Visually Guided Generative Text-Layout Pre-training for Document Intelligence

Zhiming Mao[1,2], Haoli Bai[3*], Lu Hou[3], Jiansheng Wei[3], Xin Jiang[3], Qun Liu[3], Kam-Fai Wong[1,2]

[1]The Chinese University of Hong Kong, Hong Kong, China

[2]MoE Key Laboratory of High Confidence Software Technologies, China

[3]Noah's Ark Lab, Huawei Technologies

{zmmao, kfwong}@se.cuhk.edu.hk

{baihaoli, houlu3, weijiansheng, jiang.xin, qun.liu}@huawei.com

Abstract

Prior study shows that pre-training techniques can boost the performance of visual document understanding (VDU), which typically requires models to gain abilities to perceive and reason both document texts and layouts (e.g., locations of texts and table-cells). To this end, we propose visually guided generative text-layout pre-training, named ViTLP. Given a document image, the model optimizes hierarchical language and layout modeling objectives to generate the interleaved text and layout sequence. In addition, to address the limitation of processing long documents by Transformers, we introduce a straightforward yet effective multi-segment generative pre-training scheme, facilitating ViTLP to process word-intensive documents of any length. ViTLP can function as a native OCR model to localize and recognize texts of document images. Besides, ViTLP can be effectively applied to various downstream VDU tasks. Extensive experiments show that ViTLP achieves competitive performance over existing baselines on benchmark VDU tasks, including information extraction, document classification, and document question answering[1].

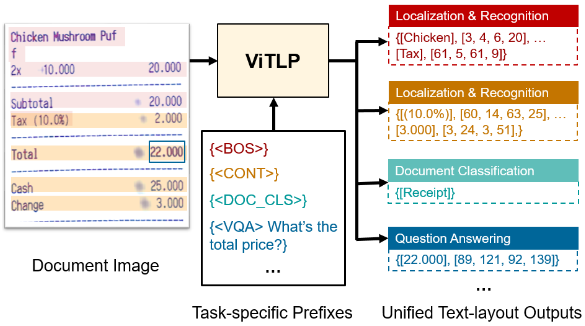


Figure 1: An overview workflow of the proposed ViTLP. Given a document image as input, ViTLP can generate sequences of text and layout (i.e., word bounding boxes) for various VDU tasks with task-specific prefixes.

1 Introduction

Processing and reasoning document images with dense texts (e.g., scanned PDF files, digital forms, and spreadsheets) is a persistent yet challenging task for the research community and industry (Katti et al., 2018; Majumder et al., 2020; Li et al., 2021a). Advances in multimodal pre-training substantially improve the performance of visual document understanding (VDU) (Xu et al., 2020, 2021; Gu et al., 2021; Appalaraju et al., 2021; Wang et al., 2022a). These pre-training methods typically take multimodal inputs of given document images including i) visual features, ii) pre-processed OCR texts, and iii) spatial layouts of document elements (e.g., 2D coordinates of texts and table-cells). Among these inputs, spatial layout information plays an essential role in connecting visual and textual features, as well as developing thorough reasoning of document structures (Chen et al., 2021; Lee et al., 2022).

Though effective, the performance of most existing VDU approaches relies heavily on the OCR pipelines, because the pre-processed OCR texts and corresponding 2D coordinates are used as intermediate inputs to pre-trained VDU models. The external OCR pipelines may produce incorrect or incomplete recognition results, which cannot be jointly optimized by the gradient back from VDU models. Another research line (Kim et al., 2022; Lee et al., 2023b) explores pre-training VDU models solely based on image inputs. Despite no OCR errors introduced, these methods focus on understanding texts from raw document images but neglect layout information modeling. Since the spatial information contained in layout locations is not exploited, it may hinder the models from understanding complex document structures, especially for documents containing nested paragraphs, forms, and tables.

In this work, we propose **V**isually guided generative **T**ext-**L**ayout **P**re-training (ViTLP) to jointly model text and layout information from document images. As shown in Figure 1, ViTLP can localize,

[1]*Haoli is the corresponding author. Code and checkpoint will be at https://github.com/Veason-silverbullet/ViTLP.

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 4713–4730
June 16-21, 2024 ©2024 Association for Computational Linguistics

recognize, and understand visual document texts given the input document image and task prefixes. To achieve this goal, ViTLP is pre-trained to generate *unified text-layout sequences* from document images. Since natively generating text and layout tokens in a flattened sequence is *token-inefficient* (see Sec. 2.1), we introduce hierarchical generation modules to achieve both effective and efficient text-layout sequence generation. To the best of our knowledge, ViTLP is the first attempt to learn OCR (i.e., text localization and recognition) and VDU (i.e., document understanding) abilities in a unified generative text-layout pre-training framework.

Besides, ViTLP is designed to handle long documents with intensive texts. Long document processing is ubiquitous in real-world scenarios. However, existing pre-trained models are constrained to certain token limits of input sequences. For instance, LayoutLMv2 (Xu et al., 2021) accepts the maximum inputs of 512 word tokens using a BERT-structure encoder. In both pre-training and fine-tuning, the exceeded text tokens are truncated, leading to incomplete document information modeling. To tackle this issue, we introduce a **multi-segment pre-training scheme** which divides the target text-layout sequence into consecutive segments to perform generative pre-training. Given that the full document information is already encoded in visual representations, ViTLP takes the suffix tokens from previous segments as prefix prompts to generate the next-segment tokens. This multi-segment pre-training scheme further enables ViTLP to process documents of arbitrary length in fine-tuning. Notably, our multi-segment generation scheme retains the intact transformer architecture. Thus, it is more feasible than other long-document modeling workarounds, e.g., *sparse attention* (Beltagy et al., 2020) and *memory modules* (Bulatov et al., 2022), which need to modify the Transformer architecture and may affect the capacity of pre-trained models.

We evaluate ViTLP on a variety of OCR and VDU tasks. Experiment results demonstrate that ViTLP can achieve superior overall performance on both OCR and VDU tasks. For instance, ViTLP achieves the 95.59% F1 score on CORD information extraction and 95.36% accuracy on RVL-CDIP document classification, both of which outperform most previous approaches. Notably, ViTLP can intrinsically generate 2D layout locations for visual grounding, which helps in certain generative VDU tasks (e.g., visual document question answering) to be more interpretable and reliable to humans.

2 Approach

2.1 Problem Formulation

We study multimodal pre-training for visual document modeling. As widely studied (Xu et al., 2020, 2021; Appalaraju et al., 2021; Li et al., 2021b; Powalski et al., 2021; Wang et al., 2022a; Huang et al., 2022; Wang et al., 2022b), document images $\mathbf{V}$, texts $\mathbf{T}$, and layouts $\mathbf{L}$ are three fundamental modalities for visual document modeling.

Unified Text-Layout Generation. We cast the pre-training objective on visual documents as text-layout sequence (i.e., $\{\mathbf{T};\mathbf{L}\}$) generation conditioned on document images $\mathbf{V}$. The document texts $\mathbf{T}$ are represented as word-token sequences. The layouts $\mathbf{L}$, following prior studies (Xu et al., 2020, 2021), can be represented by *location bounding boxes* of words. Instead of generating two separate sequences of $\mathbf{T}$ and $\mathbf{L}$, ViTLP generates the texts with corresponding layout locations in a sequence of interleaved text-layout tokens, which facilitates compact multimodal interaction between texts and layouts. For the i-th word of a document, its text-layout tokens $\{\mathbf{T};\mathbf{L}\}_i$ are represented as

$$\{\mathbf{T};\mathbf{L}\}_i = \big\{\{\boldsymbol{w}\}_i, \{z_{x_1}, z_{y_1}, z_{x_2}, z_{y_2}\}_i\big\}, \quad (1)$$

where $\{\boldsymbol{w}\}_i$ denotes the BPE tokens (Radford et al., 2019) of the i-th word, $\{z_{x_1}, z_{y_1}, z_{x_2}, z_{y_2}\}_i \in \mathbb{Z}_+^4$ are the corresponding left-top and right-bottom bounding box coordinates. Given a document with N words, the objective is to maximize the likelihood function $\log p(\mathbf{T};\mathbf{L}|\mathbf{V})$ which can be decomposed as autoregressive text and layout modeling:

$$\log p(\mathbf{T};\mathbf{L}|\mathbf{V}) = \sum_{i=1}^{N} \Big(\underbrace{\log p(\mathbf{T}_i|\mathbf{T}_{<i}, \mathbf{L}_{<i}, \mathbf{V})}_{\text{Text-modeling}} + \underbrace{\log p(\mathbf{L}_i|\mathbf{T}_{\leq i}, \mathbf{L}_{<i}, \mathbf{V})}_{\text{Layout-modeling}} \Big). \quad (2)$$

Note that Eq. (2) shares similar ideas with Chen et al. (2022), where word and bounding box generation can be formulated as language modeling on a unified text-layout sequence. However, it is in fact nontrivial to generate sequences as in Eq. (1), because real-world documents commonly contain intensive texts, generating each word followed by four coordinate tokens in a long flattened sequence is especially **token-inefficient**. This would bring prohibitive computational and space overhead[2] to the Transformer-based text-layout decoder.

[2]Recall that both the computational and space complexities of Transformers are quadratic $\mathcal{O}(L^2)$ in sequence length L.

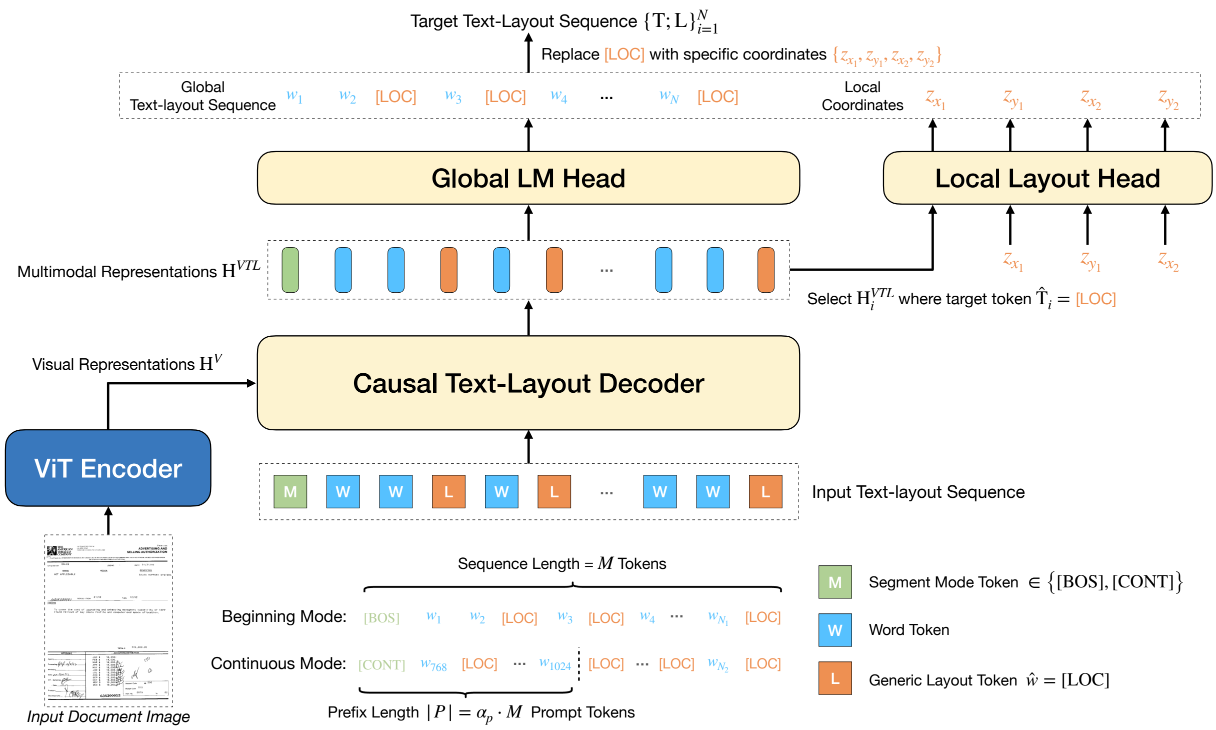


Figure 2: Overview of the ViTLP architecture. ViTLP is a generative pre-training model that performs autoregressive text-layout modeling conditioned on visual document inputs. ViTLP adopts hierarchical decoder heads to generate target text-layout sequences in a *global-to-local* manner. The segment mode tokens ∈ { [BOS], [CONT] } prompt the beginning and continuous modes of generation, respectively.

2.2 Model Architecture

The architecture of ViTLP is shown in Figure 2. ViTLP employs an encoder-decoder framework to encode document images $\mathbf{V}$ and generate target text-layout sequences $\{\mathbf{T}; \mathbf{L}\}$. Specifically, given an input document image $\mathbf{V}$, ViTLP employs a vision transformer (ViT) (Dosovitskiy et al., 2021) to learn visual representations $\mathbf{H}^V \in \mathbb{R}^{|V| \times d}$, where $|V|$ is the ViT patch number and d is the hidden size. The decoder receives the visual representations $\mathbf{H}^V$ and generates the unified text-layout sequence $\{\mathbf{T}; \mathbf{L}\}$. To address the *token-inefficiency* issue discussed in Sec. 2.1, we design the *global-to-local* text-layout generation process as follows.

2.2.1 Global Text-Layout Modeling

Instead of directly generating the text-layout sequence as in Eq. (1), we first replace the bounding box coordinates $\{z_{x_1}, z_{y_1}, z_{x_2}, z_{y_2}\}$ with a generic layout location token $\hat{w} = \texttt{[LOC]}$. This integrates the mixed text-layout sequence $\{\mathbf{T}; \mathbf{L}\}$ to unified language modeling. Given the original vocabulary $\mathcal{V}$, the **global text-layout sequence** $\hat{\mathbf{T}}$ derives from the augmented vocabulary $\hat{\mathcal{V}} = \mathcal{V} \cup \texttt{[LOC]}$. The layout token embeddings $\mathrm{E}_{\texttt{[LOC]}}$ are computed as

$$\mathrm{E}_{\texttt{[LOC]}} = \left[\mathrm{E}_x(z_{x_1}), \mathrm{E}_y(z_{y_1}), \mathrm{E}_x(z_{x_2}), \mathrm{E}_y(z_{y_2})\right],$$

where $\mathrm{E}_x(\cdot) \in \mathbb{R}^{\frac{d}{4}}$ and $\mathrm{E}_y(\cdot) \in \mathbb{R}^{\frac{d}{4}}$ denote the x- and y-axis spatial embeddings. Besides, the word tokens are embedded by $\mathrm{E}_w(\cdot) \in \mathbb{R}^d$. Given a document of N words and the corresponding bounding boxes, the text-layout input embeddings are represented as $\mathbf{H}^{TL} = \{\mathrm{E}_w, \mathrm{E}_{\texttt{[LOC]}}\} \in \mathbb{R}^{|\hat{\mathbf{T}}| \times d}$.

The ViTLP text-layout decoder performs multimodal interaction among *visual*, *textual*, and *layout* information via the Transformer cross-attention

$$\mathbf{H}^{VTL} = \text{Transformer-Decoder}(\mathbf{H}^V, \mathbf{H}^{TL}).$$

For the i-th target token $\hat{\mathbf{T}}_i$, the multimodal decoder output $\mathbf{H}_i^{VTL}$ is fed to a linear language modeling (LM) head with the softmax function to compute the conditional generative probability

$$p(\hat{\mathbf{T}}_i | \hat{\mathbf{T}}_{<i}, \mathbf{V}) = \text{Softmax}\big(\text{Linear}(\mathbf{H}_i^{VTL})\big).$$

With the generic layout token [LOC] incorporated, the text-modeling term in Eq. (2) is expressed as

$$\mathcal{L}_{\text{global-text}} = -\frac{1}{|\hat{\mathbf{T}}|} \sum_{i=1}^{|\hat{\mathbf{T}}|} \log p(\hat{\mathbf{T}}_i | \hat{\mathbf{T}}_{<i}, \mathbf{V}). \quad (3)$$

2.2.2 Local Layout Modeling

Local layout modeling aims to generate specific layout locations for each generic layout token [LOC].

To capture the spatial relation among coordinates, we employ a lightweight sequential MLP layout head (see details in Appendix B) to decode the short sequence of four layout coordinate tokens from the last hidden state of `[LOC]`. For notation simplicity, we denote $\{\mathbf{L}_{i,j}\}_{j=1}^{4} = \{z_{x_1}, z_{y_1}, z_{x_2}, z_{y_2}\}_i$ as the corresponding layout coordinates of the `[LOC]` token at the i-th position, and its generative probability is modeled as

$$p(\mathbf{L}_{i,j}|\hat{\mathbf{T}}_{\leq i}, \mathbf{L}_{i,<j}, \mathbf{V}) = \text{Softmax}\big(\text{MLP}(\mathbf{H}_{i,<j})\big),$$

where $\mathbf{H}_{i,0} = \mathbf{H}_i^{VTL}$ is selected from the learned multimodal representations where $\hat{\mathbf{T}}_i = \texttt{[LOC]}$. Here, we denote the index set of `[LOC]` tokens as $\mathcal{S}_L = \big\{i : \hat{\mathbf{T}}_i = \texttt{[LOC]} \,|\, i = 1, 2, ..., |\hat{\mathbf{T}}|\big\}$. The layout-modeling term in Eq. (2) is expressed as

$$\begin{aligned} \mathcal{L}_{\text{local-layout}} &= -\sum \log p(\mathbf{L}_i|\hat{\mathbf{T}}_{\leq i}, \mathbf{L}_{<i}, \mathbf{V}) \quad (4) \\ &= -\frac{1}{4|\mathcal{S}_L|}\sum_{i\in\mathcal{S}_L}\sum_{j=1}^{4}\log p(\mathbf{L}_{i,j}|\hat{\mathbf{T}}_{\leq i}, \mathbf{L}_{i,<j}, \mathbf{V}). \end{aligned}$$

In summary, with the global and local text-layout modeling in a hierarchy, the original pre-training objective in Eq. (2) evolves to

$$\mathcal{L} = \mathcal{L}_{\text{global-text}} + \mathcal{L}_{\text{local-layout}}. \quad (5)$$

The *global-to-local* generation process aims to be effective and efficient for text-layout modeling. On effectiveness, the interleaved text-layout sequence modeling enables compact interaction between text and layout inputs, which can effectively fuse the information of text and layout modalities. On efficiency, suppose that the average BPE tokens of a document word are $|w|$, and the *compression ratio* of the text-layout sequence is $\frac{|w|+1}{|w|+4}$, i.e., four coordinate tokens are compressed to one. In our experiment datasets, the *compression ratio* is 0.48.

2.3 Multi-segment Pre-training Scheme

Documents are usually intensive in text and layout, and it would be computationally intractable to fit the entire sequence into a generative model. To process documents with arbitrary length, we propose a multi-segment pre-training scheme that divides the long sequence into multiple segments for generation. Since a document image already contains all necessary information of the text and layout, long document modeling is feasible based on the *visual representations* and *localized generation-context*.

Given the maximum sequence length of the decoder as M, we first divide the text-layout sequence into K segmented sequences $\{\mathbf{S}_i\}_{i=1}^{K}$. The beginning segment $\mathbf{S}_1$ contains M tokens to be generated, and the continuous segment $\mathbf{S}_{i>1}$ contains $\alpha_p \cdot M$ prefix tokens and $(1-\alpha_p)\cdot M$ tokens to be generated. Here, α_p is the pre-defined prefix ratio. The overall generation process comprises beginning and continuous modes.

Beginning Generation Mode. In this mode, we prepend a special mode token `[BOS]` to the beginning sequence $\mathbf{S}_1$. The model then follows the objective in Eq. (5) to generate the first M tokens.

Continuous Generation Mode. For the continuous segments $\mathbf{S}_{i>1}$, we prepend a special mode token `[CONT]` to the input sequence. $|P| = \alpha_p \cdot M$ prefix tokens are prepended to the input sequence. These $|P|$ **prefix tokens** of segmented sequence $\mathbf{S}_i$ come from the $|P|$ **suffix tokens** of the previous segmented sequence $\mathbf{S}_{i-1}$. The prefix tokens serve as a prompt of *localized generation-context*[3] which guides the decoder to generate subsequent tokens from arbitrary locations of a document. The special token `[EOS]` is appended to the last segmented sequence $\mathbf{S}_K$ to signal the end of generation.

Segmentation in Pre-training and Fine-tuning. In pre-training, the segmented sequences of a long document are randomly scattered into different data batches. In this way, ViTLP learns to model the complete textual and layout information of a document, conditioned on different prefix history-token contexts. In fine-tuning (and inference), ViTLP can also apply the multi-segment scheme to process those long text-layout sequences, which is consistent with the pre-training phase. For instance, OCR and sequence labeling on long document texts can be processed segment by segment.

2.4 Applications of ViTLP

2.4.1 OCR Text Localization and Recognition

Text localization and recognition are two fundamental functions of OCR engines (Li et al., 2023). As ViTLP is pre-trained to generate text and layout (i.e., 2D bounding boxes) sequences from document images, it can intrinsically perform text localization and recognition by generating a unified OCR sequence of texts and bounding boxes. ViTLP can function as a word-level OCR model.

[3]The historical context contains the generated coordinate tokens from the previous segment, which serves as an informatively complete prompting signal for next-segment generation.

Approach	OCR Tasks		VDU Tasks			
	Text Local.	Text Recog.	Info. Extraction	Doc. Classification	Document VQA	VQA Grounding
OCR Pipelines	✓	✓				
Discriminative VDU Models			✓	✓	✓	
Generative VDU Models			✓	✓	✓	
ViTLP	✓	✓	✓	✓	✓	✓

Table 1: The comprehensive capabilities of ViTLP and its comparison with the associated baselines on each task.

2.4.2 Downstream VDU Tasks

Information Extraction. The information extraction task is formulated as sequence labeling on the target texts given document image input. Following BART (Lewis et al., 2020), we feed ViTLP decoder's final hidden states of a target word (with layout coordinate inputs) to a linear classifier which outputs the token-level semantic label.

Document Classification. Given an input document image to the encoder, we feed a task prefix token `[DOC_CLS]` as input to the decoder to output the document classification label.

Document Visual Question Answering. Unlike discriminative VDU models that perform extractive QA on pre-processed OCR results, ViTLP directly generates answers given a task prefix token `[VQA]` followed by the question. It is noteworthy that ViTLP can intrinsically generate interpretable grounding **regions of interest (ROI)**, i.e., layout coordinates of answers, to verify the generation.

3 Experiments

3.1 Experiment Setup

Implementation Details. We implement ViTLP with a 12-layer ViT (Dosovitskiy et al., 2021) image encoder and a 6-layer text-layout decoder. The Transformer hidden size is $d = 768$ with 12 attention heads. In pre-training, the input image height and width are 1920×1600 with the 32×32 ViT patch size, and the decoder segmented sequence length is $M = 1024$. Following LayoutLMv2 (Xu et al., 2021), the layout location coordinates are normalized into discrete bins of $[0, 1000]$, resulting that the vocabulary size of the layout head is 1001. The multi-segment prefix ratio is set as $\alpha_p = 0.25$. We use the AdamW optimizer (Loshchilov and Hutter, 2019) to train ViTLP in 250K steps, with the batch size of 384 and initial learning rate of $2e$-4 with cosine decay. More implementation details are provided in Appendix A.2.

Pre-training Data. Following prior work (Xu et al., 2021), we use IIT-CDIP Test Collection 1.0 (Lewis et al., 2006) containing 11M document images for pre-training. Following DONUT (Kim et al., 2022), we generate 2M synthetic document images with text and layout annotations. Another four supplementary datasets with 0.4M document images are also added to augment the diversity of pre-training data, including PubLayNet (Zhong et al., 2019), DocBank (Li et al., 2020), SciTSR (Chi et al., 2019), and IAM (Marti and Bunke, 2002). We use our internal OCR tool to extract words with location coordinates from the IIT-CDIP and PubLayNet images. Words with locations are provided in IAM, SciTSR, and DocBank. Refer to Appendix A.1 for more detailed data statistics.

Evaluation Tasks. We highlight that ViTLP are capable of handling both 1) *perception tasks* of document OCR and 2) *cognition tasks* of visual document understanding (VDU). To evaluate the comprehensive capabilities of ViTLP, we compare to baselines on each task as summarized in Table 1.

For OCR evaluation, we conduct two benchmark OCR sub-tasks, i.e., document text *localization* and *recognition*. We evaluate model performance on SROIE competition[4] Task #1 for text localization and Task #2 for text recognition. The text localization task is evaluated by DetEval protocol (Wolf and Jolion, 2006) which calculates the precision, recall, and F1 based on the *area of overlapping regions* between model predictions and ground-truth text coordinates. The text recognition task evaluates the word-level precision, recall, and F1 based on exact word match.

For VDU evaluation, we conduct three document understanding tasks. 1) *Form Understanding*. Given a document image and its word entities, it is a sequential labeling task to predict the BIO tags for each textual entity. We use FUNSD (Jaume et al., 2019) which contains 199 scanned forms, and the entities are labeled in four categories: *Header*, *Question*, *Answer*, and *Other*. FUNSD is divided into 149 images for training and 50 for testing. We report entity-level F1 as the evaluation score. 2)

[4] https://rrc.cvc.uab.es/?ch=13&com=tasks

Receipt Understanding. We use CORD (Park et al., 2019) containing 800 training and 100 testing images of real-world receipts. The receipt entities are labeled in 30 categories. We use entity-level F1 for evaluation. 3) *Document Classification*. We conduct experiments on the RVL-CDIP dataset (Harley et al., 2015) containing 400K scanned documents in 16 classes. We adopt classification accuracy as the evaluation metric. For the sequence labeling tasks on FUNSD, we perform multi-segment fine-tuning on those samples whose entity-word sequences exceed the maximum decoder sequence length. This differs from previous work that truncates the input sequences into certain tokens, e.g., 512 tokens in LayoutLMv2 (Xu et al., 2021).

Besides, we evaluate generative question answering tasks on the DocVQA (Mathew et al., 2020) and InfographicVQA (Mathew et al., 2022) datasets. DocVQA consists of 12K document images with 50K QA pairs, and InfographicVQA contains 5.4K document images with 30K QA pairs. Since the answer word locations are not provided in the training sets, we use an OCR tool to locate the coordinates of answer words with heuristic text matching. In this way, we feed the answers with grounding coordinates to ViTLP for document VQA fine-tuning.

3.2 OCR Evaluation Results

We compare ViTLP with representative OCR baselines on SROIE 2019 benchmark (Huang et al., 2019). The text localization baselines include CRAFT (Baek et al., 2019), YOLO-v3 (Redmon and Farhadi, 2018), CTPN (Tian et al., 2016), and EAST (Zhou et al., 2017). The text recognition baselines include BiLSTM-ResNet, BiLSTM-CTC (Lee and Osindero, 2016), UNet-CRNN (Ronneberger et al., 2015), and TrOCR (Li et al., 2023). Unlike conventional OCR models that first perform text localization and then use the localized text-regions for text recognition, ViTLP performs text localization and recognition in unified text-layout sequence generation, which does not need ground truth text-region inputs in the recognition task.

Table 2 shows the OCR evaluation performance. ViTLP outperforms most baseline methods on both localization and recognition tasks. ViTLP underperforms TrOCR, given that TrOCR is a strong pre-trained model for two-stage OCR text recognition, while ViTLP performs text localization and recognition in one stage. Note that the SROIE training samples are few, i.e., only 626 images, and the input text coordinates are at textline-level, which are different from our word-level pre-training input format and thus render it challenging to fine-tune our model. Nonetheless, ViTLP can still achieve competitive performance by fine-tuning on the limited samples without additional data augmentation (Li et al., 2023), successfully adapting to output the textline coordinates that have never met in the pre-training phase. We also provide qualitative ViTLP zero-shot OCR examples in Appendix C.

	Text Localization Task		
Method	**Area-Precision**	**Area-Recall**	**Area-F1**
CRAFT	62.73	59.94	61.31
YOLO-v3	77.29	79.32	78.29
CTPN	81.14	87.23	84.07
EAST	85.07	87.17	86.11
ViTLP	91.62	91.68	91.65
	Text Recognition Task		
Method	**Word-Precision**	**Word-Recall**	**Word-F1**
BiLSTM-ResNet	74.05	77.81	75.88
BiLSTM-CTC	83.38	87.37	85.33
UNet-CRNN	85.77	86.48	86.12
TrOCR†	95.89	95.74	95.82
ViTLP	93.07	92.52	92.79

Table 2: OCR text localization and recognition results on SROIE 2019 benchmark. †TrOCR uses the ground-truth cropped image regions as inputs, whereas ViTLP performs text localization and recognition in a unified stage. All scores are reported in percentage.

3.3 VDU Evaluation Results

We compare ViTLP with competitive pre-trained baselines including i) general method RoBERTa (Liu et al., 2019), ii) discriminative VDU models: LayoutLM (Xu et al., 2020), SPADE (Hwang et al., 2021), SelfDoc (Li et al., 2021b), TITL (Powalski et al., 2021), LayoutLMv2 (Xu et al., 2021), LiLT (Wang et al., 2022a), FormNet (Lee et al., 2022) and iii) generative VDU model DONUT (Kim et al., 2022). Table 3 shows the VDU task performance.

Information Extraction. According to Table 3, our model achieves better F1 scores compared to most baselines on FUNSD and CORD. The results indicate that ViTLP can develop a thorough understanding of form/receipt structures from images. Nonetheless, ViTLP underperforms the best discriminative baselines, i.e., LiLT on FUNSD and FormNet on CORD. We believe this is because pre-trained discriminative VDU models have natural advantages over generative models for the information extraction task, which is formulated as token-level classification. Besides, ViTLP outperforms DONUT, proving that layout modeling is as nec-

Method	Modeling Type	# Param.	Maximum Doc-Length	FUNSD (F1)	CORD (F1)	RVL-CDIP (Acc)
$\text{RoBERTa}_{\text{base}}$ (Liu et al., 2019)	Discriminative (w/ OCR Input)	125M	512	66.48	93.54	90.06
$\text{LayoutLM}_{\text{base}}$ (Xu et al., 2020)		160M	512	79.27	–	94.42
SPADE (Hwang et al., 2021)		110M	512	70.50	91.50	–
SelfDoc (Li et al., 2021b)		137M	1024	83.36	–	93.81
$\text{TILT}_{\text{base}}$ (Powalski et al., 2021)		230M	512	–	95.11	95.25
$\text{LayoutLMv2}_{\text{base}}$ (Xu et al., 2021)		200M	512	82.76	94.95	95.25
$\text{LiLT}_{\text{base}}$ (Wang et al., 2022a)		–	512	88.41	96.07	95.68
FormNet (Lee et al., 2022)		217/345M†	1024	84.69	97.28	–
DONUT (Kim et al., 2022)	Generative (w/o OCR Input)	259M	1536	–	84.10	95.30
ViTLP		253M	Any-length	87.61	95.59	95.36

Table 3: VDU evaluation results on form understanding (FUNSD), receipt understanding (CORD), and document classification (RVL-CDIP). † FormNet has different sizes of 217M and 345M for FUNSD and CORD (Lee et al., 2022). "Maximum Doc-Length" denotes the maximum tokens of an input text sequence that the model can handle.

essary as language modeling to generative VDU models. For example, for the CORD images, entities with the same semantic label `<menu.price>` are always located in the same rightmost column of the receipt, sharing adjacent layout coordinates. Layout modeling can help generative VDU models better extract such structural-aware information.

Document Classification. From Table 3, we can see that ViTLP achieves the second best performance on classification accuracy. We also observe that the performance among TILT, LayoutLMv2, DONUT, and ViTLP are quite close. This may be because document classification is a coarse-grained task, wherein the vision modality contributes the most to classification performance, and the OCR text modality brings an incremental gain. Though ViTLP is suboptimal compared to LiLT, OCR-free generative methods are more flexible and lightweight because no pre-processed OCR texts are needed for input.

3.4 Further Discussion

3.4.1 Ablation Study

We conduct ablation studies on the effect of hierarchical text-layout modeling and multi-segment pre-training scheme. We compare ViTLP with three variants: i) pre-training with the language modeling objective only, without the layout modeling objective; ii) truncating long input document sequences in pre-training, without the multi-segment strategy; iii) generating four layout coordinate tokens for each word in a long flatten sequence, without hierarchical text-layout modeling.

Table 4 displays the ablation performance. We can observe that discarding the layout modeling objective leads to a substantial performance drop, i.e., 6.19 and 4.05 F1 drops on FUNSD and CORD. The results suggest that generative pre-training on the layout modality can enhance the document understanding capability of VDU models. Besides, truncating long document inputs without the multi-segment pre-training strategy leads to lower performance. We believe that the multi-segment pre-training scheme enables ViTLP to model complete text and layout tokens of the pre-training corpora, which benefits the pre-trained model performance. We can also see that removing hierarchical text-layout modeling causes performance descent. It validates that hierarchical modeling is effective for interleaved text-layout information fusion.

Ablation Variants	FUNSD (F1)	CORD (F1)
ViTLP	87.61	95.59
w/o layout modeling	81.42	91.54
w/o multi-segment training	86.73	95.01
w/o hierarchical modeling	86.28	94.86

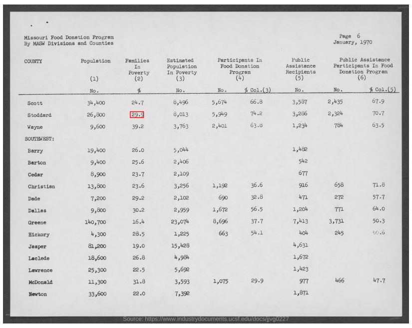
Table 4: Ablation model performance on the information extraction tasks.

Generative Model	DocVQA	InfographicVQA
Dessurt (Davis et al., 2022)	63.2	–
DONUT (Kim et al., 2022)	67.5	11.6
ViTLP	65.9	28.7

Table 5: The results are reported on Average Normalized Levenshtein Similarity (ANLS) between the model-generated answers and ground truth.

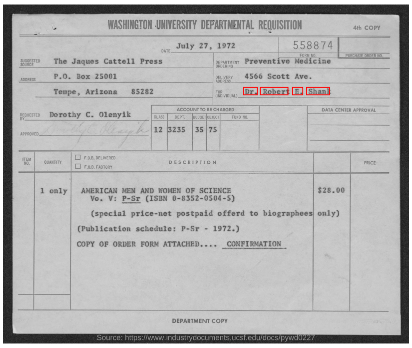
3.4.2 Generative Document VQA

Results and Analysis. Table 5 presents the performance of generative VDU models on DocVQA and InfographicVQA datasets. We can see that ViTLP underperforms DONUT by a slight margin



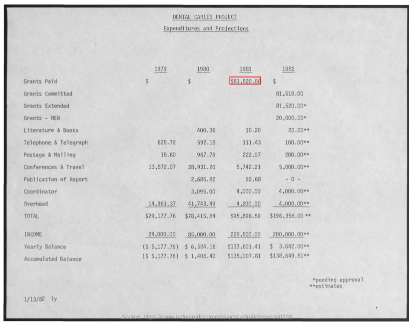
Question: What % of families are in poverty in the county 'Stoddard'?

ViTLP output: {["29.9", [314, 336, 346, 354]]}



Question: For which individual was this request made?

ViTLP output: {["Dr." , [593, 245, 626, 266]],
["Robert" , [634, 245, 702, 266]],
["E." , [713, 245, 738, 266]],
["Shank" , [748, 245, 804, 266]]}



Question: How much grands paid in the year 1981, according to the table?

ViTLP output: {["$81,520.00", [557, 229, 632, 251]]}

Figure 3: Visualization of ViTLP generated answers on DocVQA. The ViTLP output answer sequences consist of answer words (in blue) and corresponding location coordinates (in red). For direct visualization, we draw the region of interest (ROI) **referring to the output layout coordinates** on the image.

on DocVQA and surpasses DONUT by a significant margin on InfographicVQA. As discussed in Kim et al. (2022), DocVQA images are similar to the pre-training IIT-CDIP images, pre-training data quality may have a considerable influence on the performance of DocVQA. The average results show that ViTLP develops better overall document VQA performance than the strong generative model DONUT, which validates the effectiveness of our generative pre-training approach.

Document VQA with Interpretable Grounding. Owing to the fine-grained *word-level* grounding capability learned in the pre-training stage, ViTLP can be fine-tuned to predict the regions of interest (ROI) associated with the generated answers, which is unprecedented to prior document VQA models. As shown in Figure 3, the output ROI grounding-boxes *as visual rationales* can help humans easily verify the model-generated answers, making the answer generation process *interpretable to humans* where the model output derives from. See more examples of grounding document VQA with ViTLP in Appendix D.

4 Related Work

Visual document processing with multimodal pre-training is widely studied. From the perspectives of the document processing pipelines and model architectures, existing works can be generally divided into strands of research as listed below.

OCR-based Methods. Most initial VDU efforts adopt OCR tools to localize and recognize document layouts and texts, and then feed them to the multimodal pre-trained models (Xu et al., 2021; Appalaraju et al., 2021; Li et al., 2021a; Peng et al., 2022; Li et al., 2021b; Bai et al., 2023; Lee et al., 2023a). These methods usually involve multiple pre-training objectives over the vision, text, and layout. For instance, document text location (Xu et al., 2020, 2021), paragraph and table regions (Li et al., 2021b; Wang et al., 2022b) are rich in structural information to align visual features with text embeddings. Though promising, these pipeline models suffer from heavy OCR pre-processing overhead. Moreover, incorrect OCR results may propagate errors to downstream tasks like document question answering (Kim et al., 2022).

OCR-free Methods. There appear recent studies (Kim et al., 2022; Lee et al., 2023b; Kil et al., 2023) that jointly consider text reading and understanding without external OCR pipelines. For instance, Kim et al. (2022) takes document images as input to the model without prerequisite OCR results and conducts visual language pre-training. Lee et al. (2023b) further improves the pre-training objectives over large-scaled visual webpage corpora. Kil et al. (2023) employs multiple pre-training tasks jointly to encourage the pre-trained model to learn text recognition capability explicitly and spatial reasoning capability implicitly.

Our research falls within the OCR-free branch. Different from existing works, we first study generative joint text-layout modeling conditioned on input document images. Our empirical results also validate that layout information not only enhances the learned representations for downstream VDU tasks but also can make the generation outputs more interpretable with visual groundings.

LLM-backbone Methods. Most recent studies leverage large language models (LLMs) to tackle multimodal document tasks (Zhang et al., 2023; Ye et al., 2023; Wang et al., 2023). LLaVAR (Zhang et al., 2023) inherits LLaVA architecture (Liu et al., 2023) which directly projects the visual features to LLM embeddings and performs instruction tuning on visual document data. DocLLM (Wang et al., 2023) uses spatial attention to inject 2D layout information into Llama 2 (Touvron et al., 2023) with supervised fine-tuning and first enables LLMs to process document information extraction tasks. Thanks to LLMs' powerful reasoning and generation abilities, utilizing LLMs for visual document processing has become a prominent research trend.

5 Conclusion

We propose visually guided generative text-layout pre-training (ViTLP) to enhance visual document processing covering the OCR and VDU tasks. In the pre-training phase, ViTLP optimizes hierarchical language and layout modeling objectives to generate interleaved text-layout target sequences. Moreover, the proposed multi-segment pre-training scheme enables ViTLP to process long documents with arbitrary lengths. ViTLP can function as a native OCR model to locate and recognize texts of document images. Experiments also show that ViTLP achieves superior performance on various VDU tasks with document grounding capability.

Limitations

Our community has entered the era of large language models with multimodal capabilities (Dai et al., 2023; OpenAI, 2023). However, regarding the model size, ViTLP is still a rather small-scale pre-trained model[5], which limits its potential to become an interactive and generalized document AI assistant. In future work, we plan to explore two paths: i) scaling up ViTLP with more parameters and training data, extending it to a more powerful foundation document model; ii) integrating ViTLP's *document-specific* text-layout image encoder with *generalized* advanced LLMs (Chiang et al., 2023; Touvron et al., 2023) and visual instruction tuning (Liu et al., 2023; Zhu et al., 2024) to build up an interactive document AI assistant.

[5]It is because we commenced the ViTLP project in mid-2022 and finished pre-training in early 2023, see the first version at https://openreview.net/forum?id=ARtBIBAmNR.

Remarks and Future direction. i) ViTLP processes document images already calibrated in angle. Hence, we use 4 coordinates to represent the localization of words. It is feasible to pre-train ViTLP to generate 8 coordinates which can represent the angle of words. We choose word-level segmentation for pre-training because *a word is the elementary unit of document texts*. Word-level segmentation is also beneficial to fine-grained grounding, e.g., VQA with answer-word grounding. ii) We propose a multi-segment processing scheme to permit long sequence lengths on the *decoder side*. However, the document pixel inputs are also constrained by the resolution on the ViT *encoder side*. For the problem of long document processing, ViTLP only tackles the half. Processing document images with high resolutions and multiple pages is an intriguing problem for future research.

Acknowledgements

We appreciate constructive comments from anonymous ARR reviewers. We thank Bin Liang from CUHK for valuable discussion. This research work is partially supported by CUHK direct grant No. 4055209 and CUHK Knowledge Transfer Project Fund No. KPF23GWP20.

References

Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R. Manmatha. 2021. Docformer: End-to-end transformer for document understanding. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 973–983.

Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee. 2019. Character region awareness for text detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9357–9366.

Haoli Bai, Zhiguang Liu, Xiaojun Meng, Li Wentao, Shuang Liu, Yifeng Luo, Nian Xie, Rongfu Zheng, Liangwei Wang, Lu Hou, Jiansheng Wei, Xin Jiang, and Qun Liu. 2023. Wukong-reader: Multi-modal pre-training for fine-grained visual document understanding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13386–13401, Toronto, Canada. Association for Computational Linguistics.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

Aydar Bulatov, Yury Kuratov, and Mikhail Burtsev. 2022. Recurrent memory transformer. In *Advances in Neural Information Processing Systems*, volume 35, pages 11079–11091. Curran Associates, Inc.

Ting Chen, Saurabh Saxena, Lala Li, David J. Fleet, and Geoffrey Hinton. 2022. Pix2seq: A language modeling framework for object detection. In *International Conference on Learning Representations*.

Xingyu Chen, Zihan Zhao, Lu Chen, JiaBao Ji, Danyang Zhang, Ao Luo, Yuxuan Xiong, and Kai Yu. 2021. WebSRC: A dataset for web-based structural reading comprehension. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4173–4185, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zewen Chi, Heyan Huang, Heng-Da Xu, Houjin Yu, Wanxuan Yin, and Xian-Ling Mao. 2019. Complicated table structure recognition.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Wenliang Dai, Junnan Li, DONGXU LI, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 49250–49267. Curran Associates, Inc.

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

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

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. In *Advances in Neural Information Processing Systems*, volume 34, pages 39–50. Curran Associates, Inc.

Adam W. Harley, Alex Ufkes, and Konstantinos G. Derpanis. 2015. Evaluation of deep convolutional nets for document image classification and retrieval. ICDAR 2015, page 991–995.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus).

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, page 4083–4091, New York, NY, USA. Association for Computing Machinery.

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

Wonseok Hwang, Jinyeong Yim, Seunghyun Park, Sohee Yang, and Minjoon Seo. 2021. Spatial dependency parsing for semi-structured document information extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 330–343, Online. Association for Computational Linguistics.

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.

Anoop R Katti, Christian Reisswig, Cordula Guder, Sebastian Brarda, Steffen Bickel, Johannes Höhne, and Jean Baptiste Faddoul. 2018. Chargrid: Towards understanding 2D documents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4459–4469, Brussels, Belgium. Association for Computational Linguistics.

J. Kil, S. Changpinyo, X. Chen, H. Hu, S. Goodman, W. Chao, and R. Soricut. 2023. Prestu: Pre-training for scene-text understanding. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15224–15234.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2022. Ocr-free document understanding transformer. In *Computer Vision - ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, page 498–517, Berlin, Heidelberg. Springer-Verlag.

C. Lee and S. Osindero. 2016. Recursive recurrent nets with attention modeling for ocr in the wild. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2231–2239.

Chen-Yu Lee, Chun-Liang Li, Timothy Dozat, Vincent Perot, Guolong Su, Nan Hua, Joshua Ainslie, Renshen Wang, Yasuhisa Fujii, and Tomas Pfister. 2022. FormNet: Structural encoding beyond sequential modeling in form document information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3735–3754, Dublin, Ireland. Association for Computational Linguistics.

Chen-Yu Lee, Chun-Liang Li, Hao Zhang, Timothy Dozat, Vincent Perot, Guolong Su, Xiang Zhang, Kihyuk Sohn, Nikolay Glushnev, Renshen Wang, Joshua Ainslie, Shangbang Long, Siyang Qin, Yasuhisa Fujii, Nan Hua, and Tomas Pfister. 2023a. FormNetV2: Multimodal graph contrastive learning for form document information extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9011–9026, Toronto, Canada. Association for Computational Linguistics.

Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2023b. Pix2Struct: Screenshot parsing as pretraining for visual language understanding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 18893–18912. PMLR.

D. Lewis, G. Agam, S. Argamon, O. Frieder, D. Grossman, and J. 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*, page 665–666, New York, NY, USA. Association for Computing Machinery.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chenliang Li, Bin Bi, Ming Yan, Wei Wang, Songfang Huang, Fei Huang, and Luo Si. 2021a. StructuralLM: Structural pre-training for form understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6309–6318, Online. Association for Computational Linguistics.

Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. 2023. Trocr: Transformer-based optical character recognition with pre-trained models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):13094–13102.

Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. 2020. DocBank: A benchmark dataset for document layout analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 949–960, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. 2021b. Selfdoc: Self-supervised document representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5652–5660, Nashville, TN, USA.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA*.

Bodhisattwa Prasad Majumder, Navneet Potti, Sandeep Tata, James Bradley Wendt, Qi Zhao, and Marc Najork. 2020. 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, Online. Association for Computational Linguistics.

Urs-Viktor Marti and H. Bunke. 2002. The iam-database: An english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, 5:39–46.

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V. Jawahar. 2022. Infographicvqa. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2582–2591.

Minesh Mathew, Dimosthenis Karatzas, R. Manmatha, and C. V. Jawahar. 2020. Docvqa: A dataset for vqa on document images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2199–2208.

OpenAI. 2023. Gpt-4 technical report.

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.

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

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 *Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, Proceedings, Part II 16*, pages 732–747. Springer.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI 2015*. MICCAI.

Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. 2016. Detecting text in natural image with connectionist text proposal network. In *European Conference on Computer Vision*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Dongsheng Wang, Natraj Raman, Mathieu Sibue, Zhiqiang Ma, Petr Babkin, Simerjot Kaur, Yulong Pei, Armineh Nourbakhsh, and Xiaomo Liu. 2023. Docllm: A layout-aware generative language model for multimodal document understanding.

Jiapeng Wang, Lianwen Jin, and Kai Ding. 2022a. LiLT: A simple yet effective language-independent layout transformer for structured document understanding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7747–7757, Dublin, Ireland. Association for Computational Linguistics.

Zilong Wang, Jiuxiang Gu, Chris Tensmeyer, Nikolaos Barmpalios, Ani Nenkova, Tong Sun, Jingbo Shang, and Vlad Morariu. 2022b. MGDoc: Pre-training with multi-granular hierarchy for document image understanding. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3984–3993, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Christian Wolf and Jean-Michel Jolion. 2006. Object count/area graphs for the evaluation of object detection and segmentation algorithms. *Document Analysis and Recognition*, 8:280–296.

Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. 2021. LayoutLMv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2579–2591, Online. Association for Computational Linguistics.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. Layoutlm: Pre-training of text and layout for document image understanding. In *KDD 2020*, page 1192–1200, New York, NY, USA. Association for Computing Machinery.

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Yuhao Dan, Chenlin Zhao, Guohai Xu, Chenliang Li, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. 2023. mplug-docowl: Modularized multimodal large language model for document understanding.

Yanzhe Zhang, Ruiyi Zhang, Jiuxiang Gu, Yufan Zhou, Nedim Lipka, Diyi Yang, and Tong Sun. 2023. Llavar: Enhanced visual instruction tuning for text-rich image understanding.

X. Zhong, J. Tang, and A. Jimeno Yepes. 2019. Publaynet: Largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022.

Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. 2017. East: An efficient and accurate scene text detector. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2642–2651.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2024. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations*.

Dataset	Size	Proportion	Document Type
IIT-CDIP	$10,816,672$	81.89%	Scanned Document
SynthDog	$2,000,000$	15.14%	Synthetic Document
PublayNet	$261,076$	1.98%	Scientific Paper
DocBank	$125,815$	0.95%	Arxiv Paper
SciTSR	$3,536$	0.03%	Figure and Table
IAM	$1,198$	0.01%	Hand Written

Table 6: Pre-training dataset statistics.

A Experiment Details

A.1 Pre-training Data Statistics

Table 6 shows the pre-training data statistics. Following previous work, e.g., LayoutLMv2 (Xu et al., 2021), we use 11M IIT-CDIP document images as the main pre-training data. Besides, we follow Kim et al. (2022) and Davis et al. (2022) to include 2M machine-rendered synthetic documents for generative pre-training. Specifically, we adapt the official SynthDog generator[6] to generate synthetic document images with text and layout metadata. The other four corpora, i.e., PublayNet, DocBank, SciTSR, and IAM, account for only $\sim 3\%$ pre-training data whereby we aim to improve the diversity of pre-training document types.

The distribution of document sequence lengths is displayed in Figure 4. The number of text-layout sequence tokens follows a *long-tailed distribution*: there exist some long documents with the sequence lengths ranging from 1024 to 3072. This brings a trade-off to pre-training. With a relatively short sequence length (e.g., 512 tokens in LayoutLMv2), language modeling on long documents is incomplete, as the sequence tokens are truncated and wasted. However, with a relatively long sequence length (e.g., 3072), the GPU computation and memory overload would become prohibitive, which also forbids large batch sizes for better performance.[7] The proposed multi-segment pre-training scheme can circumvent this bitter trade-off. Notably, the multi-segment processing scheme can be directly applied to long document fine-tuning (and inference). For example in the OCR and sequence labeling tasks, ViTLP also employs the multi-segment scheme to process the long documents by multiple segments with prefix context tokens.

[6]https://github.com/clovaai/donut/tree/master/synthdog

[7]Even assuming sufficient computation resources, the long-tailed distribution of document lengths would also cause enormous padding tokens in long sequence input to Transformers, leading to considerable waste of computational resources.

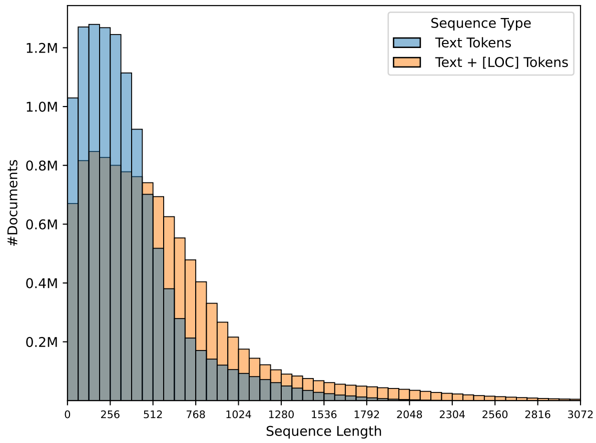


Figure 4: Distribution of document sequence lengths. The text sequences are tokenized by the standard BPE tokenizer (Radford et al., 2019).

A.2 Fine-tuning Hyperparameter Settings

OCR Text Localization and Recognition. Fine-tuning ViTLP for text localization and recognition follows the same objective Eq. (5) as pre-training. Since the SROIE 2019 (Huang et al., 2019) training set is rather small containing only 626 images, we fine-tune ViTLP for 10 epochs with the batch size of 1. The used learning rate and weight decay are $2e$-5 and $1e$-2. The input image resolution remains the same as pre-training, i.e., 1920×1600.

Information Extraction. For FUNSD (Jaume et al., 2019), the selected learning rate and weight decay are $1e$-4 and $1e$-2. For CORD (Park et al., 2019), the selected learning rate and weight decay[8] are $5e$-5 and $1e$-4. For both datasets, we fine-tune ViTLP for 75 epochs with the batch size of 8, using the same input image resolution as pre-training. Following the practice of prior work (Huang et al., 2022; Lee et al., 2023a), we use the shared segment-level layout coordinates as input instead of word-level coordinates, which can benefit the token classification accuracy in sequence labeling.

Document Classification. We use the learning rate of $1e$-4 and weight decay of $1e$-2 for the document classification task. We fine-tune ViTLP for 100 epochs with the global batch size of 320. The input image resolution is the same as pre-training.

Document VQA. Since the layout coordinates of answer words are not provided in the DocVQA (Mathew et al., 2020) and InfographicVQA

[8]For CORD, we search the configuration of learning rate in $\{2e\text{-}4, 1e\text{-}4, 5e\text{-}5, 3e\text{-}5, 2e\text{-}5, 1e\text{-}5\}$ and weight decay in $\{1e\text{-}2, 1e\text{-}4\}$.

(Mathew et al., 2022) datasets, we first conduct OCR on the training document images to obtain the texts with bounding-box coordinates. Then we apply a heuristic text-matching method to assign corresponding bounding-box coordinates to the answer words. It is worth noting that for the "Yes/No" questions that have no grounding answers on the images, we train ViTLP to generate a special answer token `[ANS_YES]` or `[ANS_NO]` without layout coordinates. For both datasets, we fine-tune ViTLP for 60 epochs with a batch size of 128. We use a learning rate of $3e$-5. Since the document images are high-resolution, for DocVQA, we set the fine-tuning image resolution as 2304×1920 which is multiplied by 1.2 based on the pre-training resolution. For InfographicVQA, the fine-tuning image resolution is set as 3200×1600. From our empirical experiments, we find that input image resolution is essential to document VQA performance, especially for InfographicVQA.

B Implementation Details of Sequential Layout Head

Given that multimodal interaction is learned by the stacked Transformer text-layout decoder layers, the LM and layout heads hereby function as a prober to output the next word and coordinate predictions. As introduced in Sec 2.2.2, the layout head predicts output probability $\mathrm{Prob}(\mathbf{L}_{i,j})$ of the four coordinates $\{\mathbf{L}_{i,j}\}_{j=1}^{4} = \{z_{x_1}, z_{y_1}, z_{x_2}, z_{y_2}\}_i$ based on the i-th global `[LOC]` token's final hidden state $\mathbf{H}_{i,0} = \mathbf{H}_i^{VTL} \in \mathbb{R}^d$ as follows.

$$\begin{cases} \mathbf{H}_{i,1} = \mathrm{GELU}\big(\mathbf{W}_h\mathbf{H}_{i,0}\big) \\ \mathbf{H}_{i,2} = \mathrm{GELU}\big(\mathbf{W}_h\mathbf{H}_{i,1} + \mathbf{E}_x^{'}(\mathbf{L}_{i,1})\big) \\ \mathbf{H}_{i,3} = \mathrm{GELU}\big(\mathbf{W}_h\mathbf{H}_{i,2} + \mathbf{E}_y^{'}(\mathbf{L}_{i,2})\big) \\ \mathbf{H}_{i,4} = \mathrm{GELU}\big(\mathbf{W}_h\mathbf{H}_{i,3} + \mathbf{E}_x^{'}(\mathbf{L}_{i,3})\big) \end{cases}$$

$$\mathrm{Prob}(\mathbf{L}_{i,j}) = \text{Softmax}\big(\mathbf{W}_L\mathbf{H}_{i,j}\big), \quad j \in \{1,2,3,4\}$$

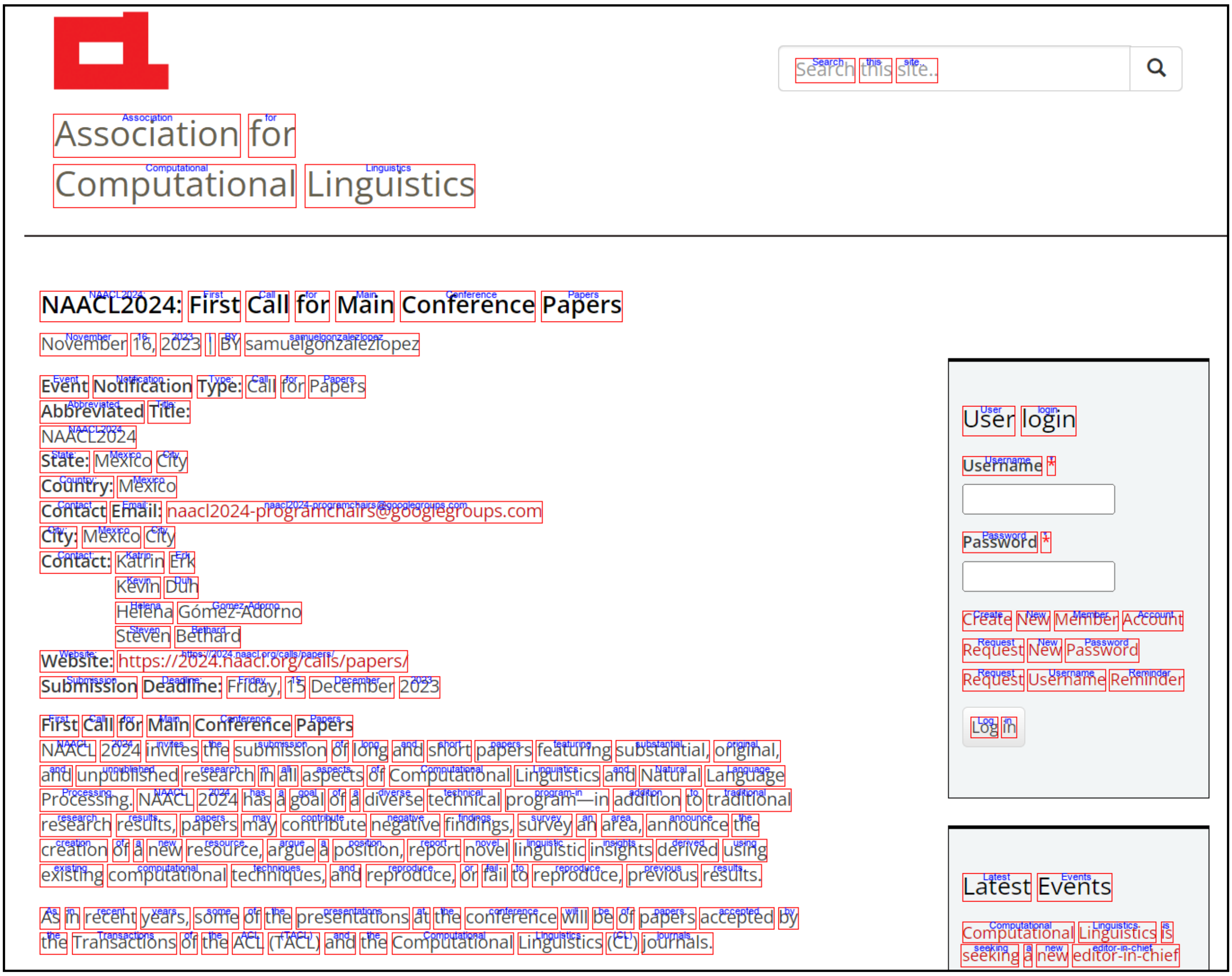
The coordinate tokens are quantized into a discrete range of $[0, 1000]$, making the layout-token vocabulary size of $|L| = 1001$. The layout head's parameters are lightweight including a hidden matrix $\mathbf{W}_h \in \mathbb{R}^{d\times d}$, two embeddings $\mathbf{E}_x^{'}(\cdot) \in \mathbb{R}^d$ and $\mathbf{E}_y^{'}(\cdot) \in \mathbb{R}^d$, and a linear projection $\mathbf{W}_L \in \mathbb{R}^{|L|\times d}$. We use the same GELU activation (Hendrycks and Gimpel, 2016) as in the Transformer layers. The layout head works sequentially, which is similar to a vanilla RNN, as each coordinate decoding step also considers the information of previous coordinates. Compared with naively using four independent linear heads, the sequential layout head can capture the spatial relation among the output coordinates (e.g., $x_1 < x_2$ and $y_1 < y_2$), bootstrapping more accurate coordinate prediction.

C Qualitative Cases of ViTLP Document OCR Functionality

Figure 5 to 7 demonstrate ViTLP's functionality on zero-shot document OCR. ViTLP outputs the interleaved OCR sequence consisting of words and corresponding bounding boxes.

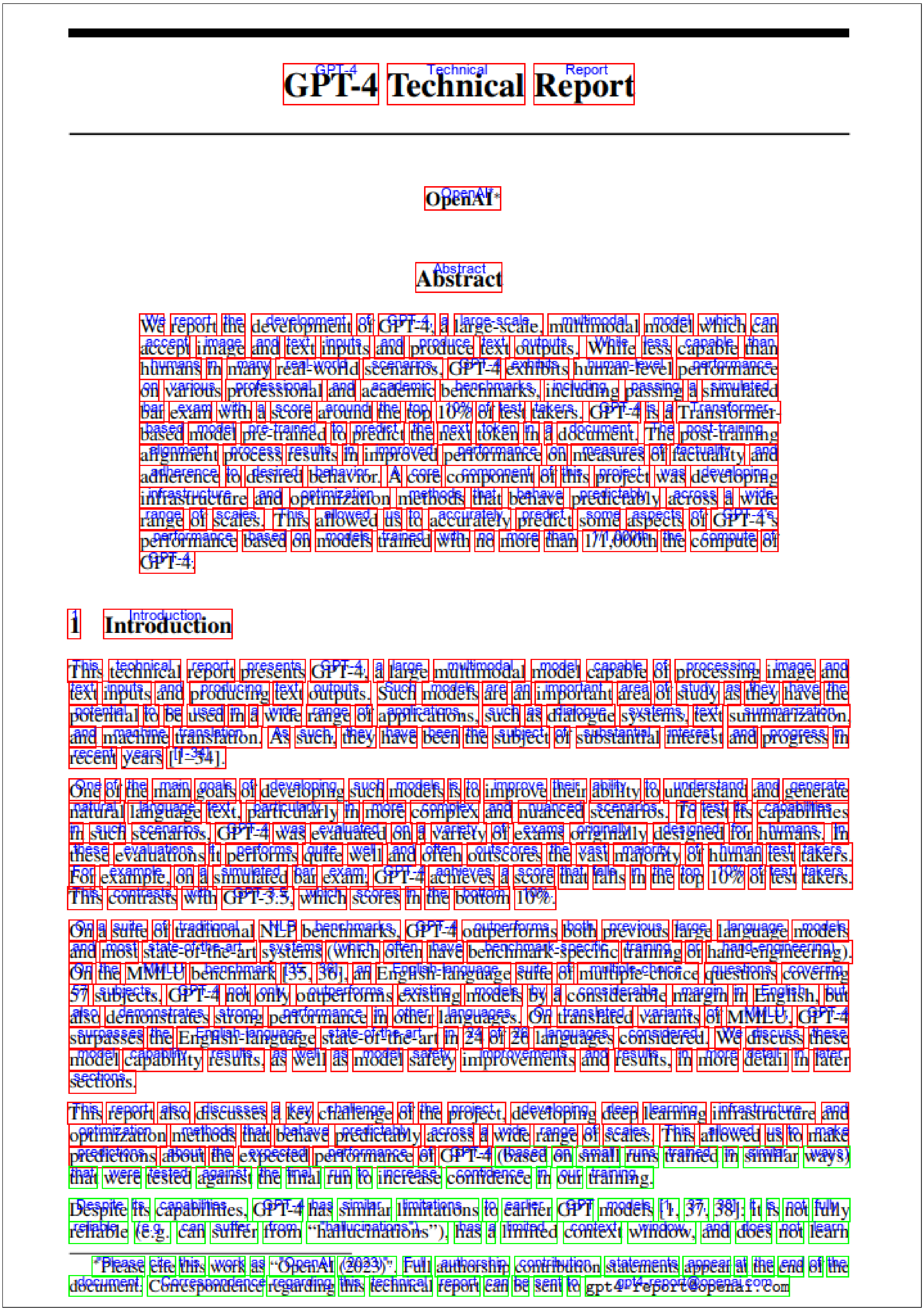
D Qualitative Cases of ViTLP Document VQA with Grounding Capability

Figure 8 showcases the ViTLP's VQA outputs on DocVQA with grounding capability. The top two examples are successful cases, and the bottom two are failure cases.



"Search", [647, 55, 695, 80]
"this", [699, 55, 725, 80]
"site..", [729, 55, 763, 80]
"Association", [41, 113, 193, 158]
"for", [200, 113, 238, 158]
"Computational", [41, 165, 239, 210]
"Linguistics", [246, 165, 385, 210]
"NAACL2024:", [30, 296, 146, 328]
"First", [151, 296, 193, 328]
"Call", [198, 296, 233, 328]
"for", [238, 296, 266, 328]
"Main", [271, 296, 319, 328]
"Conference", [324, 296, 434, 328]
"Papers", [439, 296, 505, 328]
"November", [30, 340, 101, 363]
"16,", [104, 340, 124, 363]
"2023", [128, 340, 161, 363]
"|", [165, 340, 173, 363]
"BY", [176, 340, 193, 363]
"samuelgonzalezlopez", [197, 340, 339, 363]
"Event", [30, 384, 69, 407]
"Notification", [73, 384, 154, 407]
"Type:", [158, 384, 195, 407]
"Call", [198, 384, 222, 407]
"for", [226, 384, 246, 407]
"Papers", [249, 384, 295, 407]
"Abbreviated", [30, 410, 115, 433]
"Title:", [118, 410, 152, 433]
"User", [783, 415, 826, 446]
"login", [831, 415, 876, 446]
"NAACL2024", [30, 436, 108, 459]
"State:", [30, 462, 70, 484]
"Mexico", [74, 462, 121, 484]
"City", [125, 462, 150, 484]
"Username", [783, 467, 848, 487]
"*", [852, 467, 859, 487]
"Country:", [30, 488, 90, 510]
"Mexico", [93, 488, 141, 510]
"Contact", [30, 514, 84, 536]
"Email:", [87, 514, 129, 536]
"naacl2024-programchairs@googlegroups.com", [133, 514, 440, 536]
"City:", [30, 540, 60, 562]
"Mexico", [64, 540, 112, 562]
"City", [115, 540, 140, 562]
"Password", [783, 546, 844, 567]
"*", [847, 546, 855, 567]
"Contact:", [30, 566, 88, 588]
"Katrin", [91, 566, 131, 588]
"Erk", [135, 566, 156, 588]
"Kevin", [91, 592, 128, 614]
"Duh", [131, 592, 159, 614]
"Helena", [91, 618, 138, 640]
"Gomez-Adorno", [142, 618, 243, 640]
"Create", [783, 627, 823, 648]
"New", [827, 627, 854, 648]
"Member", [858, 627, 910, 648]
"Account", [914, 627, 963, 648]
"Steven", [91, 643, 136, 666]
"Bethard", [140, 643, 193, 666]
"Request", [783, 656, 833, 680]
"New", [836, 656, 864, 680]
"Password", [867, 656, 927, 680]
"Website:", [30, 668, 90, 691]
"https://2024.naacl.org/calls/papers/", [93, 668, 330, 691]
"Request", [783, 688, 833, 710]
"Username", [836, 688, 900, 710]
"Reminder", [903, 688, 964, 710]
"Submission", [30, 695, 109, 718]
"Deadline:", [113, 695, 178, 718]
"Friday,", [182, 695, 226, 718]
"15", [230, 695, 246, 718]
"December", [250, 695, 319, 718]
"2023", [323, 695, 356, 718]
"First", [30, 735, 61, 758]
"Call", [64, 735, 90, 758]
"for", [93, 735, 113, 758]
"Main", [117, 735, 152, 758]
"Conference", [155, 735, 235, 758]
"Papers", [238, 735, 285, 758]
"Log", [790, 737, 812, 759]
"in", [815, 737, 827, 759]
"NAACL", [30, 761, 75, 784]
"2024", [79, 761, 112, 784]
"invites", [116, 761, 159, 784]
"the", [162, 761, 184, 784]
"submission", [188, 761, 264, 784]
"of", [268, 761, 281, 784]
"long", [285, 761, 314, 784]
"and", [317, 761, 343, 784]
"short", [346, 761, 382, 784]
"papers", [385, 761, 432, 784]
"featuring", [435, 761, 496, 784]
"substantial,", [500, 761, 576, 784]
"original,", [580, 761, 634, 784]
"and", [30, 787, 56, 809]
"unpublished", [59, 787, 143, 809]
"research", [146, 787, 205, 809]
"in", [208, 787, 221, 809]
"all", [224, 787, 240, 809]
"aspects", [243, 787, 294, 809]
"of", [297, 787, 311, 809]
"Computational", [315, 787, 414, 809]
"Linguistics", [417, 787, 487, 809]
"and", [490, 787, 516, 809]
"Natural", [519, 787, 570, 809]
"Language", [573, 787, 638, 809]
"Processing.", [30, 812, 106, 835]
"NAACL", [109, 812, 155, 835]
"2024", [158, 812, 191, 835]
"has", [195, 812, 218, 835]
"a", [222, 812, 230, 835]
"goal", [234, 812, 262, 835]
"of", [265, 812, 279, 835]
"a", [283, 812, 291, 835]
"diverse", [294, 812, 343, 835]
"technical", [346, 812, 406, 835]
"program—in", [410, 812, 494, 835]
"addition", [498, 812, 553, 835]
"to", [557, 812, 571, 835]
"traditional", [574, 812, 643, 835]
"research", [30, 838, 88, 861]
"results,", [92, 838, 141, 861]
"papers", [144, 838, 191, 861]
"may", [194, 838, 223, 861]
"contribute", [226, 838, 296, 861]
"negative", [300, 838, 356, 861]
"findings,", [360, 838, 416, 861]
"survey", [420, 838, 464, 861]
"an", [468, 838, 484, 861]
"area,", [488, 838, 522, 861]
"announce", [525, 838, 592, 861]
"the", [595, 838, 617, 861]
"creation", [30, 864, 85, 887]
"of", [89, 864, 102, 887]
"a", [106, 864, 114, 887]
"new", [117, 864, 146, 887]
"resource,", [149, 864, 212, 887]
"argue", [215, 864, 254, 887]
"a", [257, 864, 265, 887]
"position,", [269, 864, 327, 887]
"report", [330, 864, 373, 887]
"novel", [376, 864, 413, 887]
"linguistic", [416, 864, 475, 887]
"insights", [478, 864, 530, 887]
"derived", [533, 864, 584, 887]
"using", [587, 864, 623, 887]
"existing", [30, 890, 81, 913]
"computational", [85, 890, 182, 913]
"techniques,", [186, 890, 263, 913]
"and", [267, 890, 292, 913]
"reproduce,", [296, 890, 369, 913]
"or", [373, 890, 387, 913]
"fail", [391, 890, 411, 913]
"to", [415, 890, 428, 913]
"reproduce,", [432, 890, 505, 913]
"previous", [509, 890, 567, 913]
"results.", [570, 890, 619, 913]
"Latest", [783, 899, 839, 928]
"Events", [844, 899, 905, 928]
"As", [30, 934, 46, 957]
"in", [50, 934, 62, 957]
"recent", [66, 934, 108, 957]
"years,", [112, 934, 152, 957]
"some", [155, 934, 192, 957]
"of", [196, 934, 210, 957]
"the", [213, 934, 235, 957]
"presentations", [239, 934, 331, 957]
"at", [334, 934, 348, 957]
"the", [351, 934, 373, 957]
"conference", [377, 934, 452, 957]
"will", [455, 934, 477, 957]
"be", [481, 934, 498, 957]
"of", [501, 934, 515, 957]
"papers", [519, 934, 565, 957]
"accepted", [568, 934, 629, 957]
"by", [633, 934, 649, 957]
"Computational", [783, 949, 874, 971]
"Linguistics", [878, 949, 941, 971]
"is", [945, 949, 955, 971]
"the", [30, 960, 52, 983]
"Transactions", [56, 960, 141, 983]
"of", [144, 960, 158, 983]
"the", [162, 960, 183, 983]
"ACL", [187, 960, 212, 983]
"(TACL)", [216, 960, 258, 983]
"and", [262, 960, 287, 983]
"the", [291, 960, 313, 983]
"Computational", [317, 960, 416, 983]
"Linguistics", [420, 960, 489, 983]
"(CL)", [492, 960, 518, 983]
"journals.", [521, 960, 579, 983]
"seeking", [782, 973, 829, 996]
"a", [833, 973, 840, 996]
"new", [843, 973, 869, 996]
"editor-in-chief", [873, 973, 960, 996]

Figure 5: ViTLP OCR results on a webpage. For comprehensive visualization, we render the output texts (in blue) and bounding boxes (in red) according to the ViTLP's interleaved output sequence.


GPT-4 Technical Report

OpenAI*

Abstract

We report the development of GPT-4, a large-scale, multimodal model which can accept image and text inputs and produce text outputs. While less capable than humans in many real-world scenarios, GPT-4 exhibits human-level performance on various professional and academic benchmarks, including passing a simulated bar exam with a score around the top 10% of test takers. GPT-4 is a Transformer-based model pre-trained to predict the next token in a document. The post-training alignment process results in improved performance on measures of factuality and adherence to desired behavior. A core component of this project was developing infrastructure and optimization methods that behave predictably across a wide range of scales. This allowed us to accurately predict some aspects of GPT-4's performance based on models trained with no more than 1/1,000th the compute of GPT-4.

1 Introduction

This technical report presents GPT-4, a large multimodal model capable of processing image and text inputs and producing text outputs. Such models are an important area of study as they have the potential to be used in a wide range of applications, such as dialogue systems, text summarization, and machine translation. As such, they have been the subject of substantial interest and progress in recent years [1–34].

One of the main goals of developing such models is to improve their ability to understand and generate natural language text, particularly in more complex and nuanced scenarios. To test its capabilities in such scenarios, GPT-4 was evaluated on a variety of exams originally designed for humans. In these evaluations it performs quite well and often outscores the vast majority of human test takers. For example, on a simulated bar exam, GPT-4 achieves a score that falls in the top 10% of test takers. This contrasts with GPT-3.5, which scores in the bottom 10%.

On a suite of traditional NLP benchmarks, GPT-4 outperforms both previous large language models and most state-of-the-art systems (which often have benchmark-specific training or hand-engineering). On the MMLU benchmark [35, 36], an English-language suite of multiple-choice questions covering 57 subjects, GPT-4 not only outperforms existing models by a considerable margin in English, but also demonstrates strong performance in other languages. On translated variants of MMLU, GPT-4 surpasses the English-language state-of-the-art in 24 of 26 languages considered. We discuss these model capability results, as well as model safety improvements and results, in more detail in later sections.

This report also discusses a key challenge of the project, developing deep learning infrastructure and optimization methods that behave predictably across a wide range of scales. This allowed us to make predictions about the expected performance of GPT-4 (based on small runs trained in similar ways) that were tested against the final run to increase confidence in our training.

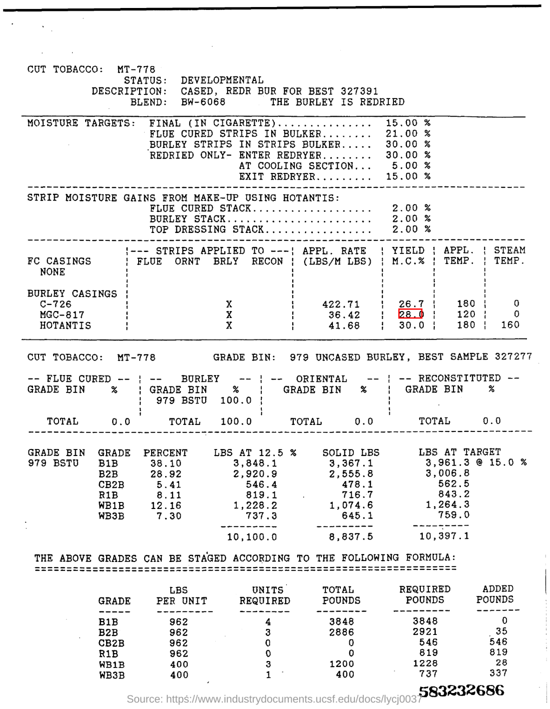
Despite its capabilities, GPT-4 has similar limitations to earlier GPT models [1, 37, 38]: it is not fully reliable (e.g. can suffer from "hallucinations"), has a limited context window, and does not learn

*Please cite this work as "OpenAI (2023)". Full authorship contribution statements appear at the end of the document. Correspondence regarding this technical report can be sent to gpt4-report@openai.com

Figure 6: ViTLP OCR results on a paper. For comprehensive visualization, we render the words and bounding boxes according to ViTLP's interleaved output sequence. The shown generated OCR results comprise two segments, as the generated tokens reach the decoder sequence length ($M = 1024$) in the first segment generation, and the generation process continues by the second segment. The bounding boxes of the first segment are in red, and the second are in green.

"GPT-4", [305, 46, 409, 77]
"Technical", [419, 46, 568, 77]
"Report", [578, 46, 688, 77]
"OpenAI\u2217", [460, 140, 542, 158]
"Abstract", [449, 199, 544, 221]
"We", [149, 238, 176, 255]
"report", [182, 238, 233, 255]
"the", [238, 238, 264, 255]
"development", [270, 238, 380, 255]
"of", [385, 238, 403, 255]
"GPT-4,", [409, 238, 470, 255]
"a", [476, 238, 485, 255]
"large-scale,", [491, 238, 588, 255]
"multimodal", [594, 238, 693, 255]
"model", [698, 238, 752, 255]
"which", [757, 238, 809, 255]
"can", [814, 238, 844, 255]
"accept", [149, 255, 204, 272]
"image", [210, 255, 264, 272]
"and", [270, 255, 301, 272]
"text", [307, 255, 340, 272]
"inputs", [346, 255, 399, 272]
"and", [405, 255, 437, 272]
"produce", [443, 255, 513, 272]
"text", [519, 255, 552, 272]
"outputs.", [558, 255, 627, 272]
"While", [636, 255, 689, 272]
"less", [696, 255, 728, 272]
"capable", [734, 255, 801, 272]
"than", [807, 255, 844, 272]
"humans", [149, 271, 216, 288]
"in", [221, 271, 238, 288]
"many", [244, 271, 292, 288]
"real-world", [297, 271, 388, 288]
"scenarios,", [393, 271, 479, 288]
"GPT-4", [485, 271, 542, 288]
"exhibits", [547, 271, 616, 288]
"human-level", [621, 271, 729, 288]
"performance", [734, 271, 844, 288]
"on", [149, 288, 170, 305]
"various", [175, 288, 238, 305]
"professional", [243, 288, 348, 305]
"and", [353, 288, 384, 305]
"academic", [389, 288, 471, 305]
"benchmarks,", [476, 288, 585, 305]
"including", [590, 288, 671, 305]
"passing", [677, 288, 741, 305]
"a", [746, 288, 756, 305]
"simulated", [761, 288, 844, 305]
"bar", [149, 305, 176, 322]
"exam", [181, 305, 227, 322]
"with", [233, 305, 271, 322]
"a", [276, 305, 286, 322]
"score", [291, 305, 337, 322]
"around", [342, 305, 402, 322]
"the", [407, 305, 433, 322]
"top", [439, 305, 466, 322]
"10%", [472, 305, 511, 322]
"of", [516, 305, 534, 322]
"test", [540, 305, 569, 322]
"takers.", [575, 305, 631, 322]
"GPT-4", [638, 305, 695, 322]
"is", [700, 305, 714, 322]
"a", [720, 305, 729, 322]
"Transformer-", [734, 305, 847, 322]
"based", [149, 321, 196, 338]
"model", [202, 321, 254, 338]
"pre-trained", [259, 321, 352, 338]
"to", [357, 321, 374, 338]
"predict", [379, 321, 437, 338]
"the", [443, 321, 469, 338]
"next", [474, 321, 510, 338]
"token", [515, 321, 562, 338]
"in", [567, 321, 584, 338]
"a", [589, 321, 598, 338]
"document.", [604, 321, 692, 338]
"The", [699, 321, 731, 338]
"post-training", [736, 321, 844, 338]
"alignment", [149, 337, 234, 354]
"process", [240, 337, 304, 354]
"results", [309, 337, 365, 354]
"in", [370, 337, 387, 354]
"improved", [392, 337, 474, 354]
"performance", [479, 337, 587, 354]
"on", [592, 337, 614, 354]
"measures", [619, 337, 699, 354]
"of", [704, 337, 722, 354]
"factuality", [727, 337, 808, 354]
"and", [813, 337, 844, 354]
"adherence", [149, 354, 236, 371]
"to", [242, 354, 259, 371]
"desired", [264, 354, 327, 371]
"behavior.", [332, 354, 412, 371]
"A", [419, 354, 434, 371]
"core", [440, 354, 477, 371]
"component", [482, 354, 579, 371]
"of", [584, 354, 602, 371]
"this", [607, 354, 639, 371]
"project", [644, 354, 704, 371]
"was", [710, 354, 743, 371]
"developing", [749, 354, 844, 371]
"infrastructure", [149, 371, 266, 388]
"and", [273, 371, 304, 388]
"optimization", [312, 371, 422, 388]
"methods", [429, 371, 503, 388]
"that", [510, 371, 543, 388]
"behave", [550, 371, 611, 388]
"predictably", [618, 371, 716, 388]
"across", [723, 371, 778, 388]
"a", [785, 371, 795, 388]
"wide", [802, 371, 844, 388]
"range", [149, 387, 197, 404]
"of", [203, 387, 221, 404]
"scales.", [228, 387, 285, 404]
"This", [294, 387, 333, 404]
"allowed", [340, 387, 408, 404]
"us", [414, 387, 434, 404]
"to", [440, 387, 457, 404]
"accurately", [464, 387, 553, 404]
"predict", [559, 387, 620, 404]
"some", [626, 387, 672, 404]
"aspects", [679, 387, 741, 404]
"of", [748, 387, 766, 404]
"GPT-4's", [772, 387, 844, 404]
"performance", [149, 403, 255, 420]
"based", [261, 403, 309, 420]
"on", [314, 403, 335, 420]
"models", [341, 403, 402, 420]
"trained", [407, 403, 466, 420]
"trained", [407, 403, 466, 420]
"with", [471, 403, 509, 420]
"no", [514, 403, 535, 420]
"more", [541, 403, 584, 420]
"than", [589, 403, 626, 420]
"1/1,000th", [632, 403, 712, 420]
"the", [717, 403, 743, 420]
"compute", [749, 403, 821, 420]
"of", [826, 403, 844, 420]
"GPT-4.", [149, 420, 209, 437]
"1", [71, 465, 85, 487]
"Introduction", [110, 465, 249, 487]
"This", [71, 503, 109, 521]
"technical", [115, 503, 194, 521]
"report", [200, 503, 252, 521]
"presents", [258, 503, 329, 521]
"GPT-4,", [335, 503, 398, 521]
"a", [404, 503, 414, 521]
"large", [420, 503, 463, 521]
"multimodal", [469, 503, 569, 521]
"model", [575, 503, 629, 521]
"capable", [635, 503, 702, 521]
"of", [708, 503, 726, 521]
"processing", [732, 503, 825, 521]
"image", [831, 503, 884, 521]
"and", [890, 503, 921, 521]
"text", [72, 520, 103, 538]
"inputs", [108, 520, 161, 538]
"and", [166, 520, 197, 538]
"producing", [203, 520, 290, 538]
"text", [295, 520, 327, 538]
"outputs.", [332, 520, 401, 538]
"Such", [408, 520, 450, 538]
"models", [456, 520, 518, 538]
"are", [523, 520, 549, 538]
"an", [555, 520, 575, 538]
"important", [580, 520, 664, 538]
"area", [669, 520, 705, 538]
"of", [710, 520, 728, 538]
"study", [733, 520, 780, 538]
"as", [785, 520, 803, 538]
"they", [808, 520, 845, 538]
"have", [850, 520, 890, 538]
"the", [895, 520, 921, 538]
"potential", [72, 537, 147, 555]
"to", [153, 537, 170, 555]
"be", [175, 537, 195, 555]
"used", [201, 537, 241, 555]
"in", [246, 537, 263, 555]
"a", [268, 537, 278, 555]
"wide", [283, 537, 325, 555]
"range", [331, 537, 379, 555]
"of", [384, 537, 402, 555]
"applications,", [408, 537, 518, 555]
"such", [523, 537, 563, 555]
"as", [569, 537, 587, 555]
"dialogue", [592, 537, 667, 555]
"systems,", [672, 537, 746, 555]
"text", [752, 537, 784, 555]
"summarization,", [790, 537, 923, 555]
"and", [72, 554, 102, 571]
"machine", [108, 554, 181, 571]
"translation.", [186, 554, 282, 571]
"As", [289, 554, 313, 571]
"such,", [318, 554, 363, 571]
"they", [369, 554, 406, 571]
"have", [411, 554, 451, 571]
"been", [456, 554, 497, 571]
"the", [502, 554, 528, 571]
"subject", [534, 554, 595, 571]
"of", [600, 554, 618, 571]
"substantial", [624, 554, 716, 571]
"interest", [721, 554, 784, 571]
"and", [790, 554, 821, 571]
"progress", [826, 554, 899, 571]
"in", [904, 554, 921, 571]
"recent", [72, 570, 124, 587]
"years", [129, 570, 174, 587]
"[1-34].", [179, 570, 242, 587]
"One", [72, 595, 106, 613]
"of", [111, 595, 128, 613]
"the", [133, 595, 158, 613]
"main", [163, 595, 205, 613]
"goals", [209, 595, 254, 613]
"of", [258, 595, 275, 613]
"developing", [280, 595, 372, 613]
"such", [377, 595, 415, 613]
"models", [419, 595, 480, 613]
"is", [485, 595, 499, 613]
"to", [503, 595, 520, 613]
"improve", [524, 595, 593, 613]
"their", [597, 595, 636, 613]
"ability", [640, 595, 694, 613]
"to", [698, 595, 715, 613]
"understand", [719, 595, 811, 613]
"and", [816, 595, 846, 613]
"generate", [850, 595, 921, 613]
"natural", [72, 611, 132, 629]
"language", [137, 611, 216, 629]
"text,", [222, 611, 259, 629]
"particularly", [265, 611, 365, 629]
"in", [371, 611, 388, 629]
"more", [393, 611, 438, 629]
"complex", [444, 611, 518, 629]
"and", [524, 611, 555, 629]
"nuanced", [561, 611, 633, 629]
"scenarios.", [638, 611, 725, 629]
"To", [733, 611, 755, 629]
"test", [760, 611, 790, 629]
"its", [796, 611, 816, 629]
"capabilities", [822, 611, 921, 629]
"in", [72, 628, 88, 645]
"such", [94, 628, 134, 645]
"scenarios,", [140, 628, 226, 645]
"GPT-4", [232, 628, 290, 645]
"was", [295, 628, 329, 645]
"evaluated", [335, 628, 417, 645]
"on", [423, 628, 445, 645]
"a", [451, 628, 460, 645]
"variety", [466, 628, 526, 645]
"of", [532, 628, 550, 645]
"exams", [556, 628, 611, 645]
"originally", [617, 628, 702, 645]
"designed", [708, 628, 785, 645]
"for", [791, 628, 816, 645]
"humans.", [822, 628, 895, 645]
"In", [903, 628, 921, 645]
"these", [72, 644, 116, 662]
"evaluations", [121, 644, 220, 662]
"it", [225, 644, 237, 662]
"performs", [243, 644, 321, 662]
"quite", [327, 644, 370, 662]
"well", [375, 644, 413, 662]
"and", [419, 644, 450, 662]
"often", [455, 644, 500, 662]
"outscores", [505, 644, 587, 662]
"the", [593, 644, 619, 662]
"vast", [625, 644, 659, 662]
"majority", [665, 644, 738, 662]
"of", [744, 644, 762, 662]
"human", [768, 644, 827, 662]
"test", [832, 644, 862, 662]
"takers.", [868, 644, 925, 662]
"For", [72, 661, 100, 679]
"example,", [105, 661, 182, 679]
"on", [187, 661, 208, 679]
"a", [213, 661, 223, 679]
"simulated", [228, 661, 310, 679]
"bar", [315, 661, 342, 679]
"exam,", [347, 661, 398, 679]
"GPT-4", [403, 661, 459, 679]
"achieves", [464, 661, 535, 679]
"a", [541, 661, 550, 679]
"score", [556, 661, 600, 679]
"that", [605, 661, 637, 679]
"falls", [642, 661, 678, 679]
"in", [683, 661, 700, 679]
"the", [705, 661, 731, 679]
"top", [736, 661, 763, 679]
"10%", [769, 661, 807, 679]
"of", [812, 661, 830, 679]
"test", [835, 661, 864, 679]
"takers.", [870, 661, 925, 679]
"This", [71, 677, 109, 695]
"contrasts", [114, 677, 190, 695]
"with", [195, 677, 233, 695]
"GPT-3.5,", [239, 677, 317, 695]
"which", [322, 677, 374, 695]
"scores", [379, 677, 433, 695]
"in", [438, 677, 455, 695]
"the", [460, 677, 486, 695]
"bottom", [492, 677, 552, 695]
"10%.", [557, 677, 602, 695]
"On", [72, 703, 98, 720]
"a", [103, 703, 112, 720]
"suite", [118, 703, 158, 720]
"of", [163, 703, 181, 720]
"traditional", [186, 703, 274, 720]
"NLP", [279, 703, 320, 720]
"benchmarks,", [325, 703, 433, 720]
"GPT-4", [438, 703, 494, 720]
"outperforms", [500, 703, 603, 720]
"both", [609, 703, 647, 720]
"previous", [652, 703, 725, 720]
"large", [730, 703, 772, 720]
"language", [778, 703, 855, 720]
"models", [860, 703, 921, 720]
"and", [72, 719, 101, 736]
"most", [106, 719, 147, 736]
"state-of-the-art", [152, 719, 276, 736]
"systems", [281, 719, 347, 736]
"(which", [352, 719, 410, 736]
"often", [415, 719, 457, 736]
"have", [462, 719, 501, 736]
"benchmark-specific", [506, 719, 670, 736]
"training", [674, 719, 739, 736]
"or", [744, 719, 762, 736]
"hand-engineering).", [766, 719, 925, 736]
"On", [72, 735, 97, 752]
"the", [102, 735, 128, 752]
"MMLU", [133, 735, 199, 752]
"benchmark", [204, 735, 297, 752]
"[35,", [303, 735, 337, 752]
"36],", [342, 735, 375, 752]
"an", [381, 735, 401, 752]
"English-language", [406, 735, 553, 752]
"suite", [559, 735, 598, 752]
"of", [603, 735, 621, 752]
"multiple-choice", [626, 735, 758, 752]
"questions", [764, 735, 843, 752]
"covering", [848, 735, 921, 752]
"57", [72, 752, 93, 769]
"subjects,", [99, 752, 173, 769]
"GPT-4", [179, 752, 236, 769]
"not", [242, 752, 269, 769]
"only", [275, 752, 313, 769]
"outperforms", [319, 752, 424, 769]
"existing", [430, 752, 498, 769]
"models", [503, 752, 566, 769]
"by", [571, 752, 593, 769]
"a", [598, 752, 608, 769]
"considerable", [613, 752, 723, 769]
"margin", [729, 752, 789, 769]
"in", [795, 752, 811, 769]
"English,", [817, 752, 889, 769]
"but", [894, 752, 921, 769]
"also", [72, 769, 106, 786]
"demonstrates", [111, 769, 224, 786]
"strong", [230, 769, 283, 786]
"performance", [289, 769, 397, 786]
"in", [402, 769, 419, 786]
"other", [425, 769, 468, 786]
"languages.", [474, 769, 564, 786]
"On", [571, 769, 597, 786]
"translated", [603, 769, 686, 786]
"variants", [692, 769, 759, 786]
"of", [764, 769, 782, 786]
"MMLU,", [788, 769, 860, 786]
"GPT-4", [865, 769, 921, 786]
"surpasses", [72, 785, 153, 802]
"the", [158, 785, 185, 802]
"English-language", [190, 785, 341, 802]
"state-of-the-art", [346, 785, 474, 802]
"in", [480, 785, 496, 802]
"24", [502, 785, 523, 802]
"of", [529, 785, 546, 802]
"26", [552, 785, 573, 802]
"languages", [579, 785, 665, 802]
"considered.", [670, 785, 768, 802]
"We", [776, 785, 804, 802]
"discuss", [809, 785, 872, 802]
"these", [877, 785, 921, 802]
"model", [72, 801, 126, 819]
"capability", [131, 801, 217, 819]
"results,", [223, 801, 285, 819]
"as", [291, 801, 309, 819]
"well", [315, 801, 352, 819]
"as", [358, 801, 376, 819]
"model", [381, 801, 436, 819]
"safety", [441, 801, 493, 819]
"improvements", [499, 801, 623, 819]
"and", [628, 801, 660, 819]
"results,", [666, 801, 728, 819]
"in", [734, 801, 751, 819]
"more", [756, 801, 801, 819]
"detail", [806, 801, 855, 819]
"in", [860, 801, 877, 819]
"later", [883, 801, 922, 819]
"sections.", [72, 818, 145, 836]
"This", [71, 843, 108, 860]
"report", [113, 843, 164, 860]
"also", [169, 843, 204, 860]
"discusses", [209, 843, 288, 860]
"a", [293, 843, 302, 860]
"key", [308, 843, 338, 860]
"challenge", [343, 843, 424, 860]
"of", [430, 843, 447, 860]
"the", [452, 843, 478, 860]
"project,", [484, 843, 548, 860]
"developing", [553, 843, 646, 860]
"deep", [652, 843, 692, 860]
"learning", [697, 843, 766, 860]
"infrastructure", [771, 843, 885, 860]
"and", [891, 843, 921, 860]
"optimization", [72, 860, 178, 877]
"methods", [183, 860, 254, 877]
"that", [259, 860, 291, 877]
"behave", [296, 860, 355, 877]
"predictably", [361, 860, 455, 877]
"across", [460, 860, 513, 877]
"a", [519, 860, 528, 877]
"wide", [533, 860, 574, 877]
"range", [579, 860, 626, 877]
"of", [631, 860, 649, 877]
"scales.", [654, 860, 709, 877]
"This", [716, 860, 753, 877]
"allowed", [759, 860, 825, 877]
"us", [830, 860, 849, 877]
"to", [854, 860, 870, 877]
"make", [876, 860, 921, 877]
"predictions", [72, 876, 167, 893]
"about", [172, 876, 220, 893]
"the", [225, 876, 251, 893]
"expected", [257, 876, 333, 893]
"performance", [338, 876, 447, 893]
"of", [452, 876, 470, 893]
"GPT-4", [475, 876, 532, 893]
"(based", [537, 876, 593, 893]
"on", [598, 876, 620, 893]
"small", [625, 876, 672, 893]
"runs", [677, 876, 714, 893]
"trained", [719, 876, 779, 893]
"in", [784, 876, 801, 893]
"similar", [806, 876, 866, 893]
"ways)", [872, 876, 922, 893]
"that", [72, 892, 103, 909]
"were", [109, 892, 150, 909]
"tested", [156, 892, 205, 909]
"against", [211, 892, 271, 909]
"the", [277, 892, 303, 909]
"final", [308, 892, 346, 909]
"run", [351, 892, 380, 909]
"to", [385, 892, 402, 909]
"increase", [407, 892, 477, 909]
"confidence", [483, 892, 575, 909]
"in", [581, 892, 597, 909]
"our", [603, 892, 631, 909]
"training.", [636, 892, 708, 909]
"Despite", [72, 917, 136, 934]
"its", [141, 917, 160, 934]
"capabilities,", [166, 917, 266, 934]
"GPT-4", [272, 917, 327, 934]
"has", [332, 917, 359, 934]
"similar", [365, 917, 423, 934]
"limitations", [428, 917, 518, 934]
"to", [523, 917, 539, 934]
"earlier", [544, 917, 598, 934]
"GPT", [603, 917, 643, 934]
"models", [648, 917, 708, 934]
"[1,", [714, 917, 737, 934]
"37,", [742, 917, 769, 934]
"38]:", [774, 917, 808, 934]
"it", [814, 917, 826, 934]
"is", [831, 917, 845, 934]
"not", [850, 917, 877, 934]
"fully", [882, 917, 922, 934]
"reliable", [72, 934, 137, 951]
"(e.g.", [143, 934, 182, 951]
"can", [190, 934, 220, 951]
"suffer", [226, 934, 277, 951]
"from", [283, 934, 325, 951]
"\"hallucinations\"),", [331, 934, 485, 951]
"has", [492, 934, 521, 951]
"a", [527, 934, 536, 951]
"limited", [542, 934, 604, 951]
"context", [610, 934, 674, 951]
"window,", [680, 934, 754, 951]
"and", [760, 934, 792, 951]
"does", [798, 934, 837, 951]
"not", [843, 934, 872, 951]
"learn", [878, 934, 921, 951]
"\"Please", [97, 961, 155, 977]
"cite", [160, 961, 188, 977]
"this", [192, 961, 220, 977]
"work", [225, 961, 264, 977]
"as", [269, 961, 285, 977]
"\"OpenAI", [290, 961, 361, 977]
"(2023)\".", [365, 961, 429, 977]
"Full", [435, 961, 466, 977]
"authorship", [471, 961, 552, 977]
"contribution", [557, 961, 651, 977]
"statements", [655, 961, 736, 977]
"appear", [741, 961, 793, 977]
"at", [797, 961, 811, 977]
"the", [816, 961, 840, 977]
"end", [844, 961, 872, 977]
"of", [877, 961, 893, 977]
"the", [897, 961, 921, 977]
"document.", [72, 976, 152, 992]
"Correspondence", [158, 976, 283, 992]
"regarding", [288, 976, 361, 992]
"this", [366, 976, 394, 992]
"technical", [399, 976, 468, 992]
"report", [473, 976, 519, 992]
"can", [524, 976, 550, 992]
"be", [555, 976, 573, 992]
"sent", [578, 976, 609, 992]
"to", [614, 976, 629, 992]
"gpt4-report@openai.com", [634, 976, 857, 992]

Figure 7: ViTLP OCR results as visualized in Figure 6 above.



CUT TOBACCO: MT-778
STATUS: DEVELOPMENTAL
DESCRIPTION: CASED, REDR BUR FOR BEST 327391
BLEND: BW-6068 THE BURLEY IS REDRIED

MOISTURE TARGETS: FINAL (IN CIGARETTE).............. 15.00 %
FLUE CURED STRIPS IN BULKER....... 21.00 %
BURLEY STRIPS IN STRIPS BULKER..... 30.00 %
REDRIED ONLY- ENTER REDRYER........ 30.00 %
AT COOLING SECTION... 5.00 %
EXIT REDRYER........ 15.00 %

STRIP MOISTURE GAINS FROM MAKE-UP USING HOTANTIS:
FLUE CURED STACK.................. 2.00 %
BURLEY STACK...................... 2.00 %
TOP DRESSING STACK................ 2.00 %

	STRIPS APPLIED TO: FLUE	ORNT	BRLY	RECON	APPL. RATE (LBS/M LBS)	YIELD M.C.%	APPL. TEMP.	STEAM TEMP.
FC CASINGS NONE								
BURLEY CASINGS								
C-726			X		422.71	26.7	180	0
MGC-817			X		36.42	28.0	120	0
HOTANTIS			X		41.68	30.0	180	160

CUT TOBACCO: MT-778 GRADE BIN: 979 UNCASED BURLEY, BEST SAMPLE 327277

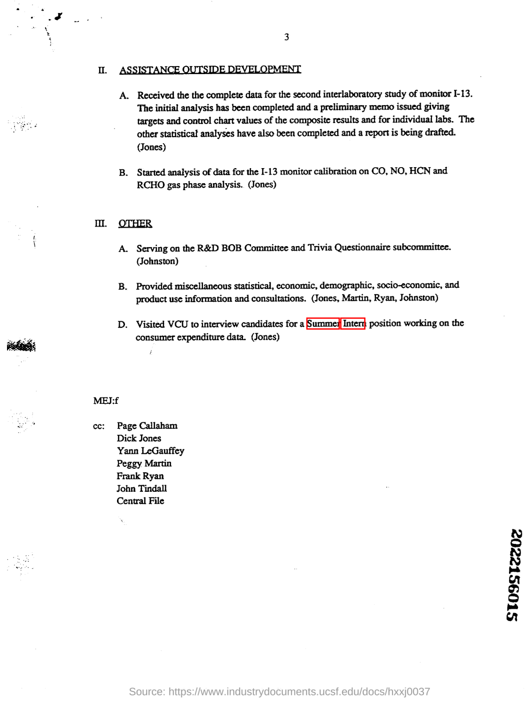
-- FLUE CURED -- GRADE BIN	%	-- BURLEY -- GRADE BIN	%	-- ORIENTAL -- GRADE BIN	%	-- RECONSTITUTED -- GRADE BIN	%
		979 BSTU	100.0				
TOTAL	0.0	TOTAL	100.0	TOTAL	0.0	TOTAL	0.0

GRADE BIN	GRADE	PERCENT	LBS AT 12.5 %	SOLID LBS	LBS AT TARGET
979 BSTU	B1B	38.10	3,848.1	3,367.1	3,961.3 @ 15.0 %
	B2B	28.92	2,920.9	2,555.8	3,006.8
	CB2B	5.41	546.4	478.1	562.5
	R1B	8.11	819.1	716.7	843.2
	WB1B	12.16	1,228.2	1,074.6	1,264.3
	WB3B	7.30	737.3	645.1	759.0
			10,100.0	8,837.5	10,397.1

THE ABOVE GRADES CAN BE STAGED ACCORDING TO THE FOLLOWING FORMULA:

GRADE	LBS PER UNIT	UNITS REQUIRED	TOTAL POUNDS	REQUIRED POUNDS	ADDED POUNDS
B1B	962	4	3848	3848	0
B2B	962	3	2886	2921	35
CB2B	962	0	0	546	546
R1B	962	0	0	819	819
WB1B	400	3	1200	1228	28
WB3B	400	1	400	737	337

Source: https://www.industrydocuments.ucsf.edu/docs/lycj0037 583232686



3

II. ASSISTANCE OUTSIDE DEVELOPMENT

A. Received the the complete data for the second interlaboratory study of monitor I-13. The initial analysis has been completed and a preliminary memo issued giving targets and control chart values of the composite results and for individual labs. The other statistical analyses have also been completed and a report is being drafted. (Jones)

B. Started analysis of data for the I-13 monitor calibration on CO, NO, HCN and RCHO gas phase analysis. (Jones)

III. OTHER

A. Serving on the R&D BOB Committee and Trivia Questionnaire subcommittee. (Johnston)

B. Provided miscellaneous statistical, economic, demographic, socio-economic, and product use information and consultations. (Jones, Martin, Ryan, Johnston)

D. Visited VCU to interview candidates for a Summer Intern position working on the consumer expenditure data. (Jones)

MEJ:f

cc: Page Callaham
Dick Jones
Yann LeGauffey
Peggy Martin
Frank Ryan
John Tindall
Central File

2022156015

Source: https://www.industrydocuments.ucsf.edu/docs/hxxj0037

Question: What is the yield M.C % for MGC-817?

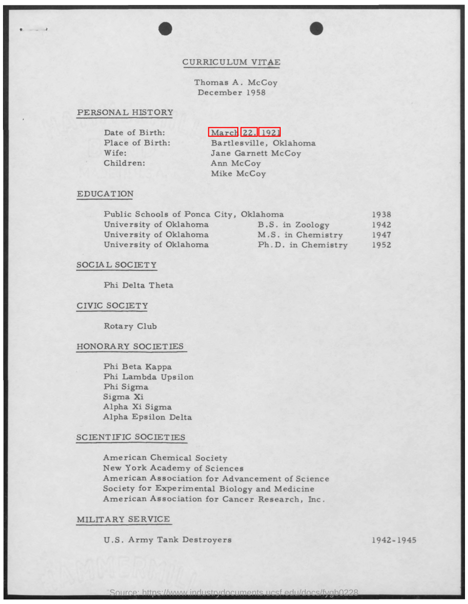
ViTLP output: {["28.0", [728, 434, 770, 446]]}

Ground-truth: {"28.0" or "28.0%"}

Question: For which position were the interviews conducted?

ViTLP output: {["Summer", [550, 457, 609, 472]], ["Intern" , [612, 457, 656, 472]]}

Ground-truth: {"Summer Intern Position"}



CURRICULUM VITAE

Thomas A. McCoy
December 1958

PERSONAL HISTORY

Date of Birth: March 22, 1921
Place of Birth: Bartlesville, Oklahoma
Wife: Jane Garnett McCoy
Children: Ann McCoy
Mike McCoy

EDUCATION

Public Schools of Ponca City, Oklahoma		1938
University of Oklahoma	B.S. in Zoology	1942
University of Oklahoma	M.S. in Chemistry	1947
University of Oklahoma	Ph.D. in Chemistry	1952

SOCIAL SOCIETY

Phi Delta Theta

CIVIC SOCIETY

Rotary Club

HONORARY SOCIETIES

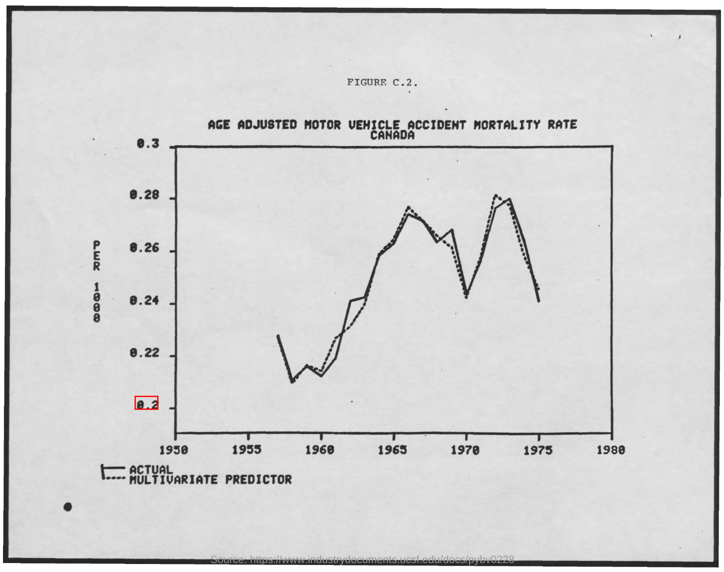
Phi Beta Kappa
Phi Lambda Upsilon
Phi Sigma
Sigma Xi
Alpha Xi Sigma
Alpha Epsilon Delta

SCIENTIFIC SOCIETIES

American Chemical Society
New York Academy of Sciences
American Association for Advancement of Science
Society for Experimental Biology and Medicine
American Association for Cancer Research, Inc.

MILITARY SERVICE

U.S. Army Tank Destroyers 1942-1945



Question: What is the date of the C. V.?

ViTLP output: {["March", [445, 207, 508, 221]], ["22" , [515, 207, 550, 221]], ["1921" , [554, 207, 600, 221]]}

Ground-truth: {"December 1958"}

Question: What is the 'actual' value per 1000, during the year 1975?

ViTLP output: {["0.2", [184, 696, 215, 718]]}

Ground-truth: {"0.24"}

Figure 8: Four examples (two successful cases & two failure cases) of ViTLP document VQA outputs with grounding locations.