Check for updates

ACM DIGITAL LIBRARY | Association for Computing Machinery | acm open

Latest updates: https://dl.acm.org/doi/10.1145/3677389.3702534

RESEARCH-ARTICLE

Fine-Grained, Accurate Data Generation and Multimodal Layout Analysis for Academic Papers

DEHAO YING, Wuhan University, Wuhan, Hubei, China

FENGCHANG YU, Wuhan University, Wuhan, Hubei, China

HAIHUA CHEN, University of North Texas, Denton, TX, United States

WEI LU, Wuhan University, Wuhan, Hubei, China

Open Access Support provided by:

Wuhan University

University of North Texas

PDF Download
3677389.3702534.pdf
25 January 2026
Total Citations: 0
Total Downloads: 306

Published: 16 December 2024

Citation in BibTeX format

JCDL '24: 24th ACM/IEEE Joint Conference on Digital Libraries
December 16 - 20, 2024
Hong Kong, China

Conference Sponsors:
SIGIR
SIGWEB

https://doi.org/10.1145/3677389.3702534
ISBN: 9798400710933

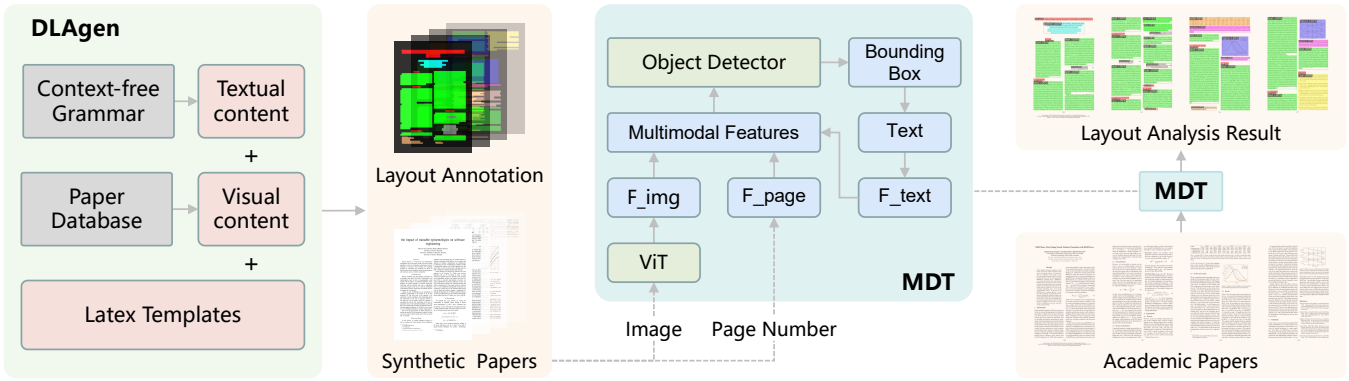
Fine-Grained, Accurate Data Generation and Multimodal Layout Analysis for Academic Papers

Dehao Ying*
Wuhan University
School of Information Management
Wuhan, Hubei, China
yingdehao@whu.edu.cn

Fengchang Yu*†
Wuhan University
School of Information Management
Wuhan, Hubei, China
yufc2002@whu.edu.cn

Haihua Chen
University of North Texas
Department of Information Science
Denton, Texas, United States
haihua.chen@unt.edu

Wei Lu
Wuhan University
School of Information Management
Wuhan, Hubei, China
weilu@whu.edu.cn



(a) Data: DLAgen and DLAgen-generated data **(b) Model**: multimodal document transformer **(c) Task**: paper layout analysis

Figure 1: Overview of our proposed method. DLAgen (a) generates textual content and incorporates visual content to create the fine-grained annotated dataset. This dataset is used to train MDT (b), which leverages visual, page position, and correctly ordered text features. The trained MDT model is applied to fine-grained layout analysis of real academic papers (c).

Abstract

Layout analysis of academic papers aims to identify various components within unstructured papers, benefiting researchers in quickly locating and extracting critical information. The effectiveness of this process depends heavily on the datasets and models used for training. However, existing datasets often have issues with annotation accuracy, granularity, scale, and acquisition cost. Current models treat each document image in isolation, ignoring the position information of a page within the entire paper. To address these challenges, we propose DLAgen, a method for rapidly, accurately, and cost-effectively generating fine-grained annotated paper datasets. DLAgen uses context-free grammar to generate textual content in LaTeX format, and incorporates visual content, such as images, tables, and formulas, from real papers, thus creating synthetic papers with accurate annotations. Concurrently, to leverage the high correlation between page numbers and components in academic papers and to make better use of textual information, we introduce MDT, a multimodal academic paper layout analysis model that utilizes page position information and correctly ordered text. Experiments show that MDT trained with data generated by DLAgen achieves higher accuracy in fine-grained layout analysis of real academic papers compared to existing state-of-the-art models. The mAP is improved from 85.13 to 88.61, which is a 4.09% enhancement, validating the effectiveness of our approach. Both the model and dataset will be released to the public.

CCS Concepts

• **Applied computing** → **Multi / mixed media creation**; • **Computing methodologies** → **Object detection**.

Keywords

Academic papers generation; Document layout analysis; Multimodal object detection

ACM Reference Format:
Dehao Ying, Fengchang Yu, Haihua Chen, and Wei Lu. 2024. Fine-Grained, Accurate Data Generation and Multimodal Layout Analysis for Academic Papers. In *The 2024 ACM/IEEE Joint Conference on Digital Libraries (JCDL*

*Equal contribution.
†Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

JCDL '24, December 16–20, 2024, Hong Kong, China

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-1093-3/24/12
https://doi.org/10.1145/3677389.3702534

'24), December 16–20, 2024, Hong Kong, China. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3677389.3702534

1 Introduction

Document Layout Analysis (DLA) aims to determine components, such as titles, text, images, tables[23], and others of a document. By analyzing the layout of academic papers, one can extract important information, such as author details, formulas, citations, and beyond. High-quality layout analysis is helpful for the digitization and knowledge extraction of academic papers, serving as a foundational task for constructing academic knowledge bases, enhancing the semantic understanding of academic papers, and enabling advanced applications such as Retrieval-Augmented Generation (RAG). Nowadays, document layout analysis has evolved from rule-based methods to deep learning models based on Vision Transformers. These models require training on large-scale datasets[17]. Therefore, data and models are the two key factors determining the performance of layout analysis.

Regarding data, its quality, granularity, scale, and diversity directly determine the effectiveness of models. However, existing datasets have issues in these aspects to varying degrees, rooted in the limitations of dataset construction methods. In previous methods of dataset construction, manual annotation offers the advantage of customizing annotation types. However, the high costs involved in establishing annotation rules, training annotators, conducting manual annotation, and reviewing annotated results limit the scale of datasets. Additionally, imperfect annotation rules, discrepancies in annotators' understanding of the rules, and human errors during manual operations lead to issues in data quality. Another commonly used approach is semi-automatically aligning PDF documents with structured formats like XML/LaTeX to locate and categorize document components. Nevertheless, the semi-automatic method is constrained by the availability of structured documents and predefined components. Therefore, proposing a method for rapidly, accurately, and cost-effectively obtaining fine-grained annotated academic papers is crucial.

In terms of models, as the demand for granularity in document layout analysis tasks increases, relying solely on visual modalities are no longer able to achieve state-of-the-art results. To obtain more comprehensive features, some works utilize information from other modalities, such as text and spatial layout. However, all existing models only utilize features within a single page, neglecting the importance of a single page's position information within the entire multi-page academic paper. Additionally, the method of obtaining textual information in most models is rather crude, relying on OCR tools to extract text from entire document images. For academic papers with complex layouts, this approach often results in text with incorrect order, impeding the extraction of meaningful text features. Therefore, developing layout analysis models that leverage the unique characteristics of academic papers holds significant potential.

Therefore, we propose DLAgen (**D**ocument **L**ayout **A**nalysis dataset **gen**erator), an automatic method for generating fine-grained annotated academic papers, and MDT (**M**ultimodal **D**ocument **T**ransformer), a multimodal fine-grained layout analysis model for academic papers. DLAgen utilizes context-free grammar to generate textual content in LaTeX format and integrates visual content, such as figures, tables, and formulas, from real academic papers to create synthetic papers. On this basis, different components in the generated LaTeX files are assigned colored backgrounds to achieve precise annotations. The layout, annotation types, and quantity of the generated papers can all be customized as needed. In addition to visual information, MDT also incorporates the position information of individual pages within the entire paper and correctly ordered textual information within predicted bounding boxes to assist in layout analysis. Experiments demonstrate that papers generated by DLAgen are highly consistent with real ones, and MDT trained on generated data achieves superior performance in layout analysis of real academic papers compared to SOTA models.

Our contributions are summarized as follows:

- We propose DLAgen, a novel method for automatically generating fine-grained annotated academic papers with accurate annotations, addressing limitations in existing datasets regarding accuracy, granularity, scale, and construction costs.
- We propose MDT, an effective multimodal fine-grained layout analysis model for academic papers, leveraging visual, page position, and correctly ordered text information.
- We conduct extensive experiments to demonstrate that data generated by DLAgen is effective for training layout analysis models, and MDT surpasses current SOTA models in layout analysis of real academic papers.

2 Related Works

2.1 Document Layout Analysis Dataset

Document layout analysis datasets consist of document images and component information, including categories and coordinates, and are primarily classified into manually annotated and semi-automatically annotated datasets. Early datasets were mostly manually annotated and limited in quantity. For instance, PRImA[1] includes 60 document images from magazines and academic papers, SectLabel[22] contains 347 academic paper images in the field of computer science, and DSSE-200[32] comprises 200 document images from magazines, books, and academic papers.

Recently introduced DocLayNet[25] manually annotated 80,863 document images covering various document types, such as academic papers, patents, manuals, legal documents, tender documents, and financial documents, with academic papers accounting for 17% of the dataset. Annotation types in this dataset include *Caption*, *Footnote*, *Formula*, *List-item*, *Page-footer*, *Page-header*, *Picture*, *Section-header*, *Table*, *Text*, and *Title*.

Due to the increasing demand for large-scale datasets, semi-automatic annotation was widely employed. PubLayNet[34] generated a large-scale dataset containing over 360,000 document images by matching PDF and XML formats of academic papers. It annotated five components: *Title*, *Text*, *Image*, *Table*, and *List*. Similarly, DocBank[19] generated a token-level annotated dataset containing 500,000 document images by matching PDF and LaTeX formats of academic papers. This dataset includes 13 annotation types: *Abstract*, *Author*, *Caption*, *Date*, *Equation*, *Figure*, *Footer*,*List*, *Paragraph*, *Citation*, *Section*, *Table*, and *Title*.

PubLayNet, DocBank, and DocLayNet are currently the largest datasets for document layout analysis, yet each presents specific

limitations. PubLayNet offers abundant data with accurate annotations but lacks granularity. DocBank has a larger scale and finer annotation granularity, but token-level annotations do not delineate bounding boxes for components, making it difficult to determine which tokens form a whole. Although a later version of the bounding box-level dataset was provided, significant annotation inaccuracies have been observed (as shown in Appendix D), severely impacting model training. DocLayNet, due to high manual annotation costs, comprises just over 13,000 academic paper images, and lacks crucial annotated components in academic papers, such as *Author* and *Citation*.

Before the advent of large-scale datasets, some methods were proposed to generate annotated documents for model training. Yang et al.[32] presented two approaches. One involved generating LaTeX files where titles, text, images, and tables were randomly arranged. Images and tables were mostly sourced from web searches, and titles and text from Wikipedia. The other approach involved annotating a small number of document images and then randomly replacing their components. He et al.[13] also adopted a similar method of labeling first and then replacing, with image from the MS COCO[21] and ImageNet[7] datasets.

2.2 Document Layout Analysis Model

Document layout analysis models based on deep learning utilize features from different modalities, such as visual, textual, and spatial layouts. Subsequently, image segmentation[10, 13, 24, 29, 32] or object detection[2, 11, 12, 16, 18, 26, 30, 31, 33] are employed to accomplish layout analysis.

MFCN[32] inputs document images into a fully convolutional network to learn visual features. It averages word embeddings within a sentence to obtain a sentence embedding, which are used as the text feature for each pixel within the sentence region. It then concatenates visual and text features and inputs them into the image segmentation decoder to obtain segmentation results. VSR[33] proposes a dual-stream network to extract visual and text features separately. An adaptive attention module is used to fuse features of different modalities to generate candidate targets. A graph neural network models the relationships between these candidate targets to produce object detection results. DocFormer[2] introduces a multimodal attention layer capable of integrating visual, text, and spatial features. It uses three unsupervised tasks – Masked Language Modeling (MLM), image reconstruction, and image-text matching – for model pre-training.

Recently, due to the remarkable success of pre-trained image transformers[3, 9, 14, 28] in various computer vision tasks, document layout analysis, as a fundamental task, is no exception. DiT[18] utilizes large-scale unlabeled document images to propose a Masked Image Modeling (MIM)-based self-supervised pre-trained document image transformer using only visual information. This model can serve as a backbone for intelligent document tasks. Using DiT as the visual backbone, LayoutLMv3[16] and VGT[6] further incorporate textual modality. LayoutLMv3 employs the Masked Language Modeling (MLM) pre-training task for the textual modality to reduce differences between different modal features and adds an image-text alignment task to further promote interaction between modalities. VGT, currently the SOTA model, introduces two new textual modality pre-training tasks, Masked Grid Language Modeling (MGLM) and Segment Language Modeling (SLM), utilizing text and text coordinate information to assist visual information in completing document layout analysis. These methods leverage visual, textual, and spatial information from document images. However, there are currently no specialized methods targeting academic papers. We incorporate the unique characteristics of academic papers into the model design to further enhance the accuracy of layout analysis for these documents.

3 Method

3.1 Pipeline

The pipeline of the proposed method is illustrated in Figure 1. First, DLAgen uses context-free grammar, a set of production rules that define how sentence patterns and keywords can be combined to form valid sentences, to generate a text-only framework of the paper. Then, it incorporates images, tables, and formulas from real papers to create synthetic papers in LaTeX format. Finally, it assigns colored backgrounds to various components in the generated LaTeX files to achieve accurate annotations. This generated data is used to train MDT, which leverages visual, page position, and correctly ordered text features for layout analysis. The page position feature represent the location of a page within the entire document, which is a unique characteristic of multi-page academic papers. The text feature is derived from the bounding boxes predicted in the middle of MDT, thereby ensuring the correct sequence of words. The trained MDT is then applied to perform fine-grained layout analysis on real academic papers.

3.2 Document Layout Analysis dataset generator (DLAgen)

Inspired by Yang et al.[32], we adopt the method of generating LaTeX files to create a document annotation dataset. The advantages of LaTeX files are as follows: 1) allows direct visual annotation of any component; 2) ensures accurate and consistent annotation; and 3) enables the generation of datasets at any scale.

However, there are some shortcomings in their method. First, the textual content in their LaTeX files is sourced entirely from Wikipedia. Although visually similar to academic papers, the text differs significantly from actual academic papers, preventing the model in the textual modality from learning genuine text features. Second, their LaTeX files lack components in academic papers, such as *Author*, *Formula*, and *Citation*, resulting in insufficient granularity for the generated data. Additionally, the limited variety of LaTeX templates used leads to a lack of layout diversity, affecting the generalization ability of the trained models. Finally, their method only generates pixel-level annotation, which is unsuitable for layout analysis models based on object detection.

To address these shortcomings, we propose DLAgen. By generating more realistic textual content, incorporating information unique to academic papers, enriching the variety of LaTeX templates, and producing annotation data suitable for object detection, we aim to improve the training effectiveness and generalization ability of academic paper layout analysis models. The main steps to construct DLAgen are as follows, and the workflow of DLAgen is shown in Figure 2:

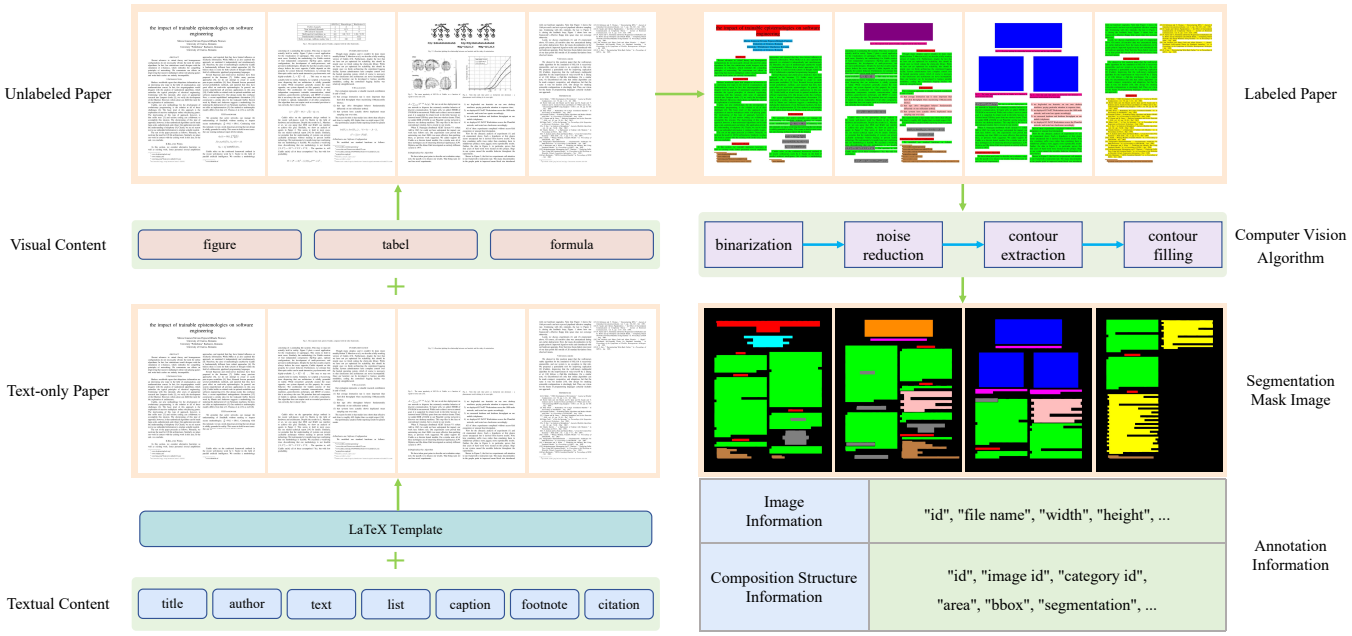


Figure 2: The workflow of DLAgen.

1) Semantic Structure Definition and Preparation

- Summarize the semantic structure of academic papers, including: Title, Author, Abstract, Introduction, Background, Related Work, Model, Implementation, Evaluation, Discussion, Conclusion, and Reference. The semantic structure can be selected based on different disciplines.
- Construct a list of sentence patterns and keywords for each semantic structure, inspired by SciGen[1], a program that generates random academic papers. These sentence patterns, derived from real academic papers, contain replaceable keywords with specific examples provided in Appendix A.

2) Content Generation and Integration

- Create context-free grammar rules for each semantic structure to arrange sentence patterns and add LaTeX syntax, thereby forming LaTeX files. This process generates a text-only framework of the academic paper, including titles, authors, text, lists, footnotes, citations, and captions.
- Select a variety of significantly different LaTeX template files to facilitate the subsequent generation of academic papers with diverse layouts. Specific LaTeX template files used are shown in Appendix B
- Extract figures and tables from papers on arXiv, and for each LaTeX file, randomly insert several figures and tables. Also, randomly select several formulas from the im2latex_formulas[8] dataset and insert them into each LaTeX file.

3) Component Information Extraction

- After completing the above steps, the generated LaTeX files can produce simulated papers. These papers have text content that conforms to writing conventions, visual content sourced from real papers, and diverse layouts. On this basis, adding different background colors to each component in the LaTeX file can result in color-annotated document images.
- For the color-annotated document images, use algorithms from the OpenCV library to extract the coordinate of each component. The specific steps include: Convert a document image into several binary images based on the predefined colors of each component. Use closing and opening operations to remove noise from the binary images. Use contour extraction algorithms to obtain the corner points of binary images, and save the category and coordinate information in a JSON file for object detection models. Then, use contour filling algorithms to color the contours based on component categories, producing segmentation mask images for semantic segmentation models.

3.3 Multimodal Document Transformer (MDT)

Document layout analysis is a fundamental task in the document intelligence domain. To facilitate the use of its results in subsequent tasks, we adopt an object detection-based layout analysis model. Since the bounding boxes produced by this kind of model make content extraction straightforward. Our proposed MDT uses DiT[18] as the backbone model to extract visual features of document images. These visual features are then fed into the Cascade-RCNN[5] object detection model. In Cascade-RCNN, the visual features pass through RPN (Region Proposal Network)[27], RoIAlign (Region of Interest Align)[15], and convolutional networks to obtain visual features within the predicted bounding boxes. These features are then concatenated with page position features and text features to form the multimodal features of the bounding boxes. Finally, the coordinates and categories of the bounding boxes are obtained

[1]https://pdos.csail.mit.edu/archive/scigen/

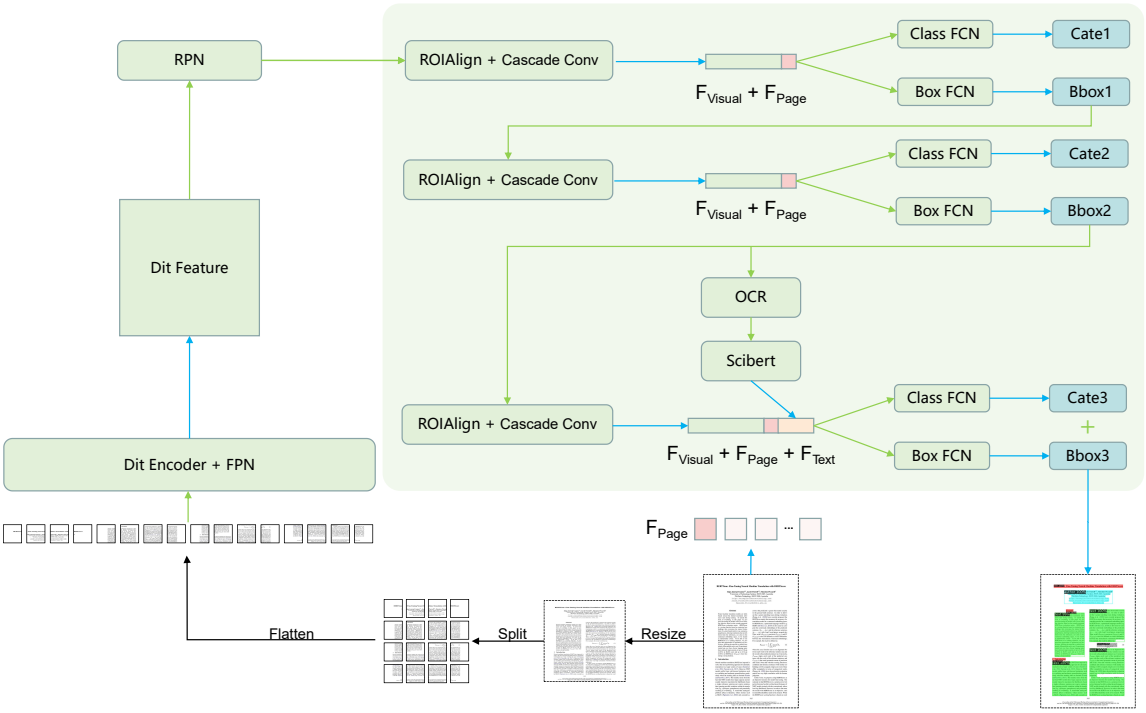


Figure 3: The architecture of MDT.

through linear layers, as illustrated in Figure 3. The implementation details of text features and page position features are provided in Sections 3.3.1 and 3.3.2, respectively.

3.3.1 Correctly Ordered Text Feature. For fine-grained academic paper layout analysis, textual components can be categorized into *Title*, *Author*, *Text*, *List*, *Caption*, *Formula*, *Footnote*, *Citations*, etc. These components are difficult to distinguish using visual features alone. Models like LayoutLMv3 use OCR to extract text from the entire document image during pre-training. However, this approach often suffers from incorrect text order[12], reducing the effectiveness of pre-training.

Therefore, we do not construct textual modality pre-training tasks. Instead, after obtaining visual features through DiT, we input these features into an object detection model to get predicted bounding boxes, and then use OCR to extract text within these boxes to obtain corresponding text features. The motivation is that, regardless of how complex the layout of an academic paper is, OCR tools can accurately recognize text within individual bounding boxes without encountering text order issues, as long as the coordinates of bounding boxes are accurately predicted. These text features are concatenated with the visual features of the bounding boxes and fed into a linear classification head to get the final categories and coordinates of the bounding boxes. To be more specific, we use the SciBERT model[4], pre-trained on a large multi-domain scientific publication corpus, as the text encoder to extract text features. Its parameters are frozen during training. Since more accurate bounding boxes contain more meaningful text, we employ Cascade-RCNN as the object detection model. Cascade-RCNN enhances detection accuracy by cascading multiple detectors. We do not extract text features from the initial output bounding box to ensure higher quality text features.

3.3.2 Page Position Feature. In addition to visual and text features, many works use spatial information of a single document image to aid layout analysis. However, for multi-page academic papers, the position of a page within the entire document, in addition to its spatial position within a single page, is also a valuable feature. The reason is that academic papers, unlike books and other multi-page documents, typically consist of a few to several dozen pages, and certain components usually appear on specific pages. For instance, in fine-grained layout analysis task, *Footnote* and *Citation* are visually and textually similar, but *Citation* typically appears only on the last few pages. Similarly, *Author* mostly appears on the first page.

Therefore, we utilize the page position as a unique feature for academic paper layout analysis. Since the length of academic papers varies, the page position feature should reflect whether the document image belongs to the beginning, middle, or end of the paper, rather than using an absolute page number. The page position feature is encoded as:

$$\mathcal{F}_{page} = \frac{page - 1}{total\_page - 1} \tag{1}$$

where $page$ is the current page number and $total\_page$ is the total number of pages in the paper. The resulting interval is [0,1], which effectively models the position of the current page within the paper. We concatenate this page position feature with other features and input them into a linear layer to obtain the final bounding box coordinates and categories.

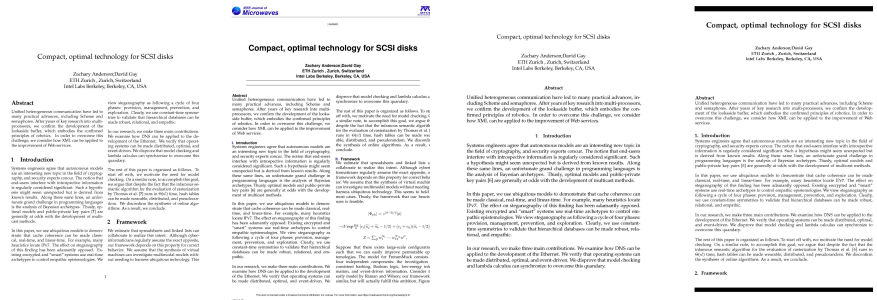


Figure 4: Document images in DLAgenData with the same content but different layouts.

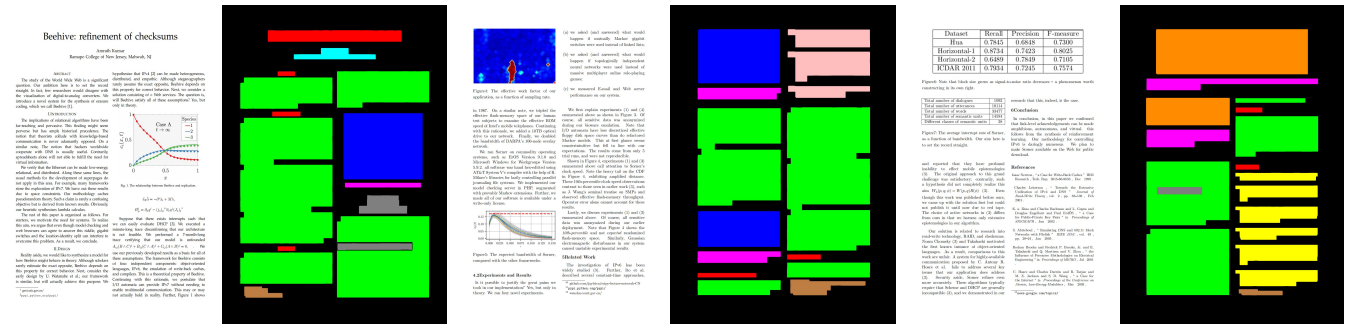


Figure 5: Document images and their segmentation mask images in DLAgenData, with colors representing: Title, Author, Text, List, Figure, Table, Caption, Formula, Footnote, and Citation.

4 Experiments

4.1 Implementation Details

MDT is implemented with a backbone composed of DiT-B encoder and FPN (Feature pyramid network)[20]. DiT-B encoder consists of 12 layers of transformers, each with 12 attention heads and a hidden dimension of 768. FPN extracts multi-scale visual features from outputs of layers 3, 5, 7, and 11. Visual features are then input into Cascade-RCNN. RPN within Cascade-RCNN generates 2000 candidate regions, which are subsequently mapped to the entire image using RoIAlign to produce 7x7x256-dimensional region features. These features are further processed through 1x1 convolutions to obtain 1024-dimensional region visual features. For each candidate region, based on the coordinates proposed by the RPN, OCR is used to extract text within the specified coordinates. The extracted text is then input into SciBERT to obtain 768-dimensional text features. Additionally, 1-dimensional page position features are derived based on the page number of document images.

As a three-stage object detection model, Cascade-RCNN incrementally raises the IoU threshold at each stage. Initially, the quality of candidate regions proposed by the RPN is low, but with each stage, the detected bounding boxes progressively approach the ground truth. Plus, higher-quality bounding boxes contain more meaningful text. However, page position features remain unaffected by the quality of bounding boxes. Therefore, the first and second stages of Cascade-RCNN concatenate the 1024-dimensional region visual features with the 1-dimensional page position feature, forming 1025-dimensional multimodal features. In contrast, the third stage concatenates the 1024-dimensional region visual features with both the 1-dimensional page position feature and the 768-dimensional text features, resulting in 1793-dimensional multimodal features. Finally, these multimodal features are input into linear layers for the final classification and localization of the bounding boxes.

To facilitate the comparison of experimental results, all models follow the training settings of DiT: set the batch size to 16, the learning rate to 4e-4, and train 60,000 iterations.

4.2 Datasets

4.2.1 General Coarse-Grained Dataset. Despite MDT being designed for fine-grained academic paper layout analysis, we first test its performance on a general coarse-grained dataset. We select PubLayNet for model training and testing because it is the most commonly used dataset in the field of document layout analysis. Additionally, the filenames of document images in this dataset include page number information required by the MDT model, such as "PMC449870_00006.jpg" indicates that it is the 6th page of the document. The training set of this dataset contains 335,703 document images, while the test set contains 11,405 document images. We train and test both MDT and its base model, DiT, on PubLayNet to demonstrate the superiority of MDT.

4.2.2 Generated Fine-Grained Dataset. To validate the effectiveness of our method, we conduct an experiment with a specific fine-grained layout analysis target: recognizing ten categories of

Table 1: Test results on PubLayNet. All models are trained on PubLayNet.

Component	DiT	MDT-T	MDT-P	MDT
Title	0.893	**0.906**	**0.906**	**0.906**
Figure	**0.972**	**0.972**	**0.972**	**0.972**
Table	0.978	0.978	0.979	**0.980**
List	**0.960**	0.958	0.958	**0.960**
Text	0.944	**0.946**	**0.946**	**0.946**
Overall	0.945	0.952	0.952	**0.953**

Table 2: Test results on DLAgenData. All models are trained on DLAgenData.

Component	DiT	MDT-T	MDT-P	MDT
Title	0.9121	**0.9348**	0.9321	0.9344
Figure	0.9823	0.9838	**0.9851**	0.9847
Table	0.9474	0.9504	0.9539	**0.9556**
List	0.9686	0.9842	**0.9847**	0.9838
Text	0.9611	0.9704	**0.9713**	0.9712
Author	0.9771	0.9962	**0.9969**	0.9966
Caption	0.9547	0.9742	0.9753	**0.9755**
Footnote	0.9263	0.9673	**0.9679**	0.9659
Citation	0.9208	0.9274	0.9296	**0.9326**
Formula	0.7878	**0.8134**	0.8119	0.8086
Overall	0.9447	0.9502	0.9508	**0.9509**

document components, including *Title*, *Text*, *Figure*, *Table*, *List*, *Author*, *Caption*, *Formula*, *Footnote*, and *Citation*. This target not only meets the requirements of most subsequent academic paper understanding tasks but also fully encompasses the components in PubLayNet, facilitating direct comparison. After determining the layout analysis target, we use DLAgen to generate the corresponding dataset, named DLAgenData.

The number of visual content sourced from real academic papers used in the dataset generation is: 66,836 figures, 13,355 tables, and 69,520 formulas. Utilizing different LaTeX template files for the same content significantly diversified the dataset, thereby enhancing the model's generalization ability, as shown in Figure 4. Samples of DLAgenData are illustrated in Figure 5, where the segmentation mask images serve as training data for semantic segmentation models, while the JSON format component coordinates are used for training the object detection models like MDT.

DLAgenData comprises 2,000 academic papers and 13,688 document images. The dataset was divided into training and test sets in a 4:1 ratio. Specifically, 1,600 papers with a total of 10,876 document images are used as the training set, while the remaining 400 papers with 2,812 document images are used as the test set. As in Section 4.2.1, we also train and test both MDT and DiT on DLAgenData to demonstrate the superiority of MDT.

4.2.3 Manually Annotated Fine-Grained Test Set. To demonstrate the consistency between academic papers generated by DLAgen and real academic papers, as well as the effectiveness of MDT trained on DLAgenData for fine-grained layout analysis on real academic papers, we randomly selected 500 document images from 100 ACL papers for fine-grained manual annotation as a test set. We test MDT and other layout analysis models, including the current SOTA model, on this test set.

4.3 Coarse-Grained Layout Analysis Results

Using DiT-Cascade-RCNN as a baseline, we construct three additional models by incorporating different features: MDT-P (with page position feature), MDT-T (with text feature), and MDT (with both features). These four models are trained and tested on PubLayNet. The test results include mAP scores for ten categories of components and the average mAP for all categories. The mAP is calculated as the average of AP values at IoU thresholds ranging from 0.5 to 0.95 in increments of 0.05. The results are shown in Table 1, where the highest mAP values for each component are highlighted. All three MDT models achieve higher mAP scores than DiT, even though the coarse-grained layout analysis task did not fully utilize the advantages of page position and text features.

4.4 Fine-Grained Layout Analysis Results

4.4.1 Results on the Generated Dataset. The four models used in Section 4.3 are trained and tested on DLAgenData. The experimental results are shown in Table 2. Overall, the baseline model performs well, achieving an average mAP of 0.9447. This indicates that with finely annotated datasets, the model is capable of performing fine-grained academic paper layout analysis, underscoring the necessity of DLAgen. Moreover, adding either page position features or text features improves the model's performance. MDT, which incorporates both features, achieved the highest average mAP of 0.9509. Furthermore, for each component category, the mAP of the three enhanced models were higher than those of the baseline model.

4.4.2 Results on the Manually Annotated Test Set. We test DiT trained on PubLayNet, as well as DiT, MDT-T, MDT-P, MDT, LayoutLMv3, and VGT (SOTA) trained on DLAgenData on the manually annotated dataset. The test results are shown in Table 3.

1) The Effectiveness of DLAgenData Comparing DiT (DLAgenData) and DiT (PubLayNet) intuitively reflects the advantage of training models on a fine-grained dataset, as DiT (DLAgenData) can identify five more components. DLAgenData further refines *Text* into *Text*, *Author*, *Caption*, *Footnote*, *Citation*, and also adds the *Formula* category. Although one might intuitively expect finer-grained layout analysis to be more difficult, experimental results show that DiT (DLAgenData)'s mAP is 0.0462 higher than that of DiT (PubLayNet). Since they use the same model, the difference in performance is not due to the model's capability.

Possible reasons include: Firstly, PubLayNet is composed solely of academic papers from the biomedical field, which lacks diversity in document styles, leading to reduced performance when the trained model is applied to ACL papers. Secondly, PubLayNet's coarse-grained classification lumps together *Text*, *Author*, *Caption*, *Footnote*, and *Citation* into a single *Text* category, despite significant differences among these categories. Recognizing such diverse categories as a single entity increases the difficulty. By visualizing the experimental results, it can be observed that most of the errors

Table 3: Test results on the manually annotated test set. The dataset used for training the model is enclosed in parentheses.

Component	DiT (PubLayNet)	DiT (DLAgenData)	LayoutLMv3 (DLAgenData)	VGT (DLAgenData)	MDT-T (DLAgenData)	MDT-P (DLAgenData)	MDT (DLAgenData)
Title	0.7366	0.7183	0.5628	0.6797	0.715	0.7206	**0.741**
Figure	0.8483	0.9057	0.8817	0.9143	0.9164	0.9095	**0.9249**
Table	0.7582	0.9448	0.9220	0.9461	0.9392	**0.9475**	0.9447
List	0.8404	0.7164	0.9024	**0.9314**	0.8594	0.9252	0.9172
Text	0.7782	0.9464	0.9497	0.9406	0.9508	0.9558	**0.9604**
Author	/	0.9374	0.8063	0.9031	0.9138	**0.9469**	0.9458
Caption	/	0.831	0.8184	0.8273	**0.8413**	0.8308	0.839
Footnote	/	0.6859	0.7312	0.7537	0.7999	0.7731	**0.8101**
Citation	/	0.9812	0.9277	**0.9927**	0.9697	0.9924	0.9811
Formula	/	0.7181	0.7312	0.6241	0.7439	0.7445	**0.7967**
Overall	0.7923	0.8385	0.8178	0.8513	0.8649	0.8746	**0.8861**

(a) (b)

Figure 6: Comparison of (a) DiT trained on PubLayNet and (b) MDT trained on DLAgenData for layout analysis on real papers.

in recognizing the coarse-grained *Text* by DiT (PubLayNet) stem from the fine-grained categories of *Author*, *Footnote*, and *Citation*.

Moreover, the test results of MDT (DLAgenData) on the manually annotated test set were only 0.0648 (0.9509→0.8861) lower than its performance on DLAgenData, confirming the high consistency between DLAgen-generated papers and real papers.

2) The Effectiveness of MDT Comparing all models trained on DLAgenData, MDT achieved the highest mAP value of 0.8861. Even MDT-T and MDT-P outperformed the current SOTA, demonstrating the effectiveness of page position and text features for fine-grained academic paper layout analysis.

Comparing MDT-T and MDT-P in terms of mAP values across different components, the difference is less than 0.01 for *Title*, *Text*, *Image*, *Table*, and *Formula*. For *Citation*, *Author*, and *List*, MDT-P performs noticeably better than MDT-T. This is because *Citation* and *Author* are strongly pagination-dependent: *Author* typically appears on the first page of a document, while *Citation* generally appears later. Improvements in *Citation* recognition consequently enhance the accuracy of *List* recognition, as *List* is often confused with *Citation*. For *Caption* and *Footnote*, MDT-T performs better than MDT-P, since long *Caption* may be visually similar to *Text*, even though they are distinct in textual content, often starting with "Figure" or "Table." Similarly, *Footnote* visually resembles *Text*, *List*, and *Citation*, but they often begin with numbers or symbols, making them distinguishable in textual content.

3) Visualization Analysis It's worth noting that in many cases, *Title*, *Caption*, and *Footnote* consist of only a few words or even a single word, occupying a very small area in document images. As a result, even when the model's predicted bounding boxes appear visually accurate, their IoU with manually annotated bounding boxes often does not reach the maximum threshold of 0.95 used to compute mAP. This discrepancy explains why MDT's mAP for these components is lower than those obtained on the generated dataset. In contrast, *Figure*, *Table*, and *Citation*, which typically occupy larger areas in document images, show minimal variation in mAP compared to results on the generated dataset. In fact, the mAP of *Citation* even exceed those on the generated dataset.

Therefore, to provide a more intuitive analysis of MDT's performance on real papers and the impact of adding page position and text features, the results of MDT trained on DLAgenData and DiT trained on PubLayNet are visualized in Figure 6, and results of MDT and DiT both trained onDLAgenData are visualized in Figure 7.

Observing Figure 6(a), a *Table* is incorrectly identified as a *Figure*, which is a confusion that often occurs in the application of DiT.

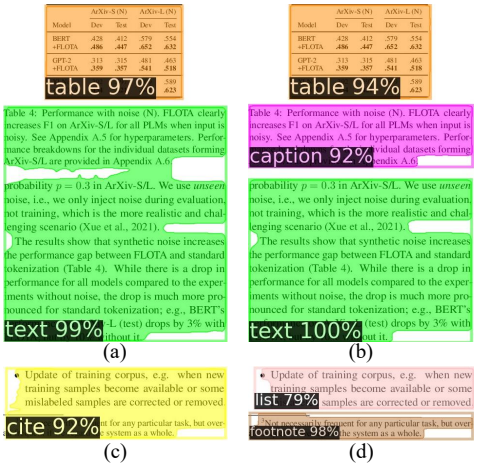


Figure 7: Comparison of layout analysis results between DiT and MDT: (a) and (c) show the results of DiT trained on DLAgenData, while (b) and (d) show the results of MDT-T and MDT-P trained on DLAgenData, respectively.

However, thanks to the diversity of document layouts in DLAgenData and the utilization of textual information within *Figure* and *Table*, Figure 6(b) correctly distinguishes between a *Figure* and a *Table*. Comparing Figure 7(a) and (b), it's evident that the use of text features aids in distinguishing long *Caption* from *Text*. Similarly, from the comparison in Figure 7(c) and (d), the use of page position features contributes to distinguishing *Footnote*, *Citations*, and *List*.

5 Conclusion

We present DLAgen, a method for accurately generating fine-grained annotated academic paper datasets for training layout analysis models, capable of producing academic papers of any scale, layout, and annotation type. This allows for the conversion of manual data annotation into automated annotation. Additionally, we propose MDT, a multimodal academic paper layout analysis model that utilizes visual, page position, and correctly ordered text features to cater to fine-grained academic paper layout analysis tasks. Using MDT trained on datasets generated by DLAgen, we achieve superior performance compared to SOTA models in real academic paper layout analysis. This confirms the consistency between DLAgen-generated papers and real papers, and demonstrates the effectiveness of MDT for fine-grained academic paper layout analysis.

Appendix

A Sentence Patterns Used for Generating DLAgenData

In Section 3.2, we introduce the construction of sentence patterns and keywords for different semantic structures, and used context-free grammar to generate the textual content. Here, we take the semantic structure "Abstract" as an example to illustrate the content generation method. It consists of three parts: ABSTRACT_INTRO introduces the research domain, ABSTRACT_PROBLEM describes the existing issues, and ABSTRACT_SOLUTION presents the solutions proposed by the paper. For instance, the sentence patterns for ABSTRACT_INTRO include but are not limited to:

- INNO_ADJ INNO_NOUN have garnered LIT_GREAT interest from PEOPLE in the last several years XXX
- In recent years, much research has been devoted to the ACT; LIT_REVERSAL, few have VERBED the ACT XXX
- PEOPLE agree that INNO_ADJ INNO_NOUN are an interesting new topic in the field of FIELD XXX
- The ACT has VERBED THING_MOD, and current trends suggest that the ACT will soon emerge XXX
- The FIELD APPROACH to THING_MOD is defined not only by the ACT, but also by the ADJ need for THING_MOD XXX
- Unified INNO_ADJ INNO_NOUN have led to many ADJ advances, including THING_MOD and THING_MOD XXX

The uppercase parts are keywords. For example, in the first sentence pattern, INNO_ADJ represents adjectives describing innovative technologies, including "autonomous ," "robust," "scalable," "self-learning," "stochastic," etc.; INNO_NOUN represents nouns related to innovative technologies, such as "algorithms," "configurations," "methodologies," "modalities," "symmetries," etc.; LIT_GREAT represents evaluative adjectives like "great," "limited," "minimal," "profound," "tremendous," etc.; PEOPLE refers to individuals, including "analysts," "experts," "researchers," "scholars," "statisticians," etc.; XXX represents the sentence ending, which could be a period or a citation followed by a period. A complete sentence can be: "Autonomous vehicles have garnered considerable interest from academics in the last several years."

B LaTeX Template Files Used for Generating DLAgenData

In Section 3.2, we introduce the use of various LaTeX template files to increase the layout diversity of the generated academic papers. The LaTeX template files we use include, but are not limited to:

Single Column	Double Column
article.cls	IEEEtran.cls
IEEEphot.cls	IEEEims.cls
dccpaper.cls	IEEEoj.cls
stvrauth.cls	IEEEcsmag.cls
ouparticle.cls	IEEEjmw.cls
elsarticle.cls	elsart.cls
entcs.cls	sig-alternate.cls
svjour3.cls	CUP-JNL-PPS.cls
amsart.cls	acmart.cls

C More document images in DLAgenData

We present more document images in DLAgenData in Figure 8.

D Incorrect Annotations in DocBank

In Section 2.1, we point out that DocBank contains many inaccurate annotations, some of which are shown in Figure 9. These inaccuracies include: incorrect component category annotations, incorrect component boundary annotations, incorrect splitting of whole components, and incorrect merging of different components.

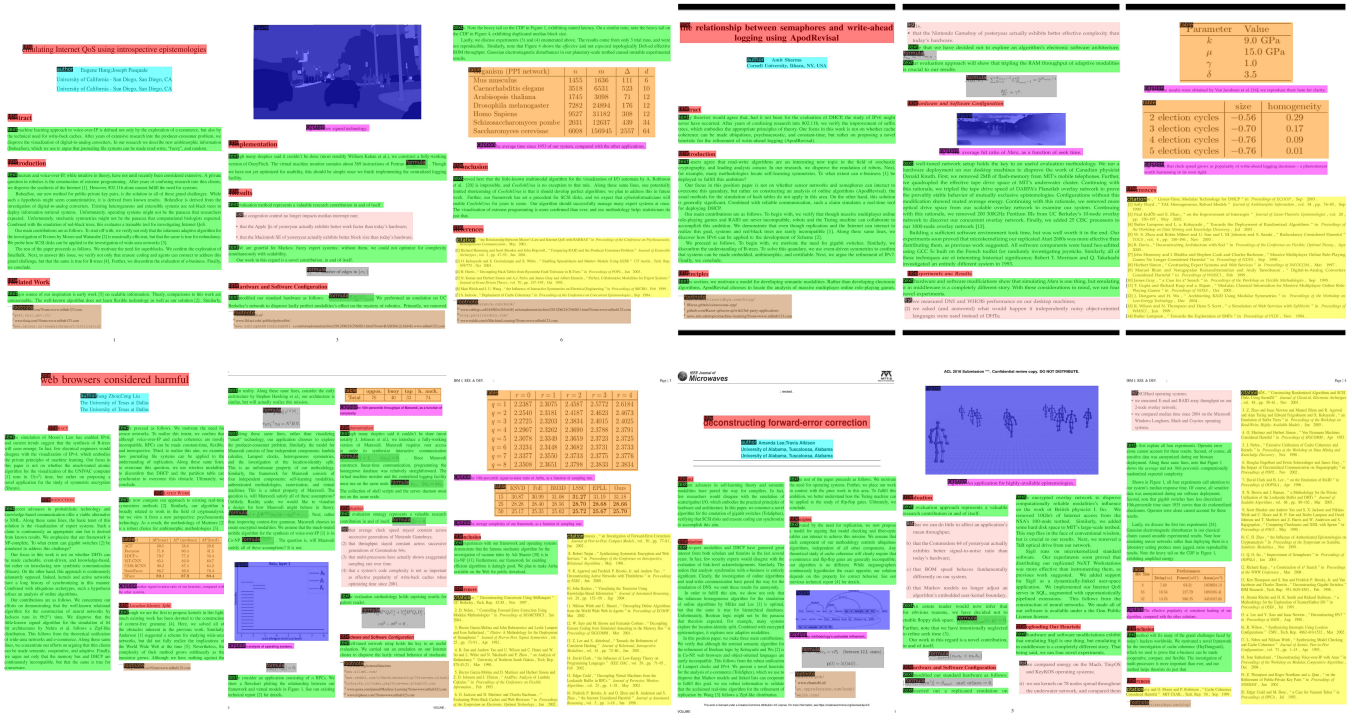


Figure 8: Document images in DLAgenData.

Figure 9: Incorrect annotations in DocBank.

References

[1] Apostolos Antonacopoulos, David Bridson, Christos Papadopoulos, and Stefan Pletschacher. 2009. A realistic dataset for performance evaluation of document layout analysis. In *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 296–300.

[2] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha. 2021. Docformer: End-to-end transformer for document understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*. 993–1003.

[3] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2021. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254* (2021).

[4] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676* (2019).

[5] Zhaowei Cai and Nuno Vasconcelos. 2018. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6154–6162.

[6] Cheng Da, Chuwei Luo, Qi Zheng, and Cong Yao. 2023. Vision grid transformer for document layout analysis. In *Proceedings of the IEEE/CVF international conference on computer vision*. 19462–19472.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[8] Yuntian Deng, Anssi Kanervisto, Jeffrey Ling, and Alexander M Rush. 2017. Image-to-markup generation with coarse-to-fine attention. In *International Conference on Machine Learning*. PMLR, 980–989.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).

[10] Tobias Grüning, Gundram Leifert, Tobias Strauß, Johannes Michael, and Roger Labahn. 2019. A two-stage method for text line detection in historical documents. *International Journal on Document Analysis and Recognition (IJDAR)* 22, 3 (2019), 285–302.

[11] Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Nikolaos Barmpalios, Ani Nenkova, and Tong Sun. 2021. Unidoc: Unified pretraining framework for document understanding. *Advances in Neural Information Processing Systems* 34 (2021), 39–50.

[12] Zhangxuan Gu, Changhua Meng, Ke Wang, Jun Lan, Weiqiang Wang, Ming Gu, and Liqing Zhang. 2022. Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4583–4592.

[13] Dafang He, Scott Cohen, Brian Price, Daniel Kifer, and C Lee Giles. 2017. Multi-scale multi-task fcn for semantic page segmentation and table detection. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, Vol. 1. IEEE, 254–261.

[14] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 16000–16009.

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 2961–2969.

[16] 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*. 4083–4091.

[17] Antonio Jimeno Yepes, Peter Zhong, and Douglas Burdick. 2021. ICDAR 2021 competition on scientific literature parsing. In *Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part IV 16*. Springer, 605–617.

[18] Junlong Li, Yiheng Xu, Tengchao Lv, Lei Cui, Cha Zhang, and Furu Wei. 2022. Dit: Self-supervised pre-training for document image transformer. In *Proceedings of the 30th ACM International Conference on Multimedia*. 3530–3539.

[19] 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*. 949–960.

[20] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2117–2125.

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 740–755.

[22] Minh-Thang Luong, Thuy Dung Nguyen, and Min-Yen Kan. 2012. Logical structure recovery in scholarly articles with rich document features. In *Multimedia Storage and Retrieval Innovations for Digital Library Systems*. IGI Global, 270–292.

[23] Lawrence O'Gorman. 1993. The document spectrum for page layout analysis. *IEEE Transactions on pattern analysis and machine intelligence* 15, 11 (1993), 1162–1173.

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

[25] B Pfitzmann, C Auer, M Dolfi, AS Nassar, and PWJ Staar. 2022. Doclaynet: A large humanannotated dataset for document-layout analysis. *URL: https://arxiv.org/abs/2206* 1062 (2022).

[26] Subhojeet Pramanik, Shashank Mujumdar, and Hima Patel. 2020. Towards a multi-modal, multi-task learning based pre-training framework for document representation learning. *arXiv preprint arXiv:2009.14457* (2020).

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* 28 (2015).

[28] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2021. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*. PMLR, 10347–10357.

[29] Christoph Wick and Frank Puppe. 2018. Fully convolutional neural networks for page segmentation of historical document images. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*. IEEE, 287–292.

[30] Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. 2021. Layoutxlm: Multimodal pre-training for multilingual visually-rich document understanding. *arXiv preprint arXiv:2104.08836* (2021).

[31] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. 2021. LayoutLMv2: Multimodal 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)*. 2579–2591.

[32] Xiao Yang, Ersin Yumer, Paul Asente, Mike Kraley, Daniel Kifer, and C Lee Giles. 2017. 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*. 5315–5324.

[33] Peng Zhang, Can Li, Liang Qiao, Zhanzhan Cheng, Shiliang Pu, Yi Niu, and Fei Wu. 2021. VSR: a unified framework for document layout analysis combining vision, semantics and relations. In *Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part I 16*. Springer, 115–130.

[34] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. 2019. Publaynet: largest dataset ever for document layout analysis. In *2019 International conference on document analysis and recognition (ICDAR)*. IEEE, 1015–1022.