TRANSFORMER-BASED APPROACH FOR DOCUMENT LAYOUT UNDERSTANDING

Huichen Yang, William Hsu

Department of Computer Science, Kansas State University
Manhattan, Kansas, USA

ABSTRACT

We present an end-to-end transformer-based framework named TRDLU for the task of Document Layout Understanding (DLU). DLU is the fundamental task to automatically understand document structures. To accurately detect content boxes and classify them into semantically meaningful classes from various formats of documents is still an open challenge. Recently, transformer-based detection neural networks have shown their capability over traditional convolutional-based methods in the object detection area. In this paper, we consider DLU as a detection task, and introduce TRDLU which integrates transformer-based vision backbone and transformer encoder-decoder as detection pipeline. TRDLU is only a visual feature-based framework, but its performance is even better than multi-modal feature-based models. To the best of our knowledge, this is the first study of employing a fully transformer-based framework in DLU tasks. We evaluated TRDLU on three different DLU benchmark datasets, each with strong baselines. TRDLU outperforms the current state-of-the-art methods on all of them.

***Index Terms*—** Document Layout Understanding, Vision Transformer, Object Detection, Document Structure Extraction

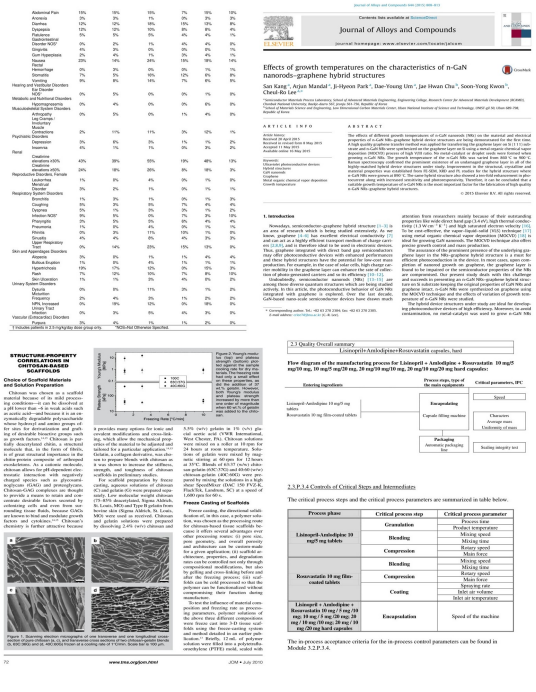

Fig. 1. Examples of complexity layouts of document image.

1. INTRODUCTION

Rapidly growing digital documents have become a key part of information transformation. However, due to the various layouts and the complex structures of documents, automatically structured analysis of documents is crucial to speed up the transformation process (Fig. 1). Document Layout Understanding (DLU) is a central step in automatic analysis, recognition of document structure, and information extraction out of document images. It leads to an important research direction for both Computer Vision (CV) and Natural Language Processing (NLP), and is a fundamental task of Document AI, which aims to automatically read, understand, and analyze documents. [1].

DLU plays an essential role in object detection tasks for document images to detect and recognize the fundamental components such as title, text body, figures, and tables in the document as objects. Some well-known deep learning-based object detection methods have been applied to DLU tasks, such as using a CNN-based neural network at pixel-level for document segmentation [2, 3] and Faster R-CNN based architecture for document layout detection [4]. Meanwhile, recent work introduces and integrates text, visual features, spatial features as the multi-modal model for DLU tasks [5, 6]. These additional information could help models obtain SOTA performance on relevant datasets. In this paper, we only use visual features for DLU tasks.

The attention-based transformer architecture has been widely employed in Natural Language Processing domain, and has been approved for its performance. It is also becoming increasingly attractive in recent object detection fields. Carion et al. [7] develop DETR which is the first transformer-based end-to-end object detection framework. In contrast to conventional one-stage and two-stage detection networks, it utilizes prediction methods which directly conduct bounding

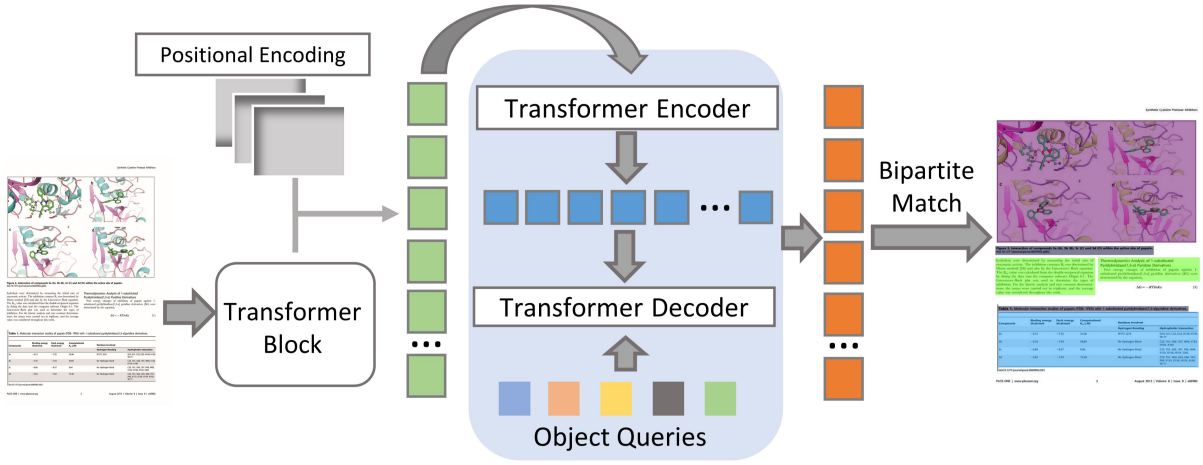

Fig. 2. The architecture of TRDLU.

box predictions with Hungarian bipartite matching, instead of using the anchor and non-maximum suppression (NMS) mechanism. However, the slow convergence and inaccuracy of small object performance of DETR make it inefficient. Zhu et al. [8] propose Deformable DETR based on DETR architecture which solves DETR's issue of slow convergence and high complexity, and they have introduced the idea of deformable convolution [9] to the attention module.

Given the impressive performance of the transformer in the CV field, it will be interesting to see if we can also take advantage of it in the DLU area. Therefore, we propose a fully transformer-based framework for document layout understanding, namely TRDLU. The TRDLU is an end-to-end DLU detector with vision transformer - Swin Transformer [10] as the backbone for feature extraction from the input image, and connect with transformer encoder-decoder for document layout detection and recognition. This study integrates the most recent work in the transformer of the object detection area and outperforms the previous transformer-based as well as CNN-based object detection frameworks [7, 8] for DLU task.

The main contributions of our paper are presented as follows: (i) this study is the first one to introduce a fully transformer-based detector pipeline for the task of DLU method, namely TRDLU; (ii) the proposed detector pipeline outperforms the previous transformer-based detector on DLU tasks, and is even better than the multi-modal feature-based detectors; (iii) the experiment results show that TRDLU outperforms the previous state of the art in DLU benchmark datasets.

2. METHODOLOGY

The overall TRDLU contains three main components: a transformer backbone, transformer encoder-decoder, and set prediction.The transformer backbone is used for visual feature extraction from the input images. The transformer encoder takes the feature in, and outputs the potential object features. The transformer decoder uses encoder outputs and object queries to generate final predictions for feed forward network (FFN). The final output will be generated by the set prediction process. The details of the detector pipeline are shown in Fig. 2.

2.1. Transformer Backbone

We use Swin Transformer [10] which is one of the state of the art architecture in the vision in transformer family as backbone for visual feature extraction from input images. Considering the input image is $H \times W \times 3$, Swin Transformer first splits the image into 4 non-overlapping patches as tokens with the patch splitting module. Then it sets the patch feature as a concatenation of the pixel values, and feeds it into the first stage of the two-stage module through a linear embedding layer, followed by two Swin Transformer blocks. Starting from the second stage, the patch features will be concatenated into 4C-dimensional by the first patch merging layer and converted into 2C-dimensional features with a linear layer. Finally, the feature transformation will be achieved by applying Swin Transformer blocks. The steps in stage 2 will be repeated in the rest of the stages. We use Swin tiny version (Swin-T) which has 4 stages as backbone, and the layer number of each stage is 2, 2, 6, and 2, respectively. The final feature output is $f \in \mathcal{R}^{\frac{H}{32} \times \frac{W}{32} \times 8C}$, where C represents the channel dimension. In addition, the position information is added into the feature map, flattened to spatial feature map $f \in \mathcal{R}^{N \times D}$, and sent to the multi-layer transformer encoder, similar to the Deformable DETR [9].

2.2. Transformer Detector

The novelty of this study is to combine merits of the most recent transformer-based works in CV, including the top-k object query [11], bounding box refinement and two-stage strategy [9], and auxiliary losses in encoder layer [12] to improve the performance in terms of accuracy and efficiency. This combination is integrated into the implementation of the transformer-based encoder-decoder detector which follows the structure in Deformable DETR.

Transformer encoder-decoder We construct the basic architecture of the transformer encoder-decoder following the structure in Deformable DETR [9].

Transformer encoder employs a multi-scale deformable attention module. The output of the previous layer is considered as the input of the current layer, which will be combined with the positional embedding as object queries. The deformable-attention reduces computational complexity by considering only the relevant keys for each query instead of every pair of queries and keys. Because of the decrease in computational complexity, we add auxiliary detection heads into the encoder layer, which will not increase the cost pressure, but improve the model performance.

Transformer decoder employs self-attention and multi-scale deformable attention modules which contain object queries as query elements. The reference point is predicted for each query and used for the multi-scale deformation attention model to extract image features. To optimize the model result, the detection head is applied to bounding box prediction to predict the deviations from the box center where the reference point was placed initially. Hence, this process facilitates the speed of model convergence.

Top-k object query The top-k object query mechanisms is introduced by Efficient DETR [11], where the encoder outputs can be used as decoder inputs and each of them is associated with an auxiliary detection head which computes a class score as a measurement of each output's objectness. The top-k encode outputs are then selected as the decoder queries based on the class score. We employ the top-k decoder query selection because it is identified to generate better results compared with the methods used in DETR [7] and Deformable DETR [9].

Bounding box refinement The implementation of bounding box refinement (BBR) follows the structure that is used in Deformable DETR [9]. The key idea of BBR is to refine the predicted bounding boxes by the current decoder layer based on the previous layer predictions. The predicted bounding box is represented by $b^d_{p\{\mathbf{x},y,w,h\}} \in \mathcal{R}$, where d is the decoder layer and p is the coordinator of prediction bounding box. The BBR process is repeatable from the first decoder layer to the last decoder layer. The final refinement result is returned by the last decoder layer. This iterative bounding box refinement mechanism can effectively improve detection performance.

Two-stage We apply the two-stage method which is introduced from two-stage Deformable DETR to our transformer detector. The object queries of the decoder layer in one-stage method are generated by predefined embeddings directly. Unlikely, the two-stage method first selects the top-k proposal boxes in the first stage based on their class scores, and feeds the selected boxes into the decoder and set positional embeddings of object queries as positional embeddings of region proposal coordinates during the bounding box refinement process. These object queries are more relevant to the current image. Following the two-stage Deformable DETR, we use multi-scale feature maps to generate anchors for each position and set the base anchor scale to be equal to 0.05. Then C (C is number of classes) category scores and 4 offsets per anchor are predicted by the detection head.

Loss function For the bounding box loss function, we use Distance Intersection over Union [13] with l_1 loss:

$$\mathcal{L}_{\text{box}}(b_{\sigma(\text{i})}, \hat{b}_{\text{i}}) = \lambda_{\text{diou}}\mathcal{L}_{\text{diou}}(b_{\sigma(\text{i})}, \hat{b}_{\text{i}}) + \lambda_{\text{L1}}||(b_{\sigma(\text{i})} - \hat{b}_{\text{i}})||_1 \quad (1)$$

where λ_{diou}, λ_{L1} are hyper-parameters, $\mathcal{L}_{\text{diou}}$ is the distance IoU loss. The Hungarian loss function is used to calculate the classification loss and bounding box regression loss between prediction and ground truth:

$$\mathcal{L}_{\text{Hungarian}}(\bar{y}, \hat{y}) = \sum_{i=1}^{N}\left[\mathcal{L}^{i,\hat{\sigma}(i)}_{class} + \mathbb{1}_{\{\bar{y}_i \neq \emptyset\}}\mathcal{L}^{i,\hat{\sigma}(i)}_{box}\right] \quad (2)$$

3. EXPERIMENT

We evaluate TRDLU on three different benchmark datasets. Two of them are document layout related datasets, and one is a table detection dataset. For fair comparisons, we use MS-COCO evaluation metric which is the same evaluation metric used by each benchmark. The code will be available at: https://github.com/huichentt/Transformer-DLU.

3.1. Benchmark Datasets

Scientific Literature Regions (SLR) is a synthesis dataset of DLU. It contains 1660 document images which are captured from three existing datasets: Article Regions [16], ICDAR-2013 [17], and GROTOAP [18]. This dataset includes 11 classes corresponding to the main regions of documents, including Title, Author, Address, Abstract, Keyword, Body, Figure, Table, Caption, Reference, and Text.

PubLayNet [19] is a large dataset for document layout analysis. The document layout is labeled by bounding boxes and polygonal segmentations. This dataset contains 360K document images and 5-region annotation classes: Title, List, Text, Figure, and Table. The ground truth of the test set is not released because the authors want to keep it for the competition. Therefore, we evaluate our model on the validation dataset.

Table 1. Detection results comparison on Scientific Literature Regions (SLR) and TNRC datasets.

Detector	Dataset	mAP	AP50	AP75	APs	APm	APl	AR
Faster R-CNN [4]	SLR	76.24	93.52	85.77	62.78	63.67	76.33	81.31
Cascade Mask R-CNN [4]	SLR	79.92	94.36	88.30	70.75	69.84	81.26	88.60
deformable_detr	SLR	80.61	95.50	88.50	58.70	66.90	83.30	87.70
TRDLU (ours)	SLR	**82.70**	**96.40**	**90.70**	**75.40**	**73.30**	**83.60**	**89.20**
deformable_detr [14]	TNRC	86.70	93.80	87.40	-	-	-	89.60
TRDLU(ours)	TNRC	**90.60**	**93.90**	**92.50**	-	-	-	**98.10**

Table 2. Detection result comparison on PubLayNet dataset.

Method	Text	Title	List	Table	Figure	mAP
VSR [5]	96.70	93.10	94.70	97.40	96.40	95.70
DocSgeTr [15]	89.90	73.60	89.50	97.50	96.60	89.40
TRDLU (ours)	95.82	92.13	**97.55**	**97.62**	**96.62**	**95.95**

TNCR [14] is a table detection dataset. It contains 9428 labels with 6612 document table images. This dataset includes 5 different classes to present the various table formats of scanned document images: No lines, Partial Lined, Merged Cells, Partial Lined Merged Cells, and Full lined.

3.2. Implementation Details

We use pre-trained Swin-Tiny Transformer [10] backbone network. The transformer includes 6 encoder and 6 decoder layers associated with the auxiliary detection head for each layer. The models are trained on Nvidia A40 GPU machine. We set batch size to 2 to train the models for 50 epochs on Scientific Literature Regions and for 30 epochs on TNCR datasets, respectively. The initial learning rate is set to 0.0002 and decays by 1/10 after the 40th and 25th epochs. For PubLayNet dataset, the model is trained by batch size 4 for 10 epochs with an initial learning rate of 0.0002 and decays by 1/10 after the 8th epoch. The rest hyperparameters are the same as those in Deformable DETR.

3.3. Performance comparison

We compare TRDLU on three DLU task-related benchmark datasets with the same tasks using state-of-the-art detection approaches. For SLR and TNCR datasets (Table 1), TRDLU outperforms all other methods and improves the mAP (mean average precision) up to 2.9 percent on SLR and 3.0 percent on TNCR. It also increases the AR (average recall) by 1.5 percent and 8.5 percent on SLR and TNCR, respectively. Table 2 shows the comparison results on PubLayNet. The TRDLU outperforms most other methods, and it is even better than the results using the VSR [5], a multi-modal framework.

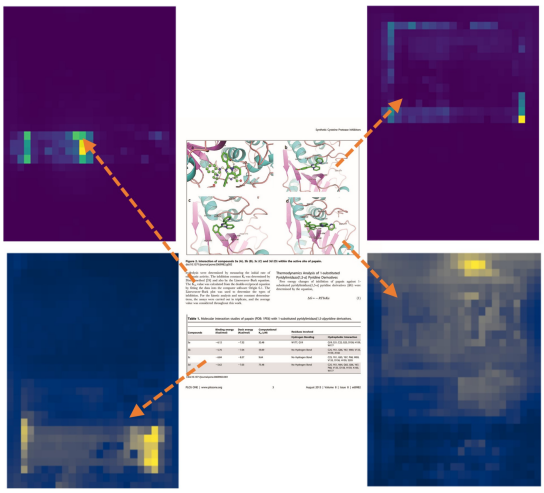

Fig. 3. Attention map visualization of TRDLU. The middle image is the input image. The two upper figures represent the decoder attention map, the lower two figures represent the encoder attention map.

3.4. Attention result analysis

Fig. 3 shows the attention map visualization results. The encoder could recognize the potential objects. It participates in the instance separation process, and gives the approximate object location. The decoder cloud gives the precise bounding boxes for different objects after model training. The attention visualization results can help us gain intuitions regarding how attention mechanisms work.

4. CONCLUSION

In this paper, we present an end-to-end transformer-based framework for document layout understanding, namely TRDLU. It integrates the merits of the most recent research works in this field. It is the first study of a fully transformer-based framework, and outperforms the experiential results generated by other research on both CNN-based and transformer-based frameworks.

5. REFERENCES

[1] Lei Cui, Yiheng Xu, Tengchao Lv, and Furu Wei, "Document ai: Benchmarks, models and applications," *arXiv preprint arXiv:2111.08609*, 2021.

[2] Xiao Yang, Ersin Yumer, Paul Asente, Mike Kraley, Daniel Kifer, and C Lee Giles, "Learning to extract semantic structure from documents using multimodal fully convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5315–5324.

[3] Sofia Ares Oliveira, Benoit Seguin, and Frederic Kaplan, "dhsegment: A generic deep-learning approach for document segmentation," in *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. IEEE, 2018, pp. 7–12.

[4] Huichen Yang and William H Hsu, "Vision-based layout detection from scientific literature using recurrent convolutional neural networks," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 6455–6462.

[5] Peng Zhang, Can Li, Liang Qiao, Zhanzhan Cheng, Shiliang Pu, Yi Niu, and Fei Wu, "Vsr: A unified framework for document layout analysis combining vision, semantics and relations," in *Document Analysis and Recognition – ICDAR 2021*, Josep Lladós, Daniel Lopresti, and Seiichi Uchida, Eds., Cham, 2021, pp. 115–130, Springer International Publishing.

[6] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha, "Docformer: End-to-end transformer for document understanding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 993–1003.

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*. Springer, 2020, pp. 213–229.

[8] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.

[9] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei, "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 764–773.

[10] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10012–10022.

[11] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang, "Efficient detr: improving end-to-end object detector with dense prior," *arXiv preprint arXiv:2104.01318*, 2021.

[12] Byungseok Roh, JaeWoong Shin, Wuhyun Shin, and Saehoon Kim, "Sparse detr: Efficient end-to-end object detection with learnable sparsity," *arXiv preprint arXiv:2111.14330*, 2021.

[13] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren, "Distance-iou loss: Faster and better learning for bounding box regression," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, vol. 34, pp. 12993–13000.

[14] Abdelrahman Abdallah, Alexander Berendeyev, Islam Nuradin, and Daniyar Nurseitov, "Tncr: Table net detection and classification dataset," *Neurocomputing*, vol. 473, pp. 79–97, 2022.

[15] Sanket Biswas, Ayan Banerjee, Josep Lladós, and Umapada Pal, "Docsegtr: An instance-level end-to-end document image segmentation transformer," *arXiv preprint arXiv:2201.11438*, 2022.

[16] Carlos Soto, "Visual detection with context for document layout analysis," Tech. Rep., Brookhaven National Lab.(BNL), Upton, NY (United States), 2019.

[17] Max Göbel, Tamir Hassan, Ermelinda Oro, and Giorgio Orsi, "Icdar 2013 table competition," in *2013 12th International Conference on Document Analysis and Recognition*. IEEE, 2013, pp. 1449–1453.

[18] Dominika Tkaczyk, Artur Czeczko, Krzysztof Rusek, Lukasz Bolikowski, and Roman Bogacewicz, "Grotoap: ground truth for open access publications," in *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries*, 2012, pp. 381–382.

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