DocFormer: End-to-End Transformer for Document Understanding

Srikar Appalaraju
AWS AI
srikara@amazon.com

Bhavan Jasani
AWS AI
bjasani@amazon.com

Bhargava Urala Kota
AWS AI
bharkota@amazon.com

Yusheng Xie
AWS AI
yushx@amazon.com

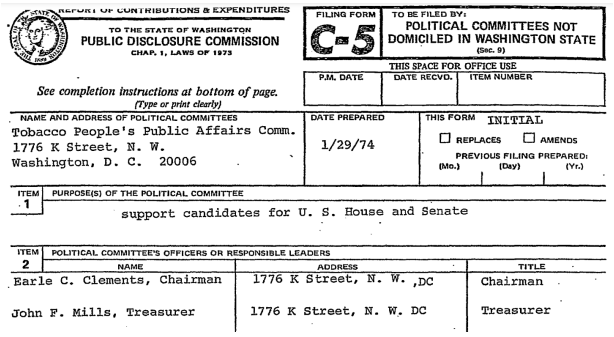
R. Manmatha
AWS AI
manmatha@amazon.com

Abstract

We present DocFormer - a multi-modal transformer based architecture for the task of Visual Document Understanding (VDU). VDU is a challenging problem which aims to understand documents in their varied formats (forms, receipts etc.) and layouts. In addition, DocFormer is pre-trained in an unsupervised fashion using carefully designed tasks which encourage multi-modal interaction. DocFormer uses text, vision and spatial features and combines them using a novel multi-modal self-attention layer. DocFormer also shares learned spatial embeddings across modalities which makes it easy for the model to correlate text to visual tokens and vice versa. DocFormer is evaluated on 4 different datasets each with strong baselines. DocFormer achieves state-of-the-art results on all of them, sometimes beating models 4x its size (in no. of parameters).

1. Introduction

The task of Visual Document Understanding (VDU) aims at understanding digital documents either born as PDF's or as images. VDU focuses on varied document related tasks like entity grouping, sequence labeling, document classification. While modern OCR engines [33] have become good at predicting text from documents, VDU often requires understanding both the structure and layout of documents. The use of text or even text and spatial features alone is not sufficient for this purpose. For the best results, one needs to exploit the text, spatial features and the image. One way to exploit all these features is using transformer models [4, 14, 51]. Transformers have recently been used for VDU [25, 54, 55]. These models differ in how the unsupervised pre-training is done, the way self-attention is modified for the VDU domain or how they fuse modalities (text and/or image and spatial). There have been text only [14], text plus spatial features only [25, 54] approaches for VDU. However, the holy-grail is to fuse all three modalities (text, visual and spatial features). This is desirable since there is some information in text that visual features miss out (language semantics), and there is some information in visual features that text misses out (text font and visual layout for example).


REPORT OF CONTRIBUTIONS & EXPENDITURES
TO THE STATE OF WASHINGTON
PUBLIC DISCLOSURE COMMISSION
CHAP. 1, LAWS OF 1973
FILING FORM C-5
TO BE FILED BY: POLITICAL COMMITTEES NOT DOMICILED IN WASHINGTON STATE (Sec. 9)
THIS SPACE FOR OFFICE USE
P.M. DATE | DATE RECVD. | ITEM NUMBER
See completion instructions at bottom of page. (Type or print clearly)
NAME AND ADDRESS OF POLITICAL COMMITTEES
Tobacco People's Public Affairs Comm.
1776 K Street, N. W.
Washington, D. C. 20006
DATE PREPARED
1/29/74
THIS FORM INITIAL
REPLACES AMENDS
PREVIOUS FILING PREPARED: (Mo.) (Day) (Yr.)
ITEM 1 PURPOSE(S) OF THE POLITICAL COMMITTEE
support candidates for U. S. House and Senate
ITEM 2 POLITICAL COMMITTEE'S OFFICERS OR RESPONSIBLE LEADERS

NAME	ADDRESS	TITLE
Earle C. Clements, Chairman	1776 K Street, N. W. ,DC	Chairman
John F. Mills, Treasurer	1776 K Street, N. W. DC	Treasurer

Figure 1: **Snippet of a Document**: Various VDU tasks on this document may include labeling each text token into fixed classes or grouping tokens into a semantic class and finding relationships between tokens e.g. ("DATE PREPARED" → Key and "1/29/74" → Value) or classifying the document into different categories. Note a document could have "other" text e.g. "C-5" which the model should ignore or classify as "other" depending on the task.

Multi-modal training in general is difficult since one has to map a piece of text to an arbitrary span of visual content. For example in Figure 1, "ITEM 1" needs to be mapped to the visual region. Said a different way, text describes semantic high-level concept(s) e.g. the word "person" whereas visual features map to the pixels (of a person) in the image. It is not easy to enforce feature correlation across modalities from text ↔ image. We term this issue as *cross-modality feature correlation* and reference it later to show how DocFormer presents an approach to address this.

DocFormer follows the now common, pre-training and fine-tuning strategy. DocFormer incorporates a novel multi-modal self-attention with shared spatial embeddings in an encoder only transformer architecture. In addition, we pro-

pose three pre-training tasks of which two are novel unsupervised multi-modal tasks: *learning-to-reconstruct* and *multi-modal masked language modeling* task. Details are provided in Section 3. To the best of our knowledge, this is the first approach for doing VDU which does not use bulky pre-trained object-detection networks for visual feature extraction. DocFormer instead uses plain ResNet50 [21] features along with shared spatial (between text and image) embeddings which not only saves memory but also makes it easy for DocFormer to correlate text, visual features via spatial features. DocFormer is trained end-to-end with the visual branch trained from scratch. We now highlight the contributions of our paper:

- A novel multi-modal attention layer capable of fusing text, vision and spatial features in a document.
- Three unsupervised pre-training tasks which encourage multi-modal feature collaboration. Two of these are novel unsupervised multi-modal tasks: *learning-to-reconstruct* task and a *multi-modal masked language modeling* task.
- DocFormer is end-to-end trainable and it does not rely on a pre-trained object detection network for visual features simplifying its architecture. On four varied downstream VDU tasks, DocFormer achieves state of the art results. On some tasks it out-performs large variants of other transformer almost 4x its size (in the number of parameters). In addition, DocFormer does not use custom OCR unlike some of the recent papers [55, 25].

2. Background

Document understanding methods in the literature have used various combinations of image, spatial and text features in order to understand and extract information from structurally rich documents such as forms [18, 57, 12], tables [44, 56, 24], receipts [27, 26] and invoices [35, 42, 37]. Finding the optimal way to combine these multi-modal features is an active area of research.

Grid based methods [29, 13] were proposed for invoice images where text pixels are encoded using character or word vector representations and classified into field types such as Invoice Number, Date, Vendor Name and Address etc. using a convolutional neural network.

BERT [14] is a transformer-encoder [51] based neural network that has been shown to work well on language understanding tasks. LayoutLM [54] modified the BERT architecture by adding 2D spatial coordinate embeddings along with 1D position and text token embeddings. They also added visual features for each word token, obtained using a Faster-RCNN and its bounding box coordinates.

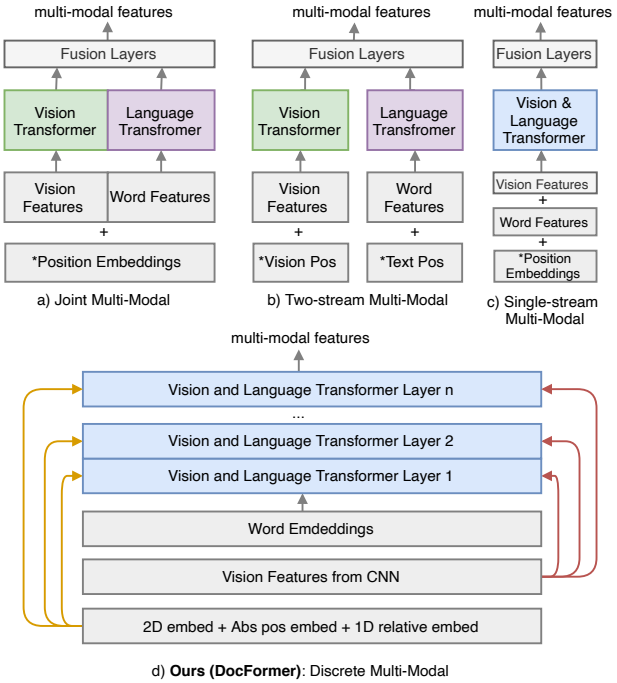


Figure 2: Conceptual Comparisons of **Transformer Multi-Modal Encoder Architectures**: The mechanisms differ in how the modalities are combined. **Type A)** Joint Multi-Modal: like VL-BERT[46], LayoutLMv2[55], VisualBERT [32], MMBT[30], UNITER [8] **Type B)** Two-stream Multi-Modal: CLIP[40], VilBERT[36], **Type C)** Single-stream Multi-Modal, **Type D)** Ours: Discrete Multi-modal. e.g. DocFormer . Note: in each transformer layer, each input modality is self-attended separately. Best viewed in color.

LayoutLM was pre-trained on 11 million unlabeled pages and was then finetuned on several document understanding tasks - form processing, classification and receipt processing. This idea of pre-training on large datasets and then finetuning on several related downstream tasks is also seen in general vision and language understanding work [46, 36, 30, 32] etc. Figure 2 shows a comparison of multi-modal transformer encoder architectures.

Recently, LayoutLMv2 [55] improved over LayoutLM by changing the way visual features are input to the model - treating them as separate tokens as opposed to adding visual features to the corresponding text tokens. Further, additional pre-training tasks were explored to make use of unlabeled document data.

BROS [26] also uses a BERT based encoder, with a graph-based classifier based on SPADE [28], which is used to predict entity relations between text tokens in a document. They also use 2D spatial embeddings added along with text tokens and evaluate their network on forms, receipts document images. Multi-modal transformer encoder-decoder architectures based on T5 [41] have been proposed

recently. Tanaka et al. propose Layout-T5 [48] for a question answering task on a database of web article document images whereas Powalski et al. propose TILT [39] combining convolutional features with the T5 architecture to perform various downstream document understanding tasks.

3. Approach

Conceptual Overview: We first present a conceptual overview of architectures used in Transformer Encoder Multi-Modal training, illustrated in Figure 2. **(a) Joint Multi-Modal**: VL-BERT [46], LayoutLMv2 [55], VisualBERT [32], MMBT [30]: In this type of architecture, vision and text are concatenated into one long sequence which makes transformers self-attention hard due to the *cross-modality feature correlation* referenced in the introduction. **(b) Two-Stream Multi-Modal** CLIP [40], ViLBERT [36]: It is a plus that each modality is a separate branch which allows one to use an arbitrary model for each branch. However, text and image interact only at the end which is not ideal. It might be better to do early fusion. **(c) Single-stream Multi-Modal**: treats vision features also as tokens (just like language) and adds them with other features. Combining visual features with language tokens this way (simple addition) is unnatural as vision and language features are different types of data. (d) **Discrete Multi-Modal**: In this paper, DocFormer unties visual, text and spatial features. i.e. spatial and visual features are passed as residual connections to each transformer layer. We do this because spatial and visual dependencies might differ across layers. In each transformer layer, visual and language features separately undergo self-attention with shared spatial features. In order to pre-train DocFormer we use a subset of 5 million pages from the IIT-CDIP document collection [31] for pre-training. In order to do multi-modal VDU, we first extract OCR, which gives us text and corresponding word-level bounding boxes for each document. We next describe the model-architecture, followed by the pre-training tasks.

3.1. Model Architecture

DocFormer is an encoder-only transformer architecture. It also has a CNN backbone for visual feature extraction. All components are trained end-to-end. DocFormer enforces deep multi-modal interaction in transformer layers using novel multi-modal self-attention. We describe how three modality features (visual, language and spatial) are prepared before feeding them into transformer layers.

Visual features: Let $v \in \mathbb{R}^{3\times h\times w}$ be the image of a document, which we feed through a ResNet50 convolutional neural network $f_{cnn}(\theta, v)$. We extract lower-resolution visual embedding at layer 4 i.e. $v_{l_4} \in \mathbb{R}^{c\times h_l\times w_l}$. Typical values at this stage are $c = 2048$ and $h_l = \frac{h}{32}, w_l = \frac{w}{32}$ (c = number of channels and h_l and w_l are the height and width of the features). The transformer encoder expects a flattened sequence as input of d dimension. So we first apply a 1×1 convolution to reduce the channels c to d. We then flatten the ResNet features to (d, $h_l \times w_l$) and use a linear transformation layer to further convert it to (d, N) where $d = 768, N = 512$. Therefore, we represent the visual embedding as $\overline{V} = linear(conv_{1\times1}(f_{cnn}(\theta, v)))$.

Language features: Let t be the text extracted via OCR from a document image. In order to generate language embeddings, we first tokenize text t using a word-piece tokenizer [53] to get t_{tok}, this is then fed through a trainable embedding layer W_t. t_{tok} looks like $[CLS], t_{tok_1}, t_{tok_2}, \ldots, t_{tok_n}$ where $n = 511$. If the number of tokens in a page is > 511, we ignore the rest. For a document with fewer than 511 tokens, we pad the sequence with a special $[PAD]$ token and we ignore the $[PAD]$ tokens during self-attention computation. We ensure that the text embedding, $\overline{T} = W_t(t_{tok})$, is of the same shape as the visual embedding $\overline{V}$. Following prior art [55], we initialize W_t with LayoutLMv1 [54] pre-trained weights.

Spatial Features: For each word k in the text, we also get bounding box coordinates $b_k = (x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$. 2D spatial coordinates b_k provide additional context to the model about the location of a word in relation to the entire document. This helps the model make better sense of the content. For each word, we encode the top-left and bottom-right coordinates using separate layers W^x and W^y for x and y-coordinates respectively. We also encode more spatial features: bounding box height h, width w, the Euclidean distance from each corner of a bounding box to the corresponding corner in the bounding box to its right and the distance between centroids of the bounding boxes, e.g. $A_{rel} = \{A^{k+1}_{num} - A^{k}_{num}\}; A \in (x, y); num \in (1, 2, 3, 4, c)$, where c is the center of the bounding box. Since transformer layers are permutation-invariant, we also use absolute 1D positional encodings P^{abs}. We create separate spatial embeddings for visual $\overline{V_s}$ and language $\overline{T_s}$ features since spatial dependency could be modality specific. Final spatial embeddings are obtained by summing up all intermediate embeddings. All spatial embeddings are trainable.

$$\overline{V_s} = W^x_v(x_1, x_3, w, A^x_{rel}) + W^y_v(y_1, y_3, h, A^y_{rel}) + P^{abs}_v \quad (1)$$

$$\overline{T_s} = W^x_t(x_1, x_3, w, A^x_{rel}) + W^y_t(y_1, y_3, h, A^y_{rel}) + P^{abs}_t \quad (2)$$

Multi-Modal Self-Attention Layer: We now describe in detail our novel multi-modal self-attention layer. Consider a transformer encoder $f_{enc}(\eta, \overline{V}, \overline{V_s}, \overline{T}, \overline{T_s})$, where η are trainable parameters of the transformer, $\overline{V}$, $\overline{V_s}$, $\overline{T}$ and $\overline{T_s}$ are visual, visual-spatial, language and language-spatial

features respectively, and are obtained as described previously. Transformer f_{enc} outputs a multi-modal feature representation $\overline{M}$ of the same shape $d = 768, N = 512$ as each of the input features.

Self-attention, i.e., scaled dot-product attention as introduced in [51], for a single head is defined as querying a dictionary with key-value pairs. i.e. in a transformer layer l and i^{th} input token in a feature length of L.

$$\overline{M}_i^l = \sum_{j=1}^{L} \frac{\exp\left(\alpha_{ij}\right)}{\sum_{j'=1}^{n} \exp\left(\alpha_{ij'}\right)} \left(x_j^l W^{V,l}\right) \tag{3}$$

where α_{ij} is defined as self-attention which is computed as (attention in layer l between tokens x_i and x_j).

$$\alpha_{ij} = \frac{1}{\sqrt{d}} \left(x_i^l W^{Q,l}\right) \left(x_j^l W^{K,l}\right)^T \tag{4}$$

Here, d is the dimension of the hidden representation, $W^{Q,l}, W^{K,l} \in \mathbb{R}^{d \times d_K}$, and $W^V \in \mathbb{R}^{d \times d_V}$ are learned parameter matrices which are not shared among layers or attention heads. Without loss of generality, we remove the dependency on layer l and get a simplified view of Eq. 4 as:

$$\alpha_{ij} = \left(x_i W^Q\right) \cdot \left(x_j W^K\right)^T \tag{5}$$

We modify this attention formulation for the multi-modal VDU task. DocFormer tries to infuse the following inductive bias into self-attention formulation: *for most VDU tasks, local features are more important than global ones*. We modify Eq. 5, to add relative features. Specifically, the attention distribution for visual features is:

$$\alpha_{ij}^v = \underbrace{(x_i^v W_v^Q)(x_j^v W_v^K)^T}_{\text{key-query attn.}} + \underbrace{(x_i^v W_v^Q a_{ij})}_{\text{query 1D relative attn.}} + \underbrace{(x_j^v W_v^K a_{ij})}_{\text{key 1D relative attn.}} + \underbrace{(\overline{V}_s W_s^Q)(\overline{V}_s W_s^K)}_{\text{visual spatial attn.}} \tag{6}$$

Here, x^v denotes visual features, W_v^K, W_v^Q denote learnable matrices for key, query visual embeddings respectively. W_s^K, W_s^Q denote learnable matrices for key, query spatial embeddings respectively. a_{ij} is 1D relative position embedding between tokens i, j i.e. $a_{ij} = W_{j-i}^{rel}$ where W^{rel} learns how token i attends to j. We clip the relative attention so DocFormer gives more importance to local features. We get a similar equation for language attention α_{ij}^t:

$$\alpha_{ij}^t = (x_i W_t^Q)(x_j W_t^K)^T + (x_i W_t^Q a_{ij}) + (x_j W_t^K a_{ij}) + (\overline{T}_s W_s^Q)(\overline{T}_s W_s^K) \tag{7}$$

Here, x is the output of the previous encoder layer, or word embedding layer if $l = 1$. An important aspect of Eq. 6 and Eq. 7 is that we share spatial weights in each layer. i.e. the spatial attention weights (W_s^Q, W_s^K) are shared across vision and language. This helps the model correlate features across modalities.

Using the visual self-attention computed using Eq. 6 in Eq. 3, gets us spatially aware, self-attended visual features $\hat{V}_l$. Similarly using Eq. 7 in Eq. 3, gets us language features $\hat{T}_l$. The multi-modal feature output is given by $\overline{M}_l = \hat{V}_l + \hat{T}_l$. It should be noted that for layers $l > 1$, features x in Eq. 7 are multi-modal because we combine visual and language features at the output of layer $l-1$. The final $\overline{M}_{12}$ is consumed by downstream linear layers.

Why do multi-modal attention this way? We untie the visual and spatial information and pass them to each layer of transformer. We posit that making visual and spatial information accessible across layers acts as an information residual connection [22, 52] and is beneficial for generating superior multi-modal feature representation hence better addressing the issue of *cross-modality feature correlation*. This is verified in our experiments (Section 4), where we show that DocFormer obtains state-of-the-art performance even when compared to models having four times the number of the parameters in some cases. Further, sharing spatial weights across modalities in each layer gives DocFormer an opportunity to learn cross-modal spatial interactions while also reducing the number of parameters. In Sec. 4, we show that DocFormer is the smallest amongst its class of models, yet it is able to show superior performance. Code in supple.

Run-time Complexity: The run-time complexity of DocFormer is of the same order as that of the original self-attention model [51] (for details see supplemental material)

3.2. Pre-training

The ability to design new and effective unsupervised pre-training strategies is still an open problem. Our pre-training process involves passing the document image, its extracted OCR text, and its corresponding spatial features. All pre-training tasks were designed such that the network needs the collaboration of both visual and language features, thereby truly learning a superior representation than training with only one of the modalities. See Figure 3 for a high-level overview of the pre-training tasks.

Multi-Modal Masked Language Modeling (MM-MLM): This is a modification of the original masked language modeling (MLM) pre-text task introduced in BERT [14], and may be thought of as a text de-noising task i.e. for a text sequence t, a corrupted sequence is generated $\tilde{t}$. The transformer encoder predicts $\hat{t}$ and is trained with an objective to reconstruct entire sequence. In our case, we use a multi-modal feature embedding $\overline{M}$ for reconstruction of the text sequence. In prior art [55, 54], for a masked text token, the corresponding visual region was also masked to prevent "cheating". Instead, we intentionally do not mask visual regions corresponding to $[MASK]$ text. This is to encourage visual features to supplement text features and thus minimize the text reconstruction loss. The masking percentage is the same as originally proposed [14]. Cross-entropy loss

is used for this task (L_{MM-MLM}).

Learn To Reconstruct (LTR): In this novel pre-text task, we do the image version of the MM-MLM task, i.e. we do an image reconstruction task. The multi-modal feature predicted by DocFormer is passed through a shallow decoder to reconstruct the image (the same dimension as the input image). In this case this task is similar to an auto-encoder image reconstruction but with multi-modal features. The intuition is that in the presence of both image and text features, the image reconstruction would need the collaboration of both modalities. We employ a smooth-L1 loss between the reconstructed image and original input image (L_{LTR}).

Text Describes Image (TDI): In this task, we try to teach the network if a given piece of text describes a document image. For this, we pool the multi-modal features using a linear layer to predict a binary answer. This task differs from the above two tasks in that this task infuses the global pooled features into the network (as opposed to MM-MLM and LTR focusing purely on local features). In a batch, 80% of the time the correct text and image are paired, for the remaining 20% the wrong image is paired with the text. A binary cross-entropy loss (L_{TDI}) is used for this task. Since the 20% negative pair scenario interferes with the LTR task (for a text $\leftrightarrow$ image pair mismatch the pair reconstruction loss would be high), the LTR loss is ignored for cases where there is a mismatch.

The final pre-training loss $L_{pt} = \lambda L_{MM-MLM} + \beta L_{LTR} + \gamma L_{TDI}$. In practice $\lambda = 5, \beta = 1$ and $\gamma = 5$. DocFormer is pre-trained for 5 epochs, then we remove all three task heads. We add one linear projection head and fine-tune all components of the model for all downstream tasks.

4. Experiments

For all experiments, we fine-tune on the training set and report numbers on the corresponding test/validation dataset. No dataset specific hyper-parameter tuning was done. We treat this as a plus and our reported numbers could be higher if dataset specific fine-tuning was done. For all downstream tasks, we use the official provided annotations unless otherwise stated. A common theme amongst these datasets is the relatively small amount of training data (most with $<$1000 samples). We posit that pre-training is much more helpful in such scenarios and helps measure the generalization capability of DocFormer .

Notations: Tables 1, 2, 3, 4, use the following notation. T: Text features, S: spatial features. I: image features. **Bold** indicates SOTA. Underline indicates second best. † denotes the use of Encoder + Decoder transformer layers. $*$ signifies approximate estimation.

Implementation details: We summarize details for pre-training and fine-tuning in Table 1 in supplemental. We emphasize the importance of warm-up steps and learning rate scale. We found that these settings have a non-trivial impact on pre-training result as well as downstream task performance. We used Pytorch [38] and the Huggingface library [50].

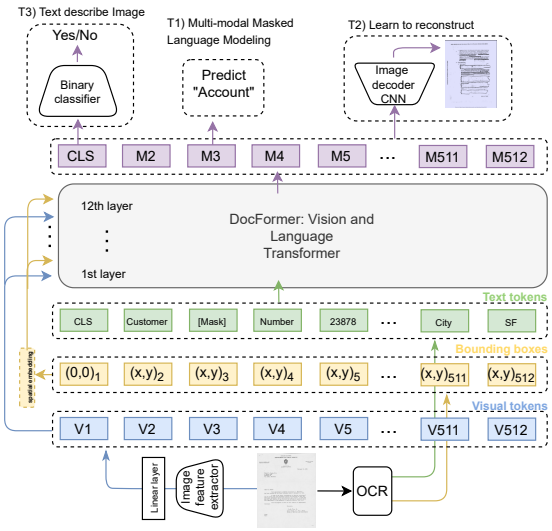


Figure 3: DocFormer **pre-training methodology**. High level overview. Note: First bounding box token corresponding to [CLS], is meant for entire page coordinates.

Models: We employ the commonly used terminology for transformer encoder models - *base* with 12 transformer layers (768 hidden state and 12 attention heads) and large with 24 transformer layers (1024 hidden state and 16 attention heads). We show that DocFormer -base gets SOTA for three of the 4 tasks beating even large models and for the 4th task is close to a large model. In addition to the multi-modal DocFormer , we also present a text and spatial DocFormer by pre-training DocFormer multi-modally but fine-tuning with only text and spatial features. We do this to show the flexibility of our model and show that during pre-training visual features were infused into DocFormer leading it to do better than pure text and spatial models.

4.1. Sequence Labeling Task

FUNSD [17] dataset is a form understanding task. It contains 199 noisy documents (149 train, 50 test) which are scanned and annotated. We focus on the semantic entity-labeling task (i.e., group tokens which belong to the same class). We measure entity-level performance using F1 score shown in Table 1. DocFormer -base achieves 83.34% F1 score which is better than comparable models: LayoutLMv2-base (+0.58), BROS (+2.13), LayoutLMv1-base (+4.07). Story repeats for DocFormer -large inspite of it trained only with 5M pages.

FUNSD performance vs Pre-training samples: We also measure the performance of DocFormer -base with in-

Model	#param (M)	Precision	Recall	F1
methods based on only text / (text + spatial) features:				
BERT-base [14]	109	54.69	61.71	60.26
RoBERTa-base [34]	125	63.49	69.75	66.48
UniLMv2-base [3]	125	63.49	69.75	66.48
LayoutLMv1-base [54]	113	76.12	81.55	78.66
BROS-base [25]	139	80.56	81.88	81.21
BERT-large [14]	340	61.13	70.85	65.63
RoBERTa-large [34]	355	67.80	73.91	70.72
UniLMv2-large [3]	355	67.80	73.91	70.72
LayoutLMv1-large [54]	343	75.36	80.61	77.89
methods based on image + text + spatial features:				
LayoutLMv1-base [54]	160	76.77	81.95	79.27
LayoutLMv2-base [55]	200	80.29	85.39	82.76
LayoutLMv2-large [55]	426	83.24	85.19	84.20
DocFormer-base (T+S)	149	77.63	83.69	80.54
DocFormer-base (I+T+S)	183	80.76	86.09	83.34
DocFormer-large (T+S)	536	81.33	85.44	83.33
DocFormer-large (I+T+S)	536	82.29	86.94	**84.55**

Table 1: **FUNSD comparison**: DocFormer does better than models its size and compares well with even larger models

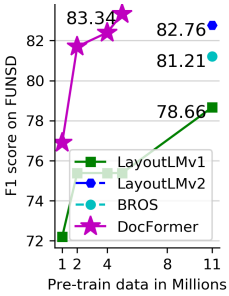


Figure 4: **Amount of Pre-training matters**: x-axis is the number of pre-training samples needed. y-axis is the F1-score on FUNSD task. DocFormer -base gets 83.34 after pre-training on only 5M pages and outperforms current SOTA LayoutLMv2-base's 82.76 which was pretrained on more than 2x more data.

creasing number of pre-training samples. As seen in Figure 4, our base model achieves state-of-the-art performance of 83.34 F1-score in-spite of being pre-trained with only 5M documents. Previous SOTA needed more than 2x pre-training documents (11M) to achieve (82.76). Also DocFormer converges faster.

DocFormer performance without images: Please note DocFormer -base T+S model which was pre-trained with (I+T+S) but was fine-tuned on FUNSD without Images gives F1 of 80.54 which is +1.88% higher than a LayoutLMv1 (78.66%) which was purely pre-trained and fine-tuned on T+S. We hypothesize that DocFormer was infused with visual features during pre-training and is better than text-only pre-trained models.

4.2. Document Classification Task

For this task we use pooled features to predict a classification label for a document. The RVL-CDIP [19] dataset consists of 400,000 grayscale images in 16 classes, with 25,000 images per class. Overall there are 320,000 training images, 40,000 validation images, and 40,000 test images. We report performance on test and eval metric is the overall classification accuracy. In line with prior art [55, 25] text and layout information is extracted using Textract OCR.

DocFormer -base achieves state-of-the-art performance of 96.17%. DocFormer gives superior performance to all existing base and large transformer variants. Some models greater than 4x in number of parameters (TILT-large, 780M parameters gives 94.02% (-2.15% gap).

Model	#param (M)	Accuracy (%)
methods based on only images:		
CNN ensemble [19]	*60	89.80
VGG-16 [1]	138	88.33
AlexNet [49]	61	90.94
GoogLeNet [9]	13	90.70
Single Vision model [10]	*140	91.11
Ensemble [10]	-	92.21
InceptionResNetV2 [47]	56	92.63
LadderNet [43]	-	92.77
methods based on text / (text + spatial) features:		
BERT-base [14]	110	89.81
UniLMv2-base [3]	125	90.06
LayoutLMv1-base [54]	113	91.78
BROS-base † [25]	139	95.58
BERT-large [14]	340	89.92
UniLMv2-large [3]	355	90.20
LayoutLMv1-large [54]	343	91.90
methods based on image + text + spatial features:		
Single Modal [11]	-	93.03
Ensemble [11]	-	93.07
TILT-base † [39]	230	93.50
LayoutLMv1-base [54]	160	94.42
LayoutLMv2-base [55]	200	95.25
LayoutLMv1-large [54]	390	94.43
TILT-large † [39]	780	94.02
LayoutLMv2-large [55]	426	95.65
DocFormer-base (I+T+S)	183	**96.17**
DocFormer-large (I+T+S)	536	95.50

Table 2: **RVL-CDIP dataset** [19] **comparison**: We report classification accuracy on the test set. DocFormer gets the highest classification accuracy and outperforms TILT-large by +2.15 which is almost 4x its size.

4.3. Entity Extraction Task

We report performance on two different entity extraction datasets:

CORD Dataset [45]: consists of receipts. It defines 30 fields under 4 categories. The task is to label each word to the right field. The evaluation metric is entity-level F1. We use the provided OCR annotations and bounding boxes for fine-tuning (Table 3). DocFormer -base achieves 96.33% F1 on this dataset besting all prior *-base and virtually all *-large variants tying with TILT-large [39] which has higher number of parameters. DocFormer -large achieves 96.99% besting all other *-large variants achieving SOTA.

Kleister-NDA [16]: dataset consists of legal NDA documents. The task with Kleister-NDA data is to extract the values of four fixed labels. The approach needs to learn to ignore unrelated text. This dataset is challenging since it

Model	#param (M)	Precision	Recall	F1
methods based on only text / (text + spatial) features:				
BERT-base [14]	109	88.33	91.07	89.68
UniLMv2-base [3]	125	89.87	91.98	90.92
SPADE [28]	-	-	-	91.50
LayoutLMv1-base [54]	113	94.37	95.08	94.72
BROS-base † [25]	139	95.58	95.14	95.36
BERT-large [14]	340	88.86	91.68	90.25
UniLMv2-large [3]	355	91.23	92.89	92.05
LayoutLMv1-large [54]	343	94.32	95.54	94.93
methods based on image + text + spatial features:				
LayoutLMv2-base [55]	200	94.53	95.39	94.95
TILT-base † [39]	230	-	-	95.11
LayoutLMv2-large [55]	426	95.65	96.37	96.01
TILT-large † [39]	780	-	-	96.33
DocFormer-base (T+S)	149	94.82	95.07	94.95
DocFormer-base (I+T+S)	183	96.52	96.14	96.33
DocFormer-large (T+S)	502	96.46	96.14	96.30
DocFormer-large (I+T+S)	536	97.25	96.74	**96.99**

Table 3: **CORD dataset** [45] **comparison**. We present entity-level Precision, Recall, F1 on test set.

has some "decoy" text, for which no label should be given. Also, there might be more than one value given for a given label and all values need to be extracted. In line with prior-art we measure F1-score on validation data (since ground truth is not provided for test data). Also we extract OCR and apply heuristics to create train/validation ground-truth on OCR (Table 4).

Model	#param (M)	F1
methods based on only text / (text + spatial) features:		
LAMBERT [15]	-	75.7
RoBERTa [34]	125	76.7
BERT-base [14]	110	77.9
UniLMv2-base [3]	125	79.5
LayoutLMv1-base [54]	113	82.7
BERT-large [14]	340	79.1
UniLMv2-large [3]	355	81.8
LayoutLMv1-large [54]	343	83.4
methods based on image + text + spatial features:		
LayoutLMv2-base [55]	200	83.3
LayoutLMv2-large [55]	426	85.2
DocFormer -base (T+S)	149	82.1
DocFormer -base (I+T+S)	183	**85.8**

Table 4: **Kleister-NDA dataset** [16] **comparison**: We present entity-level Precision, Recall, F1 on validation set. DocFormer gives the best performance, out-performing other *-large models trained with 2.5x the learning capacity.

4.4. More Experiments

We conduct further analysis on the behavior of DocFormer pertaining to pre-training tasks, network structure and spatial embedding weight sharing.

Shared or Independent Spatial embeddings? One of the benefits of our proposed DocFormer multi-modal self-attention architecture (Fig. 2 and Eq. 6,7) is that sharing spatial embeddings across vision and language makes it easier for the model to learn feature-correlation across modalities. We see ablation on this aspect in Table 5.

Configuration	Num Params	FUNSD (F1)	CORD (F1)
w. shared spatial Eq. 6,7	183 M	76.9	93.36
w/o shared spatial	198 M	75.58 (-1.32)	92.51 (-0.85)

Table 5: **Spatial Weight Sharing**: In w/o shared spatial, vision and language get their own spatial weights W_s.

Do our pre-training tasks help? Pretraining is essential for *low-to-medium* data regimes (FUNSD and CORD). but even for downstream tasks with a lot of training samples (RVL-CDIP) it helps to improve performance and convergence (Table 6).

Dataset	Train samples	with pre-train then 100 epochs (F1)	w/o pre-train 100 epochs (F1)
FUNSD [17]	149	83.34	4.18
CORD [45]	800	96.33	0.54
RVL-CDIP [19]	320,000	96.17	93.95

Table 6: **Effect of Pre-training**

Does a deeper projection head help? So far we used a single linear layer for downstream evaluation as is common practice [20, 7, 5, 6, 2] to compare against prior art. Recent publications [6, 2] in self-supervision show that a deeper projection head with ReLU activation acts as a one-way filter to enrich the representation space. We adapt this practice and see if a deeper projection head (fc → ReLU → LayerNorm → fc) can improve downstream performance. Table 7 shows that in the *low-to-medium* data regime adding a more powerful projection head is harmful and could lead to over-fitting. For the *medium-to-large* downstream task data regime, adding a deeper projection head is beneficial.

Dataset	Train samples	Linear head (F1)	Deeper head (F1)
FUNSD [17]	149	83.34	82.93 (-0.41)
CORD [45]	800	96.33	96.87 (+0.54)
RVL-CDIP [19]	320,000	96.17	96.85 (+0.68)

Table 7: **Deeper Projection Head**

4.5. Ablation Study

Since it takes a long time to pre-train on the entire 5M pages and to minimize environmental impact [23], we do all ablation experiments in Table 8 and 9 by pre-training with only 1M documents for 5 epochs. In both Table 8 and 9, we show performance in addition to the previous row in the table. Impact due to adding that component is shown in brackets. We can see in Table 8 that each of our pre-training tasks have something to contribute to the downstream task

(a) Ground Truth (b) Text + Spatial model [54] (c) **DocFormer** multi-modal

Figure 5: **DocFormer Qualitative Examples**: From DocFormer on FUNSD test-set DocFormer 83.34 F1 vs LayoutLMv1 78.66 F1. **Legend**: **Red**: **Header**-label, **Blue**: **Question**, **Green**: **Answer**. **Row 1**: "TARGET" is a **Header**-label which is very visual in nature. DocFormer correctly classifies it whereas a text + spatial model misses such visual cues. **Row 2**: This is a challenging scenario. Notice the word "Research" behind the signature. Text + spatial model gets confused and mis-classifies "Research" as **Header**, whereas DocFormer figured out that "Research" is part of "Marketing Research Director" in spite of visual occlusions. **Row 3**: Notice "Approvals" is partially hidden behind DATE. In spite of that DocFormer correctly labelled "APPROVALS" as **Question**, where as text+spatial model incorrectly labels it as **Header**. Best viewed in color and digitally. Snippets are from FUNSD file 86079776_9777, 89856243, and 87125460.

performance. The contribution also seem to vary depending on the nature of the downstream task.

Pre-training task	FUNSD (F1)	CORD (F1)
DocFormer + MLM	72.40	90.58
DocFormer + MM-MLM	73.91 (+1.51)	90.98 (+0.4)
+ Learn to Reconstruct (LTR)	74.68 (+0.77)	92.61 (+1.63)
+ Text describes Image (TDI)	76.90 (+2.23)	93.36 (+0.75)
final (DocFormer)	76.90	93.36

Table 8: **Ablation on pre-training tasks**: We show the impact of various pre-training tasks on two downstream tasks. MLM: masked language modeling [14]. MM-MLM: multi-modal MLM described in Section 3.

DocFormer architecture ablation: In this ablation we look at the impact of various architectural components of DocFormer . Depending on the down-stream task the impact of the proposed multi-modal self-attention varies from 3.89% to 1.08%. This shows that the proposed architecture has indeed learned to fuse multiple modalities.

Model / Component	FUNSD (F1)	CORD (F1)
Text only model (BERT-base)	61.56	89.23
+ spatial features	73.01 (+11.45)	92.28 (+3.05)
+ multi-modal self-attention	76.90 (+3.89)	93.36 (+1.08)
final (DocFormer)	76.90	93.36

Table 9: **Ablation on DocFormer Components**: We show the impact of various architectural components used in DocFormer on two downstream tasks (FUNSD and CORD).

Qualitative Analysis: We share some qualitative examples of the predictions from DocFormer . Figure 5 shows some sequence labeling predictions on the FUNSD dataset. (more examples are in the supplemental).

5. Conclusion

In this work, we present DocFormer , a multi-modal end-to-end trainable transformer based model for various Visual Document Understanding tasks. We presented the novel multi-modal attention and two novel vision-plus-language pre-training tasks that allows DocFormer to learn effectively without labeled supervision. We have shown experimentally that DocFormer indeed learns generalized features through its unsupervised pre-training by matching or surpassing existing state-of-the-art results on 4 datasets that cover a variety of document types. We emphasize that DocFormer showed superior performance against strong baselines in-spite of being one of the smallest model (in terms of # of parameters) in its class.

In the future, we plan to improve DocFormer's generalizability in multi-lingual settings as well as for more document types such as info-graphics, maps, and web-pages.

References

[1] Muhammad Zeshan Afzal, Andreas Kölsch, Sheraz Ahmed, and Marcus Liwicki. Cutting the error by half: Investigation of very deep cnn and advanced training strategies for document image classification. In *2017 14th IAPR Inter-*

national Conference on Document Analysis and Recognition (ICDAR), volume 1, pages 883–888. IEEE, 2017.

[2] Srikar Appalaraju, Yi Zhu, Yusheng Xie, and István Fehérvári. Towards good practices in self-supervised representation learning. *Neural Information Processing Systems (NeurIPS Self-Supervision Workshop 2020)*, 2020.

[3] Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Songhao Piao, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unilmv2: Pseudo-masked language models for unified language model pre-training, 2020.

[4] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *ICML*. 2020.

[6] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big Self-Supervised Models are Strong Semi-Supervised Learners. *arXiv preprint arXiv:2006.10029*, 2020.

[7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved Baselines with Momentum Contrastive Learning. *arXiv preprint arXiv:2003.04297*, 2020.

[8] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European Conference on Computer Vision*, pages 104–120. Springer, 2020.

[9] Gabriela Csurka, Diane Larlus, Albert Gordo, and Jon Almazan. What is the right way to represent document images? *arXiv preprint arXiv:1603.01076*, 2016.

[10] Arindam Das, Saikat Roy, Ujjwal Bhattacharya, and Swapan K Parui. Document image classification with intra-domain transfer learning and stacked generalization of deep convolutional neural networks. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3180–3185. IEEE, 2018.

[11] Tyler Dauphinee, Nikunj Patel, and Mohammad Rashidi. Modular multimodal architecture for document classification. *arXiv preprint arXiv:1912.04376*, 2019.

[12] Brian Davis, Bryan Morse, Scott Cohen, Brian Price, and Chris Tensmeyer. Deep visual template-free form parsing. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 134–141. IEEE, 2019.

[13] Timo I Denk and Christian Reisswig. Bertgrid: Contextualized embedding for 2d document representation and understanding. *arXiv preprint arXiv:1909.04948*, 2019.

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[15] Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, and Filip Graliński. Lambert: Layout-aware language modeling using bert for information extraction. *arXiv preprint arXiv:2002.08087*, 2020.

[16] Filip Graliński, Tomasz Stanisławek, Anna Wróblewska, Dawid Lipiński, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemysław Biecek. Kleister: A novel task for information extraction involving long documents with complex layout. *arXiv preprint arXiv:2003.02356*, 2020.

[17] Jean-Philippe Thiran Guillaume Jaume, Hazim Kemal Ekenel. Funsd: A dataset for form understanding in noisy scanned documents. In *Accepted to ICDAR-OST*, 2019.

[18] Maroua Hammami, Pierre Héroux, Sébastien Adam, and Vincent Poulain d'Andecy. One-shot field spotting on colored forms using subgraph isomorphism. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 586–590. IEEE, 2015.

[19] 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)*.

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *CVPR*. 2020.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[23] P. Henderson, Jie-Ru Hu, Joshua Romoff, Emma Brunskill, Dan Jurafsky, and Joelle Pineau. Towards the systematic reporting of the energy and carbon footprints of machine learning. *ArXiv*, abs/2002.05651, 2020.

[24] Jonathan Herzig, Paweł Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Martin Eisenschlos. Tapas: Weakly supervised table parsing via pre-training. *arXiv preprint arXiv:2004.02349*, 2020.

[25] Teakgyu Hong, DongHyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. Bros: A pre-trained language model for understanding texts in document. *under review https://openreview.net/references/pdf?id=uCz3OR6CJT*, 2020.

[26] Teakgyu Hong, DongHyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. Bros: A pre-trained language model for understanding texts in document. *https://openreview.net/forum?id=punMXQEsPr0*, 2020.

[27] 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.

[28] Wonseok Hwang, Jinyeong Yim, Seunghyun Park, Sohee Yang, and Minjoon Seo. Spatial dependency parsing for semi-structured document information extraction, 2020.

[29] Anoop Raveendra Katti, Christian Reisswig, Cordula Guder, Sebastian Brarda, Steffen Bickel, Johannes Höhne, and

Jean Baptiste Faddoul. Chargrid: Towards understanding 2d documents. *arXiv preprint arXiv:1809.08799*, 2018.

[30] Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, and Davide Testuggine. Supervised multimodal bitransformers for classifying images and text. *arXiv preprint arXiv:1909.02950*, 2019.

[31] David Lewis, Gady Agam, Shlomo Argamon, Ophir Frieder, D Grossman, and Jefferson Heard. 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, 2006.

[32] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019.

[33] Ron Litman, Oron Anschel, Shahar Tsiper, Roee Litman, Shai Mazor, and R Manmatha. Scatter: selective context attentional scene text recognizer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11962–11972, 2020.

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

[35] Devashish Lohani, Abdel Belaïd, and Yolande Belaïd. An invoice reading system using a graph convolutional network. In *Asian Conference on Computer Vision*, pages 144–158. Springer, 2018.

[36] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019.

[37] Bodhisattwa Prasad Majumder, Navneet Potti, Sandeep Tata, James Bradley Wendt, Qi Zhao, and Marc Najork. Representation learning for information extraction from form-like documents. In *proceedings of the 58th annual meeting of the Association for Computational Linguistics*, pages 6495–6504, 2020.

[38] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

[39] Rafał Powalski, Łukasz Borchmann, Dawid Jurkiewicz, Tomasz Dwojak, Michał Pietruszka, and Gabriela Pałka. Going full-tilt boogie on document understanding with text-image-layout transformer. *arXiv preprint arXiv:2102.09550*, 2021.

[40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

[41] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

[42] Pau Riba, Anjan Dutta, Lutz Goldmann, Alicia Fornés, Oriol Ramos, and Josep Lladós. Table detection in invoice documents by graph neural networks. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 122–127. IEEE, 2019.

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

[44] Sebastian Schreiber, Stefan Agne, Ivo Wolf, Andreas Dengel, and Sheraz Ahmed. Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, volume 1, pages 1162–1167. IEEE, 2017.

[45] Park Seunghyun, Shin Seung, Lee Bado, Lee Junyeop, Surh Jaeheung, Seo Minjoon, and Lee Hwalsuk. Cord: A consolidated receipt dataset for post-ocr parsing. 2019.

[46] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019.

[47] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[48] Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. Visualmrc: Machine reading comprehension on document images. *arXiv preprint arXiv:2101.11272*, 2021.

[49] Chris Tensmeyer and Tony Martinez. Analysis of convolutional neural networks for document image classification. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 388–393. IEEE, 2017.

[50] Wolf Thomas, Debut Lysandre, Sanh Victor, Chaumond Julien, Delangue Clement, Moi Anthony, Cistac Pierric, Rault Tim, Louf Rémi, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[52] Andreas Veit, Michael Wilber, and Serge Belongie. Residual networks behave like ensembles of relatively shallow networks. *arXiv preprint arXiv:1605.06431*, 2016.

[53] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick,

Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.

[54] 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.

[55] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. *arXiv preprint arXiv:2012.14740*, 2020.

[56] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. Publaynet: largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022. IEEE, 2019.

[57] Jun Zhou, Han Yu, Cheng Xie, Hongming Cai, and Lihong Jiang. irmp: From printed forms to relational data model. In *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 1394–1401. IEEE, 2016.