sizes and template setting in train/test (mixed, unseen).

Template	Size	Advertiser	Agency	Contract Num	Flight From	Flight To	Gross Amount	Product	TV Address	Property	Line Item	Macro	Unrepeated entities	Micro
Mixed	10	82.19	76.18	76.75	71.09	72.55	88.57	84.64	44.77	67.85	19.66	68.43	74.21	38.06
Mixed	50	94.32	85.01	93.18	88.81	86.76	94.92	90.15	75.96	82.84	42.72	83.47	88.46	58.16
Mixed	100	94.14	88.05	95.17	91.41	92.08	96.63	93.2	79.17	86.98	52.51	86.94	91.15	65.9
Mixed	200	97.58	93.16	96.69	94.43	94.66	97.21	95.13	84.77	93.18	64.24	91.11	94.32	74.74
Unseen	10	78.48	71.95	77.64	73.48	73.55	90.85	83	42.77	68.33	19.33	67.94	73.9	37.76
Unseen	50	93.86	87.53	93.73	88.53	87.19	94.15	92.37	76.6	83.69	40.66	83.83	88.94	56.87
Unseen	100	94.24	91.81	93.78	89.29	90.42	96.02	93.32	78.99	86.58	49.26	86.37	90.63	63.71
Unseen	200	96.21	95.16	96.56	90.49	91.15	95.94	94.62	85.53	93.44	60.14	89.92	93.32	71.82

3.3 Evaluation with Coordinate Embedding

As presented in [22], one encoding approach is to directly embed the normalized x—y coordinate pair of each word into the text input. As the authors state, this spatial context helps language models infer document layout relationships. For comparison, we train the LLM with this "coordinate-in-text" representation on a subset of dataset. As table 3 shows, our model generally outperforms the "coordinate" model in most fields, as indicated by higher F1 scores.

Table 3. Evaluation coordinate in text on unseen template 100 subset (F1-Scores).

Model	Advertiser	Agency	Contract Num	Flight From	Flight To	Gross Amount	Product	TV Address	Property	Line Item	Macro	Unrepeated entities	Micro
Coordinate	91.05	85.65	**96.04**	87.95	**90.54**	**96.51**	89.53	75	85.28	45.4	84.29	89.05	60.52
Proposed	**94.24**	**91.81**	93.78	**89.29**	90.42	96.02	**93.32**	**78.99**	**86.58**	**49.26**	**86.37**	**90.63**	**63.71**

3.4 Zero-Shot Evaluation

In this experiment, we pass the input prompt to our LLM to evaluate its performance without tuning. Table 4 compares the proposed fine-tuned model against this zero-shot baseline on the subset of VRDU with unseen templates. We observe that certain information, such as advertiser and product names, can be extracted even without fine-tuning. However, fine-tuning provides substantial gains, more than doubling scores across all categories by tailoring the model to the specific domain.

Table 4. Evaluation zero-shot on unseen template 100 subset (F1-Scores).

Model	Advertiser	Agency	Contract Num	Flight From	Flight To	Gross Amount	Product	TV Address	Property	Line Item	Macro	Unrepeated entities	Micro
CodeLlama	72.5	44.03	46.61	49.2	57.75	50.39	84.32	13.23	42.81	2.48	46.33	51.66	21.99
Proposed	**94.24**	**91.81**	**93.78**	**89.29**	**90.42**	**96.02**	**93.32**	**78.99**	**86.58**	**49.26**	**86.37**	**90.63**	**63.71**

3.5 Evaluation DeciLM-7B

To showcase the effectiveness of our encoding approach in combination with another LLM, we conducted a comparison with DeciLM-7B [1], a recently introduced instruction-following LLM that can handle long input context up to 8k. To ensure a fair comparison, we fine-tuned DeciLM-7B on Ad-buy dataset using the same steps as our proposed model. Table 5 presents the comparison between our proposed model and DeciLM-7B. The results show that both models perform similarly, but in our specific application, CodeLlama generally outperforms DeciLM-7B.

Table 5. Evaluation DeciLM-7B on unseen template 100 subset (F1-Scores).

Model	Advertiser	Agency	Contract Num	Flight From	Flight To	Gross Amount	Product	TV Address	Property	Line Item	Macro	Unrepeated entities	Micro
DeciLM-7B	90.79	86.13	**95.61**	85.56	**90.8**	**96.04**	90.35	78.3	82.83	46.35	84.28	88.85	61.11
Proposed	**94.24**	**91.81**	93.78	**89.29**	90.42	96.02	**93.32**	**78.99**	**86.58**	**49.26**	**86.37**	**90.63**	**63.71**

3.6 Evaluation on CORD Dataset

We expanded our evaluation to include the CORD receipt dataset in two different settings: using only the first 50 samples to assess the model's few-shot learning capabilities, and using the complete dataset of 800 samples, in line with [22]. We followed the same procedure for prompt creation, training, and testing as applied to the VRDU dataset. Table 6 compares the n-TED accuracy [14] of various models on the CORD dataset, as reported in [22]. The results indicate that our model performs competitively in both training scenarios. With 50 samples, it achieves a higher n-TED accuracy compared to Donut and LayoutLMv3LARGE, but lower than LMDXPaLM 2-S. With 800 samples, the model's accuracy increases and remains higher than Donut.

Table 6. Evaluation on CORD Dataset.

Size	Model	n-TED accuracy
50	Donut	75.44
	LayoutLMv3LARGE	87.29
	LMDX $_{\text{PaLM 2-S}}$	93.80
	proposed	89.9
800	Donut	90.23
	LayoutLMv3LARGE	96.21
	LMDX $_{\text{PaLM 2-S}}$	96.3
	proposed	91.4

4 Conclusion

In conclusion, we introduced a new approach for leveraging LLMs to extract information from documents with complex layouts. Our approach, which converts OCR outputs into HTML formats, effectively preserves the spatial layout and textual content, allowing LLMs to accurately extract information into a structured JSON format. Our experiments on the VRDU benchmark show significant improvement compared to baseline models and are comparable to SOTA results within computational limits. We have verified the effectiveness and flexibility of our method through testing on different inputs and models. Our findings highlight the importance of input formatting and the choice of LLM in the performance of information extraction. Future efforts can focus on improving text encoding and employing grounding techniques.

References

1. Deciai research team. 2024. decilm-7b-instruct, `https://huggingface.co/Deci/DeciLM-7B-instruct`
2. Tesseract open source ocr engine, `https://github.com/tesseract-ocr/tesseract`
3. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
4. Breuel, T.M.: The hocr microformat for ocr workflow and results. In: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). vol. 2, pp. 1063–1067. IEEE (2007)
5. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
7. Garncarek, Ł., Powalski, R., Stanisławek, T., Topolski, B., Halama, P., Turski, M., Graliński, F.: Lambert: Layout-aware language modeling for information ex-

traction. In: International Conference on Document Analysis and Recognition. pp. 532–547. Springer (2021)
8. Gu, Z., Meng, C., Wang, K., Lan, J., Wang, W., Gu, M., Zhang, L.: Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4583–4592 (2022)
9. Gur, I., Nachum, O., Miao, Y., Safdari, M., Huang, A., Chowdhery, A., Narang, S., Fiedel, N., Faust, A.: Understanding html with large language models. arXiv preprint arXiv:2210.03945 (2022)
10. Ha, J., Haralick, R., Phillips, I.: Recursive x-y cut using bounding boxes of connected components. In: Proceedings of 3rd International Conference on Document Analysis and Recognition. vol. 2, pp. 952–955 vol.2 (1995). https://doi.org/10.1109/ICDAR.1995.602059
11. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)
12. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., et al.: A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. arXiv preprint arXiv:2311.05232 (2023)
13. Huang, Y., Lv, T., Cui, L., Lu, Y., Wei, F.: Layoutlmv3: Pre-training for document ai with unified text and image masking. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 4083–4091 (2022)
14. Kim, G., Hong, T., Yim, M., Nam, J., Park, J., Yim, J., Hwang, W., Yun, S., Han, D., Park, S.: Ocr-free document understanding transformer. In: European Conference on Computer Vision. pp. 498–517. Springer (2022)
15. Lee, C.Y., Li, C.L., Dozat, T., Perot, V., Su, G., Hua, N., Ainslie, J., Wang, R., Fujii, Y., Pfister, T.: Formnet: Structural encoding beyond sequential modeling in form document information extraction. arXiv preprint arXiv:2203.08411 (2022)
16. Li, Q., Li, Z., Cai, X., Du, B., Zhao, H.: Enhancing visually-rich document understanding via layout structure modeling. In: Proceedings of the 31st ACM International Conference on Multimedia. pp. 4513–4523 (2023)
17. Li, Y., Qian, Y., Yu, Y., Qin, X., Zhang, C., Liu, Y., Yao, K., Han, J., Liu, J., Ding, E.: Structext: Structured text understanding with multi-modal transformers. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 1912–1920 (2021)
18. Luo, C., Cheng, C., Zheng, Q., Yao, C.: Geolayoutlm: Geometric pre-training for visual information extraction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7092–7101 (2023)
19. Mangrulkar, S., Gugger, S., Debut, L., Belkada, Y., Paul, S., Bossan, B.: Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft` (2022)
20. Naveed, H., Khan, A.U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Barnes, N., Mian, A.: A comprehensive overview of large language models. arXiv preprint arXiv:2307.06435 (2023)
21. Park, S., Shin, S., Lee, B., Lee, J., Surh, J., Seo, M., Lee, H.: Cord: a consolidated receipt dataset for post-ocr parsing. In: Workshop on Document Intelligence at NeurIPS 2019 (2019)
22. Perot, V., Kang, K., Luisier, F., Su, G., Sun, X., Boppana, R.S., Wang, Z., Mu, J., Zhang, H., Hua, N.: Lmdx: Language model-based document information extraction and localization. arXiv preprint arXiv:2309.10952 (2023)

23. Roziere, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X.E., Adi, Y., Liu, J., Remez, T., Rapin, J., et al.: Code llama: Open foundation models for code. arXiv preprint arXiv:2308.12950 (2023)
24. Sassioui, A., Benouini, R., El Ouargui, Y., El Kamili, M., Chergui, M., Ouzzif, M.: Visually-rich document understanding: Concepts, taxonomy and challenges. In: 2023 10th International Conference on Wireless Networks and Mobile Communications (WINCOM). pp. 1–7. IEEE (2023)
25. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023)
26. Wang, Z., Zhou, Y., Wei, W., Lee, C.Y., Tata, S.: A benchmark for structured extractions from complex documents. arXiv preprint arXiv:2211.15421 (2022)
27. Xu, Y., Xu, Y., Lv, T., Cui, L., Wei, F., Wang, G., Lu, Y., Florencio, D., Zhang, C., Che, W., et al.: Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. arXiv preprint arXiv:2012.14740 (2020)
28. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., Zhou, M.: 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. pp. 1192–1200 (2020)