

# MGDoc: Pre-training with Multi-granular Hierarchy for Document Image Understanding

Zilong Wang<sup>1\*</sup> Jiuxiang Gu<sup>2</sup> Chris Tensmeyer<sup>2</sup> Nikolaos Barmpalios<sup>2</sup>  
Ani Nenkova<sup>2</sup> Tong Sun<sup>2</sup> Jingbo Shang<sup>1</sup> Vlad I. Morariu<sup>2</sup>

<sup>1</sup>University of California, San Diego <sup>2</sup>Adobe Research

<sup>1</sup>{zlwang, jshang}@ucsd.edu

<sup>2</sup>{jigu, tensmeyer, barmpalios, nenkova, tsun, morariu}@adobe.com

## Abstract

Document images are a ubiquitous source of data where the text is organized in a complex hierarchical structure ranging from fine granularity (e.g., words), medium granularity (e.g., regions such as paragraphs or figures), to coarse granularity (e.g., the whole page). The spatial hierarchical relationships between content at different levels of granularity are crucial for document image understanding tasks. Existing methods learn features from either word-level or region-level but fail to consider both simultaneously. Word-level models are restricted by the fact that they originate from pure-text language models, which only encode the word-level context. In contrast, region-level models attempt to encode regions corresponding to paragraphs or text blocks into a single embedding, but they perform worse with additional word-level features. To deal with these issues, we propose MGDoc, a new multi-modal multi-granular pre-training framework that encodes page-level, region-level, and word-level information at the same time. MGDoc uses a unified text-visual encoder to obtain multi-modal features across different granularities, which makes it possible to project the multi-granular features into the same hyperspace. To model the region-word correlation, we design a cross-granular attention mechanism and specific pre-training tasks for our model to reinforce the model of learning the hierarchy between regions and words. Experiments demonstrate that our proposed model can learn better features that perform well across granularities and lead to improvements in downstream tasks.

## 1 Introduction

Document images are ubiquitous and are often used as a representation for forms, receipts, printed papers, etc. Unlike plain text documents, document images express rich information via both textual

\* This work was completed while the author was working as an intern at Adobe Research.

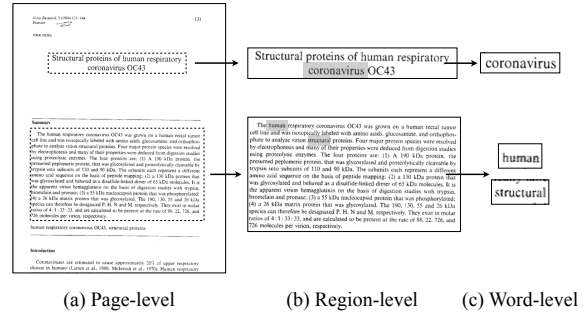


Figure 1: The multi-granular structure in document images. The image is from the RVL-CDIP dataset. Important information is encoded at the page-level (e.g., the document type), region-level (constructs such as paragraphs, tables, etc), and word-level (specific semantics). Our proposed model reasons about all of these jointly to leverage information across granularities.

content and heterogeneous layout patterns, which leads to barriers to the automatic processing of these document images. Here, *layout pattern* refers to how the text is spatially arranged on the document page and involves information from multiple levels of granularity. Specifically, a layout pattern divides the entire page into individual regions and, within each region, the fine-grained textual content is distributed following a certain format, such as paragraphs, columns, lists, as shown in Figure 1.

The layout of a document provides important cues for interpreting the document through spatial structures such as alignment, proximity, and hierarchy between content at different levels of granularity. For example, a numeric text field is more likely to be the total price of a grocery receipt if it is located at the bottom right of a table region; a region is more likely to correspond to the title area of a form if there are a lot of bold types inside of the region. In these two examples, it is important to understand page-level information (e.g., that the document is a receipt or a form), region-level information (e.g., that a region is a table or title), and word/token-level information (e.g., the font

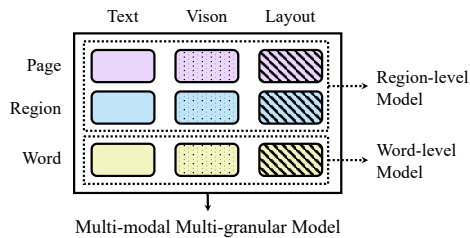


Figure 2: Comparison between MGDoc and existing methods. While previous methods have explored multi-modal inputs, and explored word+page and region+page level features, MGDoc combines multi-modal reasoning, and joint word-, region-, and page-level reasoning.

style of a word, or that a token is a number), as well as how these relate to each other. Therefore, to facilitate the automatic processing of such documents, it is essential to consider the features of multiple granularities and let the model learn the hierarchy between different levels to encode the multi-granular structure in the document images.

However, existing methods in document image understanding formulate the document image understanding tasks either at the word-level or region-level and thus do not use both cues. They mostly follow language modeling methods designed for plain text settings, formulating document image understanding tasks using word-level information and augmenting semantic features with spatial and visual features to exploit the word-level context (Xu et al., 2020b,a; Hong et al., 2021; Garncarek et al., 2021). Recent works go beyond fine-grained word-level inputs and focus on regions instead of words to acquire useful signals (Li et al., 2021; Gu et al., 2021). By encoding the regions corresponding to paragraphs or text blocks, these region-level models manage to save training resources and achieve good performance with rich locality features. However, these models fail to leverage the cross-region word-level correlation, which is also necessary to tackle fine-grained tasks.

Motivated by this observation, we propose MGDoc, a new multi-modal multi-granular pre-training framework that encodes document information at different levels of granularity and represents them using multi-modal features as highlighted in Figure 2. Specifically, we use the OCR engine to decompose a document page into three granularities: page-level, region-level, and word-level. Following previous works (Xu et al., 2020b,a; Gu et al., 2021), our multi-modal features represent text, layout (represented by bounding

boxes), and image modalities. The input consists of information at different levels of granularity, and can be organized into a hierarchy within the page, which means words are included in the corresponding regions and the page includes all of them. We leverage attention to learn the correlation between inputs from different levels of granularity and add special attention weights to encode the hierarchical structure and relative distances (Xu et al., 2020a; Garncarek et al., 2021; Powalski et al., 2021). We rely on pre-training to encourage the model to learn the alignment between regions at different levels of granularity. In addition, we use masked language modeling for the word-level inputs and extend this idea into the more coarse-grained inputs. We mask a proportion of regions and ask the model to minimize the difference between the masked contextual features and the input features corresponding to the selected region.

We validate MGDoc on three public benchmarks, the FUNSD dataset (Jaume et al., 2019) for form understanding, the CORD dataset (Park et al., 2019) for receipt extraction, and the RVL-CDIP dataset (Lewis et al., 2006) for document image classification. Extensive experiments demonstrate the effectiveness of our proposed approach with great improvements on fine-grained tasks and good results on coarse-grained tasks. We summarize our contribution as follows:

- We propose MGDoc, a multi-modal multi-granular pre-training framework, which encodes the hierarchy in document images and integrates features from text, layout patterns, and images.
- A cross-granularity attention mechanism and a new pre-training task designed to enable the model to learn the alignment between different levels. This work extends the masked language modeling to different granularity to encode the contextual information.
- Extensive experiments demonstrate the effectiveness of MGDoc on three representative benchmarks.

## 2 Method

### 2.1 Overview

MGDoc is a multi-modal multi-granular pre-training framework for document image understanding tasks. The framework encodes features from different levels of granularity in a document page and leverages the spatial hierarchy between them. There are three stages in our architecture.

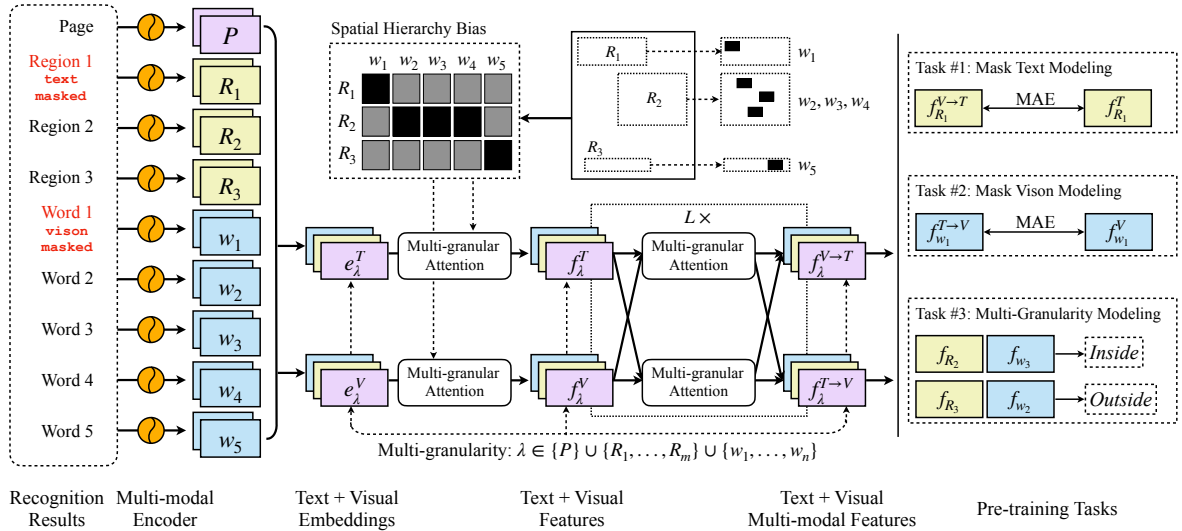


Figure 3: The multi-modal multi-granular pre-training framework of MGDoc. Inputs consist of visual, textual, and spatial representations of the page, regions on the page, and individual words. Multi-granular attention learns relationships within and across granularities within single modalities, followed by multi-granular attention across modalities. The final output consists of an embedding for each region at each granularity, on top of which three self-supervision tasks are added to pre-train the model. Specifically, the multi-granularity task ensures that our model makes use of multi-granular inputs and multi-granular attention to solve spatial tasks.

First, OCR engine, human annotations, or digital parsing provide us with the text and the bounding boxes of contents at different levels of granularity. We focus on pages, regions, and words in this paper and leave the more fine-grained pixel-level and coarse-grained multi-page modeling for future work. We input the document image, textual content, and bounding boxes at these three levels to the multi-modal encoder to encode multi-granular information into text and image embeddings. Following previous work (Xu et al., 2020b), we use spatial embeddings to encode spatial layout information. Next, we design a multi-granular attention mechanism to extract the correlation between the features from different levels. Distinct from the normal self-attention mechanism in BERT (Lu et al., 2019), multi-granular attention computes the dot product between the features and encodes the hierarchical relationship between regions and words by adding an attention bias. Then, the cross-attention mechanism is used to combine the features from different modalities. Finally, the sum of the final text and visual features is used in the pre-training or fine-tuning tasks.

## 2.2 Multi-modal Encoding

The multi-modal encoding is designed to encode the text, visual, and spatial information of the multi-granular inputs into the embedding space. We first

acquire the inputs at the word-level, encoding the text and bounding box of each word. At the region-level, words are grouped into regions where all words within the region are combined and the enclosing bounding box of all the words is used as the region bounding box. At the page-level, the textual input is the sequence of all words in the page and the width and height of the document image is used as the page-level bounding box. Now, the inputs at different levels of granularity consist of the textual content and the bounding box. We denote the inputs as  $P$ ,  $\{R_1, \dots, R_m\}$ ,  $\{w_1, \dots, w_n\}$ , where  $m, n$  are the number of regions and words.

The multi-modal encoding takes the textual content of each input unit, ranging from a single word, to sentences, to the whole textual content of a page, and encodes it with a pre-trained language model, e.g., SBERT (Reimers and Gurevych, 2019). We add the spatial embeddings to the text encoder outputs where the fully-connected layer is used to project the bounding boxes into a hyperspace. In this way, the text embeddings of our model are augmented with spatial information. Then, MGDoc encodes the entire image with a visual feature extractor, e.g., ResNet (He et al., 2016), and extracts region feature maps using bounding boxes as the Region of Interest (ROI) areas. The results of the vision encoder have different resolutions due to the sizes of bounding boxes, but they lie in the

same feature space. Similarly, we add the spatial embeddings from the bounding boxes to the visual embeddings. The multi-modal embeddings are represented as follows,

$$\begin{aligned} e_\lambda^T &= \text{Enc}^T(\text{text}_\lambda) + \text{FC}(\text{box}_\lambda) + \text{Emb}^T \\ e_\lambda^V &= \text{Enc}^V(\text{img})[\text{box}_\lambda] + \text{FC}(\text{box}_\lambda) + \text{Emb}^V \end{aligned}$$

where  $e_\lambda^T$  and  $e_\lambda^V$  are the text and visual embeddings;  $\lambda \in \{P\} \cup \{R_1, \dots, R_m\} \cup \{w_1, \dots, w_n\}$  denotes different levels of granularity;  $\text{Enc}^T$  and  $\text{Enc}^V$  are the text and image encoders, respectively;  $\text{text}$  and  $\text{box}$  refer to the textual contents and bounding boxes;  $\text{img}$  is the entire document image;  $\text{FC}(\cdot)$  is the fully-connected layer;  $\text{Emb}^T$  and  $\text{Emb}^V$  are the type embeddings for text and vision.

### 2.3 Multi-granular Attention

Given the multi-modal embeddings described above, we design a multi-granular attention to encode the hierarchical relation between regions and words. Specifically, we add attention biases to the original self-attention weights to strengthen the region-word correlation. We apply multi-granular attention to the text embeddings and visual embeddings individually because the purpose of this module is to learn the interaction between different levels of granularity rather than to fuse modalities. Therefore, without loss of generality, we omit the notation of modality in the expressions. The attention weight is computed as

$$\begin{aligned} A_{\alpha,\beta} &= \frac{1}{\sqrt{d}} (W^Q e_\alpha)^\top (W^K e_\beta) \\ &+ \text{HierBias}(\text{box}_\alpha \subseteq \text{or } \not\subseteq \text{box}_\beta) \\ &+ \text{RelBias}(\text{box}_\alpha - \text{box}_\beta), \end{aligned}$$

where  $\alpha, \beta \in \{P\} \cup \{R_1, \dots, R_m\} \cup \{w_1, \dots, w_n\}$ ; the first part is the same as the attention mechanism in the original BERT;  $\text{RelBias}(\cdot)$  is the attention weight bias to encode the relative distance between the bounding boxes;  $\text{HierBias}(\cdot)$  is the attention weight bias to encode the inside or outside relation which models the spatial hierarchy within the page. Since the regions are created by grouping the words, all the words correspond to a specific region. We embed the binary relation into a fixed-sized vector and each value in this vector is added to an attention head of the multi-granular attention module. After the multi-granular attention module, self-attention is applied to the

input embeddings to learn contextual information and hierarchical relationships between the multi-granular inputs. We denote the resulting textual and visual features by  $f_\lambda^T$  and  $f_\lambda^V$ , respectively, where  $\lambda \in \{P\} \cup \{R_1, \dots, R_m\} \cup \{w_1, \dots, w_n\}$ .

### 2.4 Cross-modal Attention

As mentioned in Section 2.3, multi-granular attention is only designed for the interaction between different levels of granularity; however, it is also essential to fuse information from multiple modalities. Therefore, we design the cross-modal attention module to conduct the modality fusion. Following previous works in visual-language modeling, we use a cross-attention mechanism to fuse the textual and visual features. Specifically, the cross-attention function is formulated as,

$$\text{CrossAttn}(\alpha|\beta) = \sigma \left( \frac{(W^Q \alpha)^\top (W^K \beta)}{\sqrt{d}} \right) W^V \beta$$

where  $\alpha$  and  $\beta$  are matrices of the same size;  $\sigma$  is the Softmax function to normalize the attention matrix;  $W^Q, W^K, W^V$  are the trainable weights in each of the attention heads for query, key and value. Then we list text and visual features from different levels of granularity as  $F^T = \{f_\lambda^T\}$  and  $F^V = \{f_\lambda^V\}$  and compute the multi-modal features as follows,

$$\begin{aligned} f_\lambda^{T \rightarrow V} &= \text{CrossAttn}(F^T | F^V)[\lambda] \\ f_\lambda^{V \rightarrow T} &= \text{CrossAttn}(F^V | F^T)[\lambda] \end{aligned}$$

From these expressions, we can see that the cross attention uses the dot product between multi-modal features as attention weights. In this way, the given modality can learn from the other modality and the module also bridges the gap between the modalities. We call the output of this module text or visual multi-modal features to distinguish from the text and visual features in Section 2.2, and denote them as  $f_\lambda^{T \rightarrow V}$  and  $f_\lambda^{V \rightarrow T}$ . The final representation is the sum of the textual and visual multi-modal features,  $f_\lambda = f_\lambda^{T \rightarrow V} + f_\lambda^{V \rightarrow T}$ ,  $\lambda \in \{P\} \cup \{R_1, \dots, R_m\} \cup \{w_1, \dots, w_n\}$ , and is used in the pre-training and downstream tasks.

### 2.5 Pre-training Tasks

Large-scale pre-training has shown strong results in document image understanding tasks. With a large amount of unlabeled data, pre-trained models can learn the latent data distribution without manually labelled supervision and can easily transfer

the learned knowledge to downstream tasks. The design of pre-training tasks is crucial for successful pre-training. We go beyond the classic mask modeling and apply the mask text modeling and the mask vision modeling on all the inputs from different levels of granularity. Due to the unified multi-model encoder (see Section 2.2), it is possible for us to treat all levels of granularity equally and introduce a unified masking task for each modality. Because we believe that spatial hierarchical relationships are essential for encoding documents, we design a pre-training task that requires the model to identify the spatial relationship between content at different levels of granularity. The final training loss is the sum of the pre-training tasks,  $\mathcal{L} = \mathcal{L}_{MTM} + \mathcal{L}_{MVM} + \mathcal{L}_{MGM}$ . Below we provide details for each component.

**Mask Text Modeling** The mask text modeling task requires the model to understand the textual inputs of the model. Specifically, we randomly select a proportion of regions or words, and their textual contents are replaced with a special token [MASK]. We run the model to obtain the contextual features of these masked inputs and compare them with the encoding result of original textual inputs. We use the Mean Absolute Error as the loss function.

$$\mathcal{L}_{MTM} = \sum_{\lambda \in \Lambda} \left| e_{\lambda}^T - f_{[\text{MASK}]|\bar{\lambda}}^{V \rightarrow T} \right|$$

where  $\Lambda = \{P\} \cup \{R_1, \dots, R_m\} \cup \{w_1, \dots, w_n\}$ ;  $e_{\lambda}^T$  denotes the encoding result of the original textual contents;  $\bar{\lambda}$  denotes the multi-granular context without  $\lambda$ ;  $f_{[\text{MASK}]|\bar{\lambda}}^{V \rightarrow T}$  is the contextual feature of the masked textual inputs.

**Mask Vision Modeling** Similarly to the mask text modeling task, we use mask vision modeling to learn visual contextual information. Instead of replacing the [MASK] token as is done in mask text modeling, we set the visual embeddings of the selected areas to zero vectors. The loss function computes the Mean Absolute Error between the contextual feature of masked areas and the original visual embeddings. The mask vision modeling loss is formulated as,

$$\mathcal{L}_{MVM} = \sum_{\lambda \in \Lambda} \left| e_{\lambda}^V - f_{[\emptyset]|\bar{\lambda}}^{T \rightarrow V} \right|$$

where  $f_{[\emptyset]|\bar{\lambda}}^{T \rightarrow V}$  is the contextual feature of the zero vector given the unmasked inputs.

**Multi-Granularity Modeling** The multi-granularity modeling task asks the model to understand the spatial hierarchy between different levels of granularity. Since the page-level input includes all regions and words, it is trivial for the model to learn it. We only focus on the hierarchical relation between the regions and words. Although the relation is also encoded in the multi-granular attention, it is necessary to reinforce the model to emphasize the region-word correspondence. Otherwise, the spatial hierarchy biases are random add-ons to the attention matrix.

The model takes the region-level and word-level features and predicts which region the given word is located in. We first compute the dot product of the region-level and word-level features as the score and use the Cross-entropy as the loss function.

$$\mathcal{L}_{MGM} = \sum_{w \in W} \frac{e^{f_w^T f_{r^*}}}{e^{f_w^T f_{r^*}} + \sum_{r \in R - \{r^*\}} e^{f_w^T f_r}}$$

where  $W = \{w_1, \dots, w_n\}$  and  $R = \{R_1, \dots, R_m\}$ ;  $r^*$  is the region that includes the word  $w$ .

## 3 Experiments

### 3.1 Pre-training Settings

We use the RVL-CDIP dataset (Harley et al.) as our pre-training corpus. The RVL-CDIP dataset is a scanned document image dataset containing 400,000 grey-scale images and covering a variety of layout patterns. We use OCR engines to recognize the location of textual content in the document images and also the location of the individual words. Following Gu et al. (2021), we use EasyOCR<sup>1</sup> with two different output modes: non-paragraph and paragraph. The difference is that the non-paragraph mode extracts the individual words in the pages, and the paragraph mode groups these results into regions. The OCR engine allows us to design the architecture and the pre-training tasks focusing on the multi-granularity of document images. Therefore, the paragraph results serve as the region-level inputs, and the non-paragraph results serve as the word-level inputs.

### 3.2 Fine-tuning Tasks

We select three representative tasks to evaluate the performance of our model and use the publicly-available benchmarks for each tasks.

<sup>1</sup><https://github.com/JaidedAI/EasyOCR>

Scale	Model	Pre-training			FUNSD	CORD	RVL-CDIP	
		Corpus	#Data	#Param.	(F1)	(F1)	(Accuracy)	
Word	BERT <sub>BASE</sub>	-	-	110M	60.26	89.68	89.81	
	BERT <sub>LARGE</sub>	-	-	340M	65.63	90.25	89.92	
	LayoutLM <sub>BASE</sub>	IIT-CDIP	11M	113M	78.66	94.72	94.42	
	LayoutLM <sub>LARGE</sub>	IIT-CDIP	11M	343M	78.95	94.93	94.43	
	BROS <sub>BASE</sub>	IIT-CDIP	11M	110M	83.05	96.50	-	
	BROS <sub>LARGE</sub>	IIT-CDIP	11M	340M	84.52	97.28	-	
	LayoutLMv2 <sub>BASE</sub>	IIT-CDIP	11M	200M	82.76	94.95	95.25	
	LayoutLMv2 <sub>LARGE</sub>	IIT-CDIP	11M	426M	84.20	96.01	95.64	
	TILT <sub>BASE</sub>	RVL-CDIP+	1.1M	230M	-	95.11	95.25	
	TILT <sub>LARGE</sub>	RVL-CDIP+	1.1M	780M	-	96.33	95.52	
	DocFormer <sub>BASE</sub>	IIT-CDIP-	5M	183M	83.34	96.33	96.17	
	DocFormer <sub>LARGE</sub>	IIT-CDIP-	5M	536M	84.55	96.99	95.50	
	Region	SelfDoc	RVL-CDIP	320K	-	83.36	-	92.81
		SelfDoc+VGG-16	RVL-CDIP	320K	-	-	-	93.81
UDoc		IIT-CDIP-	1M	272M	87.96	96.64	93.96	
UDoc <sup>‡</sup>		IIT-CDIP-	1M	272M	87.93	96.86	95.05	
<b>Region+Word</b>	<b>MGDoc (Ours)</b>	<b>RVL-CDIP</b>	<b>320K</b>	<b>203M*</b>	<b>89.44</b>	<b>97.11</b>	<b>93.64</b>	

Table 1: The experiment results and comparison. \* indicates that non-trainable parameters are not included. The total #param. of MGDoc is 312M. ‡ implies unfreezing the sentence encoder during the finetuning. RVL-CDIP+: TILT uses extra training pages in pre-training; IIT-CDIP-: DocFormer uses a subset of IIT-CDIP in pre-training.

**Form Understanding** The goal of the form understanding task is to predict the label of semantic entities in document images. We use the FUNSD dataset (Jaume et al., 2019) for this task. The FUNSD dataset consists of 199 fully-annotated, noisy-scanned forms with various appearances and formats. There are 149 and 50 pages in the training set and the testing set, respectively. Each entity is labeled into 3 categories: *Header*, *Question*, and *Answer*. We use the provided OCR results from the dataset and input the textual contents and bounding boxes of entities to the model. We report the entity-level F1 score as metrics.

**Receipt Understanding** The goal of the receipt understanding task is to recognize the role of a series of text lines in a document. We use the CORD dataset (Park et al., 2019) for this task. The CORD dataset is fully annotated with bounding boxes and textual contents and contains 800 and 100 pages in the training and testing sets, respectively. There are 30 entity types marked in the dataset; we report entity-level F1 score for our experiments.

**Document Image Classification** The document image classification task aims to classify the pages into different semantic categories. We use the RVL-CDIP dataset (Harley et al.) for this task, which is a subset of the IIT-CDIP dataset (Lewis et al., 2006). The RVL-CDIP dataset contains 400,000 pages, each annotated with 16 semantic categories. The input features for this dataset are extracted by the EasyOCR engine in our experiments. The RVL-

CDIP dataset is also used in the pre-training, but no labeling information is involved in the pre-training tasks, so there is no concern about data leakage. In the downstream task, the RVL-CDIP dataset is divided into training, validation, and test subsets with 8:1:1 ratio. We report classification accuracy over the 16 categories for our experiments.

### 3.3 Implementation Details

In the multi-modal encoder, we use the BERT-NLI-STSb-base model as the text encoder and ResNet-50 as the vision encoder. In the modality fusion, we use 12 layers of cross-modal attention in MGDoc. We set the hidden state size as 768 and the attention head number as 12. We freeze the pre-trained weights of the multi-modal encoder and randomly initialize the remaining parameters, which are then learned during our pre-training stage. We run the pre-training for 5 epochs with 8 NVIDIA V100 32G GPUs and the AdamW optimizer. The batch size is set to 64; the learning rate is set to  $10^{-6}$ ; the warmup is conducted in the first 20% training steps.

### 3.4 Results

We compare MGDoc with the strong baselines in the document understanding tasks in Table 1. We list out the specific settings of each model in the layout-rich pre-training, to clearly demonstrate the effectiveness of our model. All these baseline models resort to different techniques to achieve competitive results. BERT (Devlin et al., 2018), Lay-

outLM (Xu et al., 2020b), LayoutLMv2 (Xu et al., 2020a), BROS (Hong et al., 2021), TILT (Powalski et al., 2021), and DocFormer (Appalaraju et al., 2021) encode word-level features, and SelfDoc (Li et al., 2021), and UDoc (Gu et al., 2021) encodes region-level features. MGDoc surpasses all the existing methods with the help of the information from all different levels of granularity, and achieves a new state-of-the-art performance in the fine-grained tasks, *i.e.*, the form understanding task and receipt understanding task. It also achieves promising performance on the coarse-grained task, *i.e.*, the document image classification task. Specifically, MGDoc improves the entity-level F1 score of the FUNSD dataset by 1.48% and improves the entity-level F1 score of the CORD dataset by 0.25%, compared with the second-best model. We partially attribute the performance difference on the RVL-CDIP dataset to the OCR engine, since LayoutLMv2 and TILT use the Microsoft OCR and BROS uses the CLOVA OCR, and these commercial OCR engines provide more accurate results. As discussed in Gu et al. (2021), the quality of the OCR engine influences the performance of the document image classification. It is also worth mentioning that our model involves relatively smaller number of trainable parameters and also requires less pre-training data, which makes MGDoc more applicable in realistic scenarios.

The performance of the form and receipt understanding tasks is improved by region-level information. UDoc surpasses the word-scale models by large margins, and our proposed, MGDoc, even further improves the UDoc by modeling the alignment between regions and words. We conclude that the region-level information strengthens the locality of the feature extraction, and the word-level information further improves the classification results. Such connection is realized by region-word alignment, which is visualized in Section 3.6.

### 3.5 Ablation Study

To study the importance of the pre-training tasks, we design an ablation study that skips several pre-training tasks. The results are shown in Table 2. In the first setting, we skip the entire pre-training stage so all the parameters can only be learned in the downstream tasks. In the second setting, we include the commonly-used masking techniques. The model is pre-trained with the two masking tasks in our design, the mask sentence modeling, and

Model	FUNSD (F1)	RVL-CDIP (Accuracy)
MGDoc		
<i>w/o pre-training</i>	83.01	91.23
<i>w/ MTM+MVM</i>	87.20	93.92
<b><i>w/ MTM+MVM+MGM</i></b>	<b>89.44</b>	<b>93.64</b>

Table 2: The results of the ablation study for pre-training. Performance steadily increases as pre-training tasks are added.

Model	FUNSD (F1)	CORD (F1)
MGDoc		
<i>w/ Region</i>	80.82	94.24
<i>w/ Region + Word</i>	86.96	95.49
<i>w/ Page + Region</i>	81.65	94.69
<b><i>w/ Page + Region + Word</i></b>	<b>89.44</b>	<b>97.11</b>

Table 3: The results of the ablation study for multi-granular features. We observe a steady increment with more features involved, and the word-level features contribute most to the improvement.

the mask vision modeling. Performance steadily increases as pre-training tasks are added; overall, pre-training improves the performance by 6.43%, 2.69% on FUNSD and RVL-CDIP, respectively.

We believe that the masking strategy enables the model to learn from the multi-modal context of the page. In the third experiment in the table, we add the alignment techniques between words and regions designed to strengthen the connection between multiple granularities. The performance on FUNSD is further improved by 2.24%, while there is also a decrease of 0.28% in the performance on RVL-CDIP. Local connections between words and regions are helpful in fine-grained tasks but may introduce some noise to coarse-grained tasks.

To study the role of features from each granularity, we also conduct an ablation study using different combinations of multi-granular features, where we feed the model with features from region-level inputs, region-level and word-level inputs, page-level and region-level inputs, respectively. We report the performance on FUNSD and CORD in Table 3. We observe a steady increment with more features involved, and the word-level features contribute more to the improvement.

### 3.6 Region-word Correlation Visualization

We visualize the correlation between regions and words using heat maps. We select four examples from the FUNSD dataset and show the heat maps of the final feature dot product in Figure 5. The x-axis

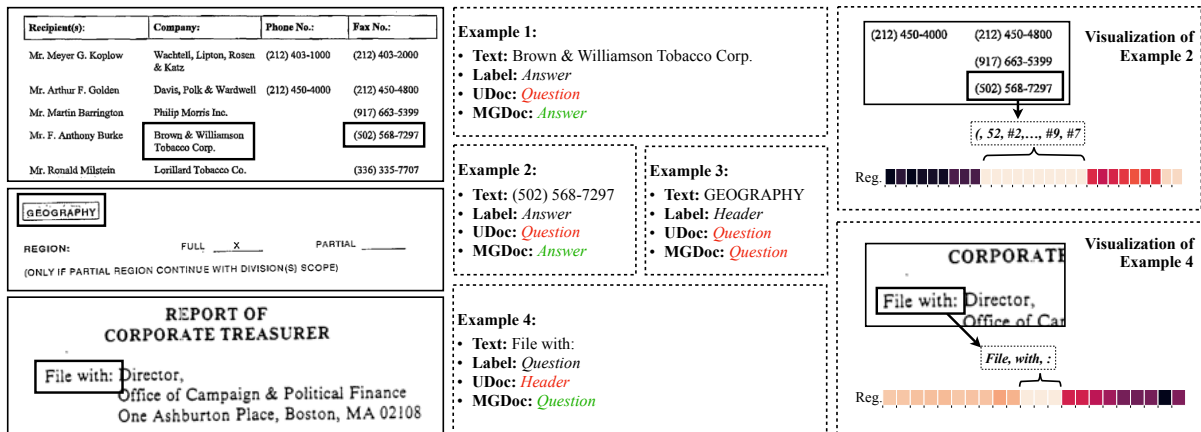


Figure 4: The error analysis and visualization

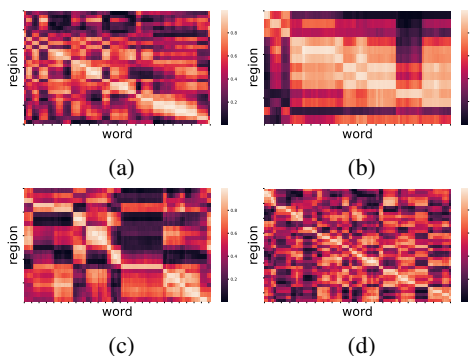


Figure 5: The correlation weight visualization between regions and words

and the y-axis correspond to the words and regions, respectively, and the lighter the color is, the higher correlation there is. Some cropping is applied for clearer visualization. From the heat maps, we can observe that there are highlighted areas along the matrix diagonal, which means our model learns the region-word hierarchy in the pre-training stage and can leverage such correspondence in downstream tasks. We also see some lighter colored blocks in the matrix. Since all the words and regions are serialized in positional order, these lighter colored blocks indicate the model is able to use the localized features in the model with the help of multi-granular inputs. This ability further confirms that our intuition that combining information from different levels of granularity will be beneficial is correct.

### 3.7 Error Analysis

We select several representative cases in the comparison between UDoc and MGDoc and show them in Figure 4. We also visualize the weight matrix

of the entities in the same way as in Section 3.6. In these examples, our proposed model can leverage the more fine-grained signal from word-level inputs and make the correct prediction. In example 2, the entity is labeled as *Answer* where the corresponding question, “Fax No.:", is at the top of this column. Due to the large distance of this question-answer pair, UDoc predicts the entity as *Question*, while MGDoc can give the right prediction by directly learning from the digits inside of the text fields, which is a strong signal for answers. From the heat map, we can also see that a lighter color appears in the corresponding area of the entity. Meanwhile, the word-level information even strengthens the multi-modal features since it provides more details of a given text field. As we can observe in example 4, the entity “File with:” is likely to be *Header* or *Question* given its textual contents and location in the page, but MGDoc can predict from the rich visual features that this field is a part of normal text and less likely to be *Header*; these rich inputs allow MGDoc to make the correct prediction where UDoc cannot. However, in example 3, both UDoc and MGDoc cannot predict correctly. The ground-truth label is *Header* but both models predict the entity as *Question*. The entity is not at the top of the page where the header entities are more likely to be located, so we attribute this error to the dependence of MGDoc to the spatial information.

## 4 Related Work

**Word-level Models** Word-level models inherit the architecture of pure-text pre-trained language models. Word-level contextual information is encoded by a multi-layered transformer, and spatial



and visual features are added to refine the representation. Inspired by the positional embeddings in Vaswani et al. (2017); Raffel et al. (2019); Dai et al. (2019), absolute or relative spatial features based on the bounding boxes are proposed to encode the words' layout with respect to each other (Xu et al., 2020b,a; Hong et al., 2021; Garncarek et al., 2021). Computer vision deep models (He et al., 2016; Xie et al., 2017) are used to extract features from the document images, and self-supervised learning methods are applied to learn the cross-modal correlation between images and words (Xu et al., 2020a; Powalski et al., 2021).

**Region-level Models** Region-level models encode the regions in the document page including text blocks, headings, and paragraphs (Li et al., 2021; Gu et al., 2021). Similar spatial and visual features are used in these models as in the word-level models. With the help of coarse-grained inputs, region-level models can emphasize the rich locality features and catch high-level cues. Another difference with the word-level models is that the number of regions is much smaller than the word number on the page, so the region-level models are more efficient when processing long documents.

## 5 Conclusions and Future Work

We present MGDoc, a multi-modal multi-granular pre-training framework, which goes beyond the existing region-level or word-level models and leverages the contents at multiple levels of granularity to understand the document pages better. Existing models fail to use the informative multi-granular features in the document due to the restriction from the word-level model architecture, and lead to unsatisfactory results. We solve these issues with the new architecture design and tailored pre-training tasks. With a unified multi-modal encoder, we embed the features from pages, regions, and words into the same hyperspace, and design a multi-granular attention mechanism and multi-granularity modeling task for MGDoc to learn the spatial hierarchical relation between them. Experiments show that our proposed model can understand the spatial relation between the multi-granular features and lead to improvements in downstream tasks.

As for future work, since we have not fully exploited the multi-granular information, we will go beyond the page level and investigate the possibility of encoding multiple pages. We are also interested in inputs that are more fine-grained than word level,

such as pixels.

## 6 Acknowledgement

This work was supported in part by Adobe Research. We thank anonymous reviewers and program chairs for their valuable and insightful feedback.

## Limitations

Although we inherit the idea of using region-level inputs from (Gu et al., 2021; Li et al., 2021), we cannot keep their merits of saving computing resources. Region-level models encode regions instead of all the words in the page, so the smaller number of features are included in the self-attention layers. However, we want to leverage the fine-grained word-level information as (Xu et al., 2020b,a; Hong et al., 2021), so the words are also considered in the multi-granular attention and the multi-modal attention layers. Compared to existing works, our work requires more memory storage during training and testing.

## Ethical Considerations

This paper presents a new framework for document image understanding tasks. Our model is built on open-source tools and datasets, and we aim at increasing the efficiency of processing various documents and also bringing convenience to ordinary people's life. Thus, we do not anticipate any major ethical concerns.

## 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*, pages 993–1003.
- Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, Michał Turski, and Filip Graliński. 2021. Lambert:

- Layout-aware language modeling for information extraction. In *International Conference on Document Analysis and Recognition*, pages 532–547. Springer.
- Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Nikolaos Barmpalios, Ani Nenkova, and Tong Sun. 2021. Unidoc: Unified pretraining framework for document understanding. *Advances in Neural Information Processing Systems*, 34:39–50.
- Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *International Conference on Document Analysis and Recognition (ICDAR)*.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. 2021. Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents. *arXiv preprint arXiv:2108.04539*.
- Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 2, pages 1–6. IEEE.
- 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*, pages 665–666.
- Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. 2021. Selfdoc: Self-supervised document representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5652–5660.
- Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.
- 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 *International Conference on Document Analysis and Recognition*, pages 732–747. Springer.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.
- Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500.
- Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. 2020a. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. *arXiv preprint arXiv:2012.14740*.
- Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020b. 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.