DUBLIN - Document Understanding By Language-Image Network

Kriti Aggarwal[*], **Aditi Khandelwal**[*], **Kumar Tanmay**[*], **Owais Mohammed Khan, Qiang Liu, Monojit Choudhury, Hardik Hansrajbhai Chauhan, Subhojit Som, Vishrav Chaudhary, Saurabh Tiwary**
Microsoft Turing
{kragga, t-aditikh, t-ktanmay, owais.mohammed, qiangliu}@microsoft.com,
{monojitc, hachauhan, subhojit.som, vchaudhary, satiwary}@microsoft.com

Abstract

Visual document understanding is a complex task that involves analyzing both the text and the visual elements in document images. Existing models often rely on manual feature engineering or domain-specific pipelines, which limit their generalization ability across different document types and languages. In this paper, we propose DUBLIN, which is pretrained on web pages using three novel objectives: Masked Document Text Generation Task, Bounding Box Task, and Rendered Question Answering Task, that leverage both the spatial and semantic information in the document images. Our model achieves competitive or state-of-the-art results on several benchmarks, such as Web-Based Structural Reading Comprehension, Document Visual Question Answering, Key Information Extraction, Diagram Understanding, and Table Question Answering. In particular, we show that DUBLIN is the first pixel-based model to achieve an EM of 77.75 and F1 of 84.25 on the WebSRC dataset. We also show that our model outperforms the current pixel-based SOTA models on DocVQA, InfographicsVQA, OCR-VQA and AI2D datasets by 4.6%, 6.5%, 2.6% and 21%, respectively. We also achieve competitive performance on RVL-CDIP document classification. Moreover, we create new baselines for text-based datasets by rendering them as document images to promote research in this direction.

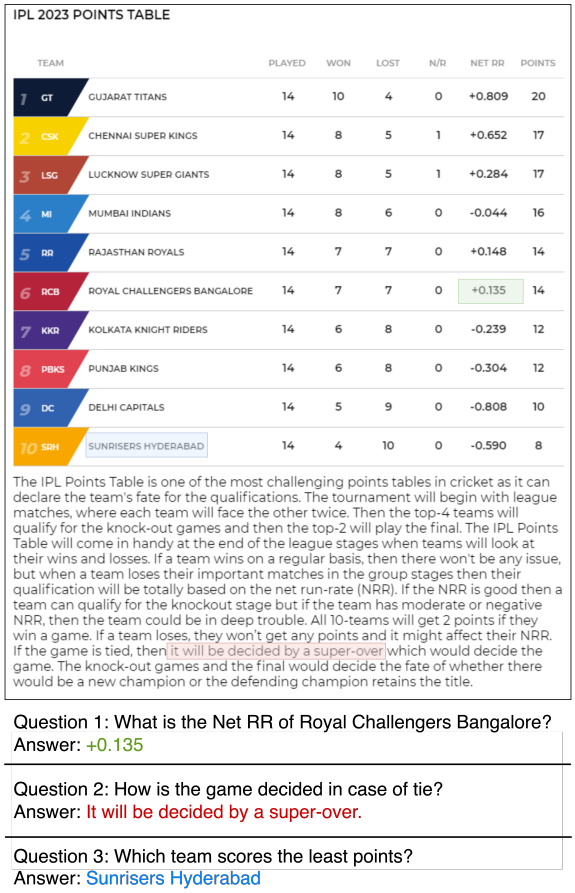

Figure 1: Example image to illustrate the Visual Question Answering Task. Dublin directly encodes both the input image and the textual question and decodes the output text (highlighted in the image with different colors).

1 Introduction

In today's digital era, the availability of vast amounts of information in various document formats has grown exponentially (Dong et al., 2014). These documents encompass a wide range, including scientific research papers, official reports, online articles and webpages, and PowerPoint presentations, holding valuable knowledge and insights, both in textual and visual forms.

A broad range of abilities is necessary for a thorough understanding of such a diverse range of documents, including the capacity to recognize text, comprehend language, and take into account various visual contexts (Lee et al., 2022).

Visual Document Understanding (VDU) models strive to enable machines to comprehend and interpret documents in a way that parallels human understanding. Humans naturally integrate various types of information, such as text, tables, charts,

[*] Equal contribution

figures, and diagrams, to form a coherent understanding when reading a document or examining a webpage. They effortlessly adjust their comprehension strategies based on the specific content, regardless of its format or structure. Developing models that can autonomously discern and process various elements and structures within a document without relying on explicit human-defined rules opens up new opportunities for intelligent information extraction, knowledge synthesis, and data-driven decision-making. This advancement in VDU improves search and retrieval capabilities for document images, facilitating easier access to relevant information within documents (Li et al., 2022; Klaiman and Lehne, 2021; Hong et al., 2022; Garncarek et al., 2021). VDU enhances accessibility for individuals with visual or reading impairments. It also facilitates task automation, including data entry, document verification, and document summarization, and can be applied to various industries such as legal document analysis, historical document, and more (Majumder et al., 2020; Tito et al., 2021; Antonacopoulos et al., 2011; Lombardi and Marinai, 2020; Schweighofer and Merkl, 1999).

However, most current document understanding models often rely on manual feature engineering or domain-specific processing pipelines, which limits their capability to handle diverse document types effectively. Traditionally, visual document understanding has relied on optical character recognition (OCR) techniques to extract textual information from document images (Xu et al., 2020, 2022; Sage et al., 2020; Katti et al., 2018; Majumder et al., 2020; Hwang et al., 2021b,c). However, OCR techniques can be error-prone, particularly for handwritten or non-Latin scripts, and fail to adequately capture the visual information present in the document images, such as graphics, tables, and charts (Taghva et al., 2006; Hwang et al., 2021a; Rijhwani et al., 2020). In contrast, VDU can provide a more comprehensive understanding of the document's content by analyzing both the textual and visual information in the document images.

Our work is motivated by the desire to bridge this gap and empower models to analyze documents and webpages with the same level of adaptability as humans. This paper presents a transformer-based encoder-decoder model known as DUBLIN. It serves as a versatile document understanding model that can effectively perform various downstream tasks. Figure 1 shows an illustration of the capabilities of DUBLIN. The model is trained on web pages, which provide a diverse range of data for training. To enhance the model's capabilities, it undergoes pretraining on three novel objectives. First, there is the Bounding Box Task, where the model predicts the location of a bounding box based on the word/sentence within the document and vice versa. Second, the Rendered Question Answering Task focuses on answering questions about the document given a specific query. Lastly, the Masked Document Content Generation Task involves generating the textual content of an entire document using a masked image of the document as context. Alongside these objectives, the model also undergoes standard multimodal pretraining tasks such as masked autoencoding. Our contributions are threefold:

1. We introduce three new pretraining tasks: Bounding Box Task, Rendered Question Answering Task, and Masked Document Content Generation Task for effective pretraining of multimodal document understanding models.

2. We demonstrate robust performance comparable to the state-of-the-art (SOTA) on a wide range of tasks, including Web-Based Structural Reading Comprehension, Document Visual Question Answering, Diagram Understanding, and Table Question Answering.

3. We create the baselines for text-based datasets by rendering them, setting a precedent as the first to undertake such an approach for Squad1.1 and WikiSQL datasets.

2 Related Works

The transformer architecture has been highly successful in the field of document understanding. The LayoutLM family of models sought to apply transformer models like BERT from natural language processing to document visuals. LayoutLMv1 (Xu et al., 2020) was pretrained like BERT (Devlin et al., 2019) but included 2D spatial positional information. LayoutLMv2 (Xu et al., 2022) improved the architecture by adding visual tokens and spatially biased attention to learn cross-modality interaction between visual and textual information. LayoutLMv3 (Huang et al., 2022) was pretrained with a crossmodal alignment objective to match text and image modalities by identifying missing image patches for text words. Models in the LayoutLM family have been evaluated without taking

text recognition into account, despite the fact that text recognition is an essential task. DocFormer introduced using only visual features extracted near text tokens spatially (Appalaraju et al., 2021). Ernie-Layout introduced reading order prediction as a correlative pretraining task for document reading order (Peng et al., 2022). TILT trained generative language models on labeled and unlabeled document data using generative training objectives (Powalski et al., 2021).

Recent advancements in document understanding have focused on self-supervised learning methods and multimodal embeddings. UDoc is a unified pretraining framework for document understanding that uses multimodal embeddings and self-supervised losses to learn joint representations for words and visual features from document images (Gu et al., 2022). SelfDoc, a task-agnostic pre-training framework for document image understanding, models positional, textual, and visual document components using coarse-grained multimodal inputs, cross-modal learning, and modality-adaptive attention (Li et al., 2021). UDOP, a foundation Document AI model, unifies text, image, and layout modalities with a Vision-Text-Layout Transformer and a prompt-based sequence generation scheme to enable document understanding, generation, and editing across diverse domains (Tang et al., 2023).

The above-described models depend on off-the-shelf OCR tools for text processing in documents. This reliance on OCR limits their applications and adds to computational expenses. Recent end-to-end models like Donut (Kim et al., 2022), Dessurt (Davis et al., 2022), and Pix2Struct (Lee et al., 2022) are image-to-text models that do not use OCR at inference time. Pix2struct is a pretrained image-to-text model for purely visual language understanding that can be fine-tuned on tasks containing visually-situated language (Lee et al., 2022). It was pretrained by learning to parse masked screenshots of web pages into simplified HTML and enables resolution flexibility to a variety of visual language domains. Matcha was proposed pretraining objectives to enhance the mathematical reasoning and chart derendering capability of visual language models (Liu et al., 2022).

3 Method

In this section, we present an in-depth exploration of DUBLIN, describing its architecture, the diverse set of pretraining objectives it is trained upon, and the extensive array of pretraining datasets utilized to bolster its learning process.

3.1 Model Architecture

DUBLIN is a novel end-to-end framework that combines the Bletchley (Mohammed et al., 2023) image encoder and the text decoder of InfoXLM (Chi et al., 2021). Bletchley is a multimodal model that employs a bootstrapping mechanism to train image and text encoders that can handle different modalities. InfoXLM is a cross-lingual model that learns a universal language representation that can handle diverse languages. We adopt Bletchley's image encoder and InfoXLM's text decoder as the initial weights for our model and then further pretrain them on various datasets using a combination of multi-task pretraining objectives and curriculum learning. The pretraining datasets comprise CCNews 200M, Google NQ Dataset, Bing QA, Rendered InfoXLM EN, Wikipedia and Synthetic TableQNA, which are detailed further in Section 3.3. We incorporate cross-attention layers between the image encoder and the text decoder to model the interaction between the visual and textual modalities. This enables the decoder to attend to pertinent regions in the image based on the query or context. The total number of trainable parameters in our model is 976M.

3.2 Pretraining Objectives

We present a novel pretraining framework with four different levels of objectives for effectively capturing the complex structures of visual documents. These objectives include tasks at the language, image, document structure, and question-answering levels. By focusing on multiple levels, our proposed framework is designed to enhance the model's ability to comprehend and reason about visual documents in a holistic manner. Figure 2 shows the generative pretraining tasks on which DUBLIN is trained. Overall, our proposed pretraining objectives provide a comprehensive and effective approach for training models to understand and reason about visual documents. Now, we describe in detail the pretraining objectives used in our model.

Masked Autoencoding Task. Inspired by ViT-MAE, we use the MAE task. We mask out 15% patch tokens of the image randomly in a similar fashion as was suggested in VIT-MAE (He

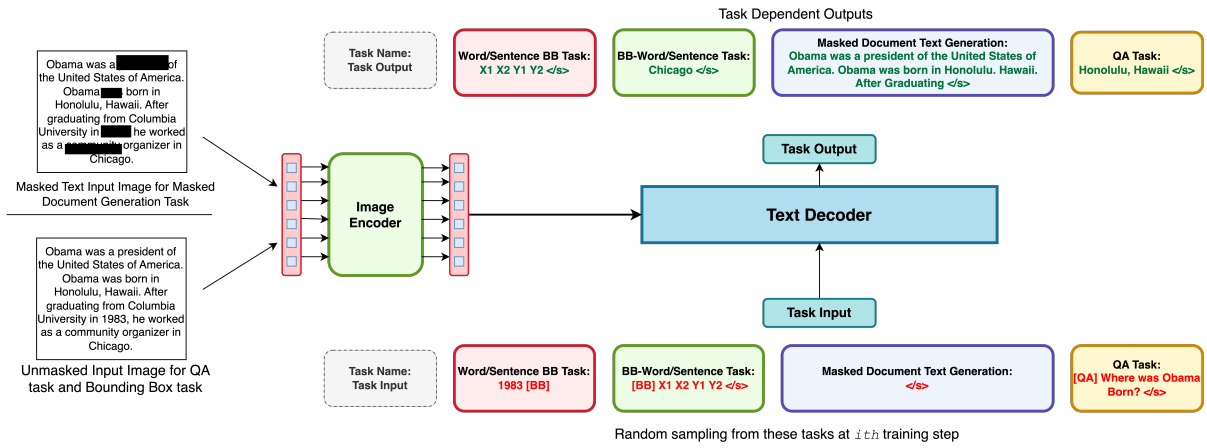

Figure 2: Illustration of three tasks in the DUBLIN pretraining framework: Bounding Box, Rendered QA, and Masked Document Text Generation.

et al., 2021). The task is to reconstruct the masked patches in the original image. We use 1-D fixed sinusoidal position embeddings to inject order information for the MAE task. The image encoder and decoder are trained using a normalized mean squared error (MSE) pixel reconstruction loss, which measures the difference between the normalized target image patches and the reconstructed patches. This loss is specifically calculated for the masked patches. For a better understanding, Figure 3 illustrates the MAE task, depicting the input image with masks and inverted predictions (inverted predictions are shown in the input image just for illustration and not added in the actual masked input image).

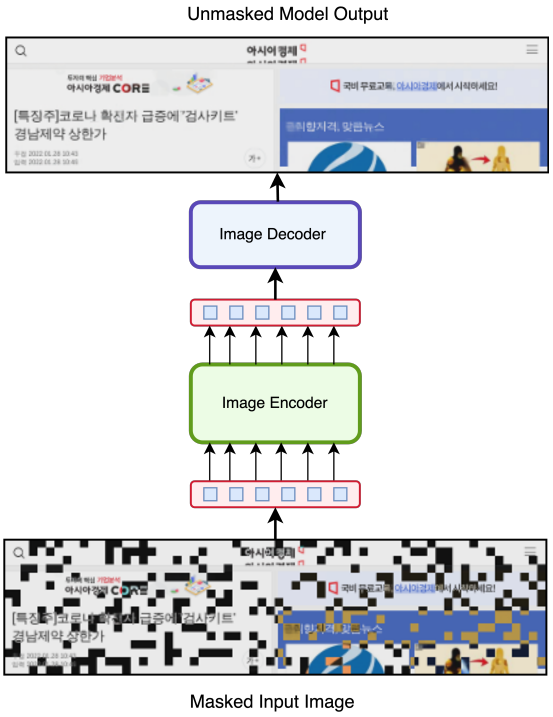

Figure 3: Illustration of the MAE task with the masked image with model predictions inverted to better understand the masked patches.

3.2.1 Generative Modeling Tasks

Masked Document Text Generation Task In this pretraining task, we randomly mask out some phrases from the text present in the input image before feeding it into the image encoder, similar to the approach in (Rust et al., 2023). The model is then trained to predict the entire document's textual content given this masked image as a context.. We employ two loss functions for this task. One is the OCR loss which is applied between the ground truth text sequence (for non-masked tokens) and the non-masked text tokens generated by the text decoder. Masked Language Modeling loss is applied between the masked text tokens in the input image and generated unmasked text tokens by the text decoder. Both the loss functions are cross-entropy; depending on which token it is, the mask takes care of it. In this way, DUBLIN learns the ability to understand the language given the visual information only, thereby bridging the gap between the language and visual modalities.

Bounding Box Task. We use this objective to predict the coordinates of the bounding box that encloses a given word/sentence in the document image and the word/sentence given the bounding box coordinates enclosing it. This way, the model learns the layout structure of the document image in a supervised fashion. We use the cross-entropy

loss function for this task. We use the special tokens [BB] and </s> for distinguishing this task from other generative modeling tasks.

Rendered Question Answering Task. We introduce this task specifically to aid the model in document image question answering. We employ a proprietary Bing QA dataset comprising passage and question-answer pairs. We render the passage containing the answer as an image and also render the question on top of this image. This image is then input to the image encoder; the question is used as a prefix for the text decoder. The decoder then generates the answer to the question. This supervised learning task helps the model to answer questions given the context image. We use the cross-entropy loss function for this task. We use a special token [QA] for this task. Besides the Bing QA dataset, we utilize Synthetic Table QA dataset, CCNews 200M dataset and Google NQ dataset for this task.

3.3 Pretraining Data

CCNews 200M We use this dataset to obtain document images, texts, and bounding box coordinates in various web domains and languages. This is done by scraping the URLs from the CCNews 200M dataset (Crawl, 2016) using the method outlined in CCNet (Wenzek et al., 2020) followed by rendering the HTML pages as screenshots and storing the document texts and their corresponding bounding boxes with the help of the Selenium library. We use samples from this dataset in all our pretraining tasks.

Google NQ Dataset This is a publicly available dataset (Kwiatkowski et al., 2019) based on open domain question answering. It contains around 307k training samples, along with the URL/webpage link for each sample. We scrape the webpage content using the HTML URLs. The webpage content is rendered as an image with the question added at the top. The question will also be used as a prefix for the decoder. We train our model on this dataset on the Rendered Question Answering task.

Bing QA Dataset We leverage Bing to obtain question-answer pairs along with their passage in English. We randomly sample question-answer pairs from Bing and render their passages and questions in a similar way as we did for the Google NQ dataset. In order to make our model generalization ability better over different kinds of texts, we render the text with random font size, color, and style using the Google Fonts library. We use this dataset for the Rendered QA task

Synthetic Table Structure QA Dataset In order to teach the model how to understand the table structure, we curate Synthetic Table QA dataset by randomly selecting 1 million webpages that contain tables and using Selenium to extract the HTML table elements from these webpages. To further enhance our training dataset, we perform data augmentation by employing five different CSS styles for rendering the HTML representation of each table as an image. These styles encompass various attributes such as border, font size, table separators, background, and text color. We devise this task of training the model to recognize table structure in the document images. During the training process, for each table, we randomly select one out of the five available styles. This ensured a diverse range of table appearances for our model to learn from. To generate synthetic questions and answers, we developed eleven distinct templates. These templates, reminiscent of SQL-like queries, were designed to reflect the content and format of the tables. An example template is as follows: "What is the value in the cell in the [column_name] column, where the row contains [row_content]?" Further elaboration on the templates and additional details can be found in Appendix B.

Model Pretraining. We pretrain our model for a total of 600k steps with different objectives and resolutions. We use curriculum learning to gradually increase the difficulty and complexity of the pretraining tasks and data. We use the XLM-RoBERTa tokenizer from the HuggingFace Transformers library and augment our vocabulary with special tokens such as [BB], [QA] and 1024 patch tokens. We use AdamW Optimizer with a learning rate of $1e^{-4}$, 10000 warmup steps, effective batch size of 1024 with low-resolution images and 256 with high-resolution images, weight decay of 0.01, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The pretraining procedure consists of four stages, with each stage adding new tasks/complexity to the training process. As tasks are added, we sample from both the newly added

task and the previous tasks at each step of training. In the first stage, we resize the input image to 224×224 and split it into fixed patches of 14×14 to feed to the image encoder. The combined MAE and document content generation tasks are randomly sampled at each step of training on low-resolution images from the CCNews 200M, GoogleNQ, and Rendered InfoXLM EN datasets (Chi et al., 2021) for 50k steps. In the second stage, we introduce the Rendered Question Answering Task using the Google NQ and Bing QA datasets for 350k steps at the same resolution. The third stage involves increasing the resolution to 896×896 for 55k steps while maintaining the same objectives. Finally, in the fourth stage, we add bounding box prediction objectives and continue training for another 150k steps on high-resolution images (896×896). We also include the Synthetic Table QA dataset in this stage.

4 Finetuning

We conduct comprehensive experiments on various types of documents, such as handwritten, typewritten, born-digital, scanned, infographics, diagrams, tables, forms, and webpages, to assess the model's generalization capability across diverse document varieties. We use the DUE-Benchmark datasets to measure our model's performance on different document understanding tasks, such as Visual Question Answering, Information Extraction and Table Question Answering/NLI. Moreover, we evaluate our model on other tasks, such as Document Classification, Question Answering over illustrations, UI understanding, and natural image understanding. We employ a methodology wherein we append the question/key, rendered as an image, onto the document image itself, following the Pix2Struct method. Subsequently, we utilize the question/key as a prefix for the text decoder.

Variable Resolution Finetuning. Pix2Struct paper addresses the aspect ratio distortion by rescaling input images either up or down to ensure the extraction of the maximum number of patches that fit within the designated sequence length. However, this resizing technique can lead to a potential loss of information when downsizing. In contrast, our approach focuses on preserving information by adopting a different resizing strategy that resizes the image to an aspect ratio which is an even power of 2 (e.g., 1, 4, 16, 64, etc.). By doing so, we maintain the desired aspect ratio while accommodating the maximum allowable number of patches (4096) within the given sequence length.

4.1 Downstream Tasks

Question Answering. To evaluate the effectiveness of our approach, we utilize DocVQA and InfographicsVQA from the DUE benchmark. These datasets allow us to assess the performance of our model in document question answering and question answering on infographics, respectively. Additionally, we evaluate our model's performance on the WebSRC dataset, which pertains to Web-based Structural Reading Comprehension. Furthermore, for question-answering tasks related to illustrations, we fine-tune our model using the ChartQA, AI2D, and OCR-VQA datasets. We then compare the performance of our model with a pure pixel-based baseline approach to determine its efficacy.

Key Information Extraction. We leverage the DeepForm dataset from the Due Benchmark for the key information extraction task. To accomplish this task, we overlay the extracted key information on top of the corresponding image and utilize it as a prefix for the text decoder.

Table Question Answering/NLI. To evaluate the model's performance on table question answering, we utilize the WikiTable Questions dataset from the DUE benchmark and the WikiSQL Question Answering dataset. However, since the WikiSQL dataset lacks table images, we address this limitation by rendering the table JSON as images in various styles. Additionally, for the fact verification task, we evaluate on the Tabfact dataset, which contains tables and requires a comprehensive understanding of the table content.

Document Classification. To evaluate our model's performance on document classification, we conduct finetuning experiment on the RVL-CDIP dataset (Harley et al., 2015). This dataset comprises scanned document images categorized into 16 classes, including letter, form, email, resume, memo, and more.

Model	Question Answering		Information Extraction	Table QA/NLI	
	DocVQA	InfoVQA	DeepForm	WTQ	TabFact
Metrics	ANLS	ANLS	F1	EM	Accuracy
Text-based Baselines					
$BERT_{large}$	67.5	-	-	-	-
$T5_{large}$	70.4	36.7	74.4	33.3	58.9
$T5_{large}$+U	76.3	37.1	82.9	38.1	76.0
Text+Layout Baselines					
$T5_{large}$+2D	69.8	39.2	74.0	30.8	58.0
$T5_{large}$+2D+U	81.0	46.1	83.3	43.3	78.6
Text+Pixels+Layout Baselines					
$LayoutLMv3_{large}$	83.4	45.1	84.0	45.7	78.1
UDOP	84.7	47.4	85.5	47.2	78.9
Pixels only					
$Pix2Struct_{large}$	76.7	40.0	-	-	-
$Dublin_{fixed\_res}$	78.2	36.8	62.2	25.7	**72.0**
$Dublin_{variable\_res}$	**80.3**	**42.6**	**64.0**	**28.6**	-

Table 1: Comparison with existing published models on the Due Benchmark.

4.2 Results

Table 1 displays the results of the evaluation of DUBLIN and other document understanding models on the DUE Benchmark. Our pixel-based model exhibits superior performance in Question Answering tasks on DocVQA and InfographicsVQA datasets. Infographics and DeepForm include images that possess extreme aspect ratio. Upon resizing the images to 896×896, the fixed resolution model encounters challenges in understanding the text present within the images owing to the reduction in resolution and loss of intricate details. Our methodology of incorporating variable aspect ratio has proven effective, as evidenced by significant improvements in DeepForm and a nearly 16% increase in the performance (ANLS) on Infographics compared to the fixed resolution model. Regarding performance on Table QA datasets, for WikiTable, the performance of our model is suboptimal. The main reason for this is attributed to the fact that the dataset comprises of questions that require a profound understanding of tables and complex operations that can be carried out on them. The operations in question involve aggregation and multistep reasoning, which are beyond the capabilities of our present model, as it has not been trained to execute these operations. In the context of Tabfact dataset, our model exhibits a competitive performance with the other models that utilize multiple modalities.

Model	QA over Illustrations		Classification
	AI2D	**OCR-VQA**	**RVL-CDIP**
Metrics	EM	EM	Accuracy
Donut	30.8	66.0	**95.3**
$Pix2Struct_{large}$	42.1	71.3	-
$Dublin_{fixed\_res}$	**51.1**	**73.1**	94.8

Table 2: Performance of our model QA over illustrations and Document Classification.

Table 2 shows the results of DUBLIN's performance on question answering over illustrations and document classification. The performance of our model on the AI2D dataset has resulted in an EM score of 51.1, which exceeds the performance of all previous pixel-based methods by a huge margin. The DUBLIN model has demonstrated state-of-the-art performance in OCR-VQA. This suggests that the model exhibits proficient comprehension of the illustrations within a document, reason about them, and answer questions. We also achieve competitive performance on document classification task well.

Table 3 shows DUBLIN's performance on rendered datasets and web-based structured reading comprehension task. Our model is the first pixel-based model to tackle question answering on the WebSRC dataset, which encompasses three types

Datasets	Metrics	Baseline	DUBLIN
WikiSQL	EM	89.2 (TAPEX)	75.26
Rendered Squad	EM/F1	85.1/91.8 (Bert)	79.69/87.07
WebSRC	EM/F1	81.66/86.24 (TIE)	77.75/84.25

Table 3: Our model's performance on rendered datasets and WebSRC.

of question-answering tasks: Comparison based QA, K-V Pair QA, and Table Question Answering. We have achieved impressive performance on this dataset, competitive to the current state-of-the-art (SoTA) model called TIE (Zhao et al., 2022). TIE, the existing SoTA model, employs a specialized pipeline specifically designed for the WebSRC dataset. It combines the Graph Attention Network (GAT) and a Pretrained Language Model (PLM) to leverage the topological information of logical structures and the spatial structures in the dataset. Our results demonstrate a strong baseline and establish a new state-of-the-art performance on the WebSRC benchmark for pixel-based models. Regarding Rendered Squad and WikiSQL, we have made significant contributions. We are the pioneers in rendering these datasets as images and creating strong baselines specifically tailored for pixel-based models.

5 Conclusion

We have presented DUBLIN, a novel framework for visual document understanding. DUBLIN is a transformer-based encoder-decoder model that can analyze both the text and the visual elements in document images. DUBLIN is pretrained on webpages using three novel objectives that capture the spatial and semantic relationships between the document elements. We have evaluated DUBLIN on several downstream tasks and shown that it achieves competitive or superior results compared to the state-of-the-art models. Our work shows that DUBLIN is a versatile and robust model for multimodal document understanding that does not rely on external OCR systems and can be finetuned easily in an end-to-end fashion. Our work opens up new possibilities for document analysis and understanding in various domains and applications.

References

A. Antonacopoulos, C. Clausner, C. Papadopoulos, and S. Pletschacher. 2011. Historical document layout analysis competition. In *2011 International Conference on Document Analysis and Recognition*, pages 1516–1520.

Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R. Manmatha. 2021. Docformer: End-to-end transformer for document understanding.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020. Tabfact: A large-scale dataset for table-based fact verification.

Xingyu Chen, Zihan Zhao, Lu Chen, Danyang Zhang, Jiabao Ji, Ao Luo, Yuxuan Xiong, and Kai Yu. 2021. Websrc: A dataset for web-based structural reading comprehension.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. Infoxlm: An information-theoretic framework for cross-lingual language model pre-training.

Common Crawl. 2016. News dataset available.

Brian Davis, Bryan Morse, Bryan Price, Chris Tensmeyer, Curtis Wigington, and Vlad Morariu. 2022. End-to-end document recognition and understanding with dessurt.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 601–610, New York, NY, USA. Association for Computing Machinery.

Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, Michał Turski, and Filip Graliń ski. 2021. LAMBERT: Layout-aware language modeling for information extraction. In *Document Analysis and Recognition – ICDAR 2021*, pages 532–547. Springer International Publishing.

Jiuxiang Gu, Jason Kuen, Vlad I. Morariu, Handong Zhao, Nikolaos Barmpalios, Rajiv Jain, Ani Nenkova, and Tong Sun. 2022. Unified pretraining framework for document understanding.

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

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2021. Masked autoencoders are scalable vision learners.

Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. 2022. Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking.

Wonseok Hwang, Hyunji Lee, Jinyeong Yim, Geewook Kim, and Minjoon Seo. 2021a. Cost-effective end-to-end information extraction for semi-structured document images.

Wonseok Hwang, Jinyeong Yim, Seunghyun Park, Sohee Yang, and Minjoon Seo. 2021b. Spatial dependency parsing for semi-structured document information extraction.

Wonseok Hwang, Jinyeong Yim, Seunghyun Park, Sohee Yang, and Minjoon Seo. 2021c. 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.

Anoop Raveendra Katti, Christian Reisswig, Cordula Guder, Sebastian Brarda, Steffen Bickel, Johannes Höhne, and Jean Baptiste Faddoul. 2018. Chargrid: Towards understanding 2d documents.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. 2016. A diagram is worth a dozen images.

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.

Shachar Klaiman and Marius Lehne. 2021. Docreader: Bounding-box free training of a document information extraction model.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*.

Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. 2022. Pix2struct: Screenshot parsing as pretraining for visual language understanding.

Junlong Li, Yiheng Xu, Lei Cui, and Furu Wei. 2022. Markuplm: Pre-training of text and markup language for visually-rich document understanding.

Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I. Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. 2021. Selfdoc: Self-supervised document representation learning.

Fangyu Liu, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Yasemin Altun, Nigel Collier, and Julian Martin Eisenschlos. 2022. Matcha: Enhancing visual language pretraining with math reasoning and chart derendering.

Francesco Lombardi and Simone Marinai. 2020. Deep learning for historical document analysis and recognition—a survey. *Journal of Imaging*, 6(10).

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.

Minesh Mathew, Viraj Bagal, Rubèn Pérez Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V Jawahar. 2021a. Infographicvqa.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. 2021b. Docvqa: A dataset for vqa on document images.

Owais Khan Mohammed, Kriti Aggarwal, Qiang Liu, Saksham Singhal, Johan Bjorck, and Subhojit Som. 2023. Bootstrapping a high quality multilingual multimodal dataset for bletchley. In *Proceedings of The 14th Asian Conference on Machine Learning*, volume 189 of *Proceedings of Machine Learning Research*, pages 738–753. PMLR.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables.

Qiming Peng, Yinxu Pan, Wenjin Wang, Bin Luo, Zhenyu Zhang, Zhengjie Huang, Teng Hu, 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.

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.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text.

Shruti Rijhwani, Antonios Anastasopoulos, and Graham Neubig. 2020. OCR Post Correction for Endangered Language Texts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5931–5942, Online. Association for Computational Linguistics.

Phillip Rust, Jonas F. Lotz, Emanuele Bugliarello, Elizabeth Salesky, Miryam de Lhoneux, and Desmond Elliott. 2023. Language modelling with pixels.

Clément Sage, Alex Aussem, Véronique Eglin, Haytham Elghazel, and Jérémy Espinas. 2020. End-to-end extraction of structured information from business documents with pointer-generator networks. In *Proceedings of the Fourth Workshop on Structured Prediction for NLP*, pages 43–52, Online. Association for Computational Linguistics.

Erich Schweighofer and Dieter Merkl. 1999. A learning technique for legal document analysis. In *Proceedings of the 7th International Conference on Artificial Intelligence and Law*, ICAIL '99, page 156–163, New York, NY, USA. Association for Computing Machinery.

Stacey Svetlichnaya. 2020. DeepForm: Understand Structured Documents at Scale — wandb.ai. https://wandb.ai/stacey/\deepform_v1/reports/DeepForm-Understand-Structured-Documents/-at-Scale--VmlldzoyODQ3Njg. [Accessed 15-May-2023].

Kazem Taghva, Russell Beckley, and Jeffrey Coombs. 2006. The effects of ocr error on the extraction of private information. In *Document Analysis Systems VII*, pages 348–357, Berlin, Heidelberg. Springer Berlin Heidelberg.

Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Cha Zhang, and Mohit Bansal. 2023. Unifying vision, text, and layout for universal document processing.

Rubèn Tito, Minesh Mathew, C. V. Jawahar, Ernest Valveny, and Dimosthenis Karatzas. 2021. Icdar 2021 competition on document visualquestion answering.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

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. 2022. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding.

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 *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1192–1200.

Zihan Zhao, Lu Chen, Ruisheng Cao, Hongshen Xu, Xingyu Chen, and Kai Yu. 2022. Tie: Topological information enhanced structural reading comprehension on web pages.

Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning.

APPENDIX

A Finetuning Datasets

DocVQA DocVQA dataset (Mathew et al., 2021b) focuses on question-answering tasks using single-page excerpts from real-world industry documents that include printed, handwritten and digital documents. The questions in this dataset often require understanding and processing various elements such as images, free text, tables, lists, forms, or a combination of these components.

InfographicsVQA The InfographicVQA dataset (Mathew et al., 2021a) contains questions that are specifically targeted at Infographics that can be found online. The inclusion of large images with extreme aspect ratios is one distinguishing feature of this dataset. Answering questions about visualized data found in a variety of Infographics is part of the task. The information needed to answer these questions can be presented using a variety of elements, including text, plots, graphs, or infographic layout components.

WebSRC WebSRC, also known as Web-based Structural Reading Comprehension, is a dataset consisting of 440,000 question-answer pairs (Chen et al., 2021). These pairs were collected from a diverse collection of 6,500 web pages. Each entry in the dataset includes not only the questions and answers but also the HTML source code, screenshots, and metadata associated with the respective web page. Answering questions in the WebSRC dataset requires a certain level of understanding of the structure of the web page. The answers can take the form of specific text excerpts, Key Information Extraction (KIE), or table question answering. To assess the performance on this dataset, we use metrics such as Exact Match (EM) and F1 score (F1). The training and development datasets are obtained using the official split provided by the authors. However, it's important to note that the authors have not released the testing set, so the results are solely based on the development set.

DeepForm We make use of the Key Information Extraction (KIE) dataset DeepForm (Svetlichnaya,

2020), which includes important election finance-related documents. The goal of this dataset is to extract crucial data from advertising disclosure forms submitted to the Federal Communications Commission (FCC), such as contract numbers, advertiser names, payment amounts, and air dates. Instead of the query, we provide the "Key" to the text decoder for the model to extract information from the image.

SQuAD1.1 To evaluate our model's extractive question-answering performance, we fine-tune it on the SQuAD dataset (Rajpurkar et al., 2016). We render this dataset as images on the fly, choosing a random font text, font style, etc., for each data point to maintain diversity and to test that, at inference time, the model is not biased toward answering questions from documents that all look a certain way but rather diverse in their fonts, styles, etc. The SQuAD dataset consists of over 100,000 question-answer pairs for over 500 articles. Given a question and its corresponding context paragraph, the task is to extract the span of text that contains the answer to the question. We follow the standard evaluation metrics for this dataset, including Exact Match (EM) and F1 score (F1), which measure the model's ability to output an answer that exactly matches the ground truth and its overlap with the ground truth, respectively. By evaluating this widely used benchmark, we can compare the performance of our model against the state-of-the-art approaches in extractive question answering.

WikiTable WikiTableQuestions dataset (Pasupat and Liang, 2015) utilized in this study focuses on question answering using semi-structured HTML tables obtained from Wikipedia. The authors specifically aimed to provide challenging questions that require multi-step reasoning on a series of entries within the given table, involving operations such as comparison and arithmetic calculations. We use the table images provided by the DUE Benchmark.

TabFact TabFact dataset includes entailed and refuted statements corresponding to a single row or cell to investigate fact verification using semi-structured evidence from clean and straightforward tables sourced from Wikipedia (Chen et al., 2020). Despite the task's binary classification nature, it presents challenges that go beyond simple categorization. The task requires sophisticated linguistic and symbolic reasoning to achieve high accuracy. We pass the table image to the image encoder and expect a binary output from the text decoder for this table fact verification task.

WikiSQL WikiSQL is a large crowd-sourced dataset consisting of 80,654 meticulously annotated examples of questions and corresponding SQL queries (Zhong et al., 2017). These examples are derived from 24,241 tables extracted from Wikipedia. This dataset mainly focuses on translating text to SQL. However, given our model's focus on answering questions based on documents, we transformed the denotations of this dataset into question-answer pairs in a natural language format. We rendered the tables as images by converting the table's JSON to HTML and then obtaining their screenshots in a similar fashion as described for the synthetic table structure QA dataset.

AI2D AI2 Diagrams (AI2D) is a comprehensive dataset consisting of over 5000 science diagrams typically found in grade school textbooks, along with more than 150,000 annotations, including ground truth syntactic parses and over 15,000 corresponding multiple choice questions (Kembhavi et al., 2016). The diagrams cover a wide range of scientific topics, such as geological processes, biological structures, and more. The multiple-choice questions are based on the science diagrams and are designed to test students' comprehension of the content. The dataset provides only train and test splits, with 1 percent of the train split set aside for validation.

B Synthetic Table Question Answering Dataset

Template	Example
What is the cell value in row [row_number] and column [column_number]?	What is the cell value in row 3 and column 2?
What is the cell value in column [column_number] and row [row_number]?	What is the cell value in column 7 and row 2?
What does the cell in the row [row_number] and column [column_number] contain?	What does the cell in row 4 and column 9 contain?
What does the cell in column [column_number] and row [row_number] contain?	What does the cell in column 1 and row 3 contain?
What is the cell value in column [column_name] and row [row_number]?	What is the cell value in column "Price" and row 4?
What is the value of cell where column is [column_name] and row number is [row_number]?	What is the value of cell where column is "Address" and row number is 9?
What is the value in the cell in [column ordinal] column where the row contains [row entry]?	What is the value in the cell in second column where the row contains "Mangoes"?
What is the value for [column 1st entries]?	What is the value for "City"?
How many rows are there in this table?	-
How many columns are there in this table?	-
What is the caption of the table?	-

Table 4: SQL-like query templates for generating QA pairs for the synthetic table-based question answering dataset.