Are Layout-Infused Language Models Robust to Layout Distribution Shifts? A Case Study with Scientific Documents

Catherine Chen[♠*] **Zejiang Shen**[°] **Dan Klein**[♠]
Gabriel Stanovsky[♢♡] **Doug Downey**[♢□] **Kyle Lo**[♢]

[♠]University of California Berkeley, [°]Massachusetts Institute of Technology,
[♢]Allen Institute for AI, [♡]Hebrew University of Jerusalem, [□]Northwestern University

{cathychen,klein}@berkeley.edu {zjshen}@mit.edu {gabis,dougd,kylel}@allenai.org

Abstract

Recent work has shown that infusing layout features into language models (LMs) improves processing of visually-rich documents such as scientific papers. Layout-infused LMs are often evaluated on documents with familiar layout features (e.g., papers from the same publisher), but in practice models encounter documents with unfamiliar distributions of layout features, such as new combinations of text sizes and styles, or new spatial configurations of textual elements. In this work, we test whether layout-infused LMs are robust to layout distribution shifts. As a case study, we use the task of scientific document structure recovery, segmenting a scientific paper into its structural categories (e.g., TITLE, CAPTION, REFERENCE). To emulate distribution shifts that occur in practice, we re-partition the GROTOAP2 dataset. We find that under layout distribution shifts model performance degrades by up to 20 F1. Simple training strategies, such as increasing training diversity, can reduce this degradation by over 35% relative F1; however, models fail to reach in-distribution performance in any tested out-of-distribution conditions. This work highlights the need to consider layout distribution shifts during model evaluation, and presents a methodology for conducting such evaluations.[1]

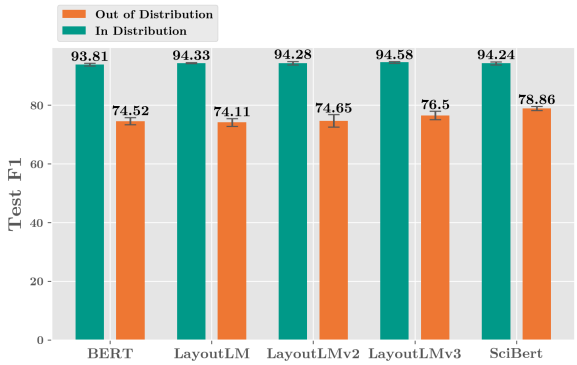

Figure 1: Model performance on document structure recovery, comparing training and testing in-distribution vs out-of-distribution. Error bars indicate standard deviation across runs. Layout distribution shifts degrade model performance by up to 20 F1 (Section 5.2). Simple training strategies such as few-shot fine-tuning and increasing training diversity partially mitigate the drop shown here (Section 5.3, Section 5.4).

1 Introduction

Humans use layout to understand the organizational structure of visually-rich documents such as scientific papers, newspaper articles, and web pages. For instance, a reader might use fontsize and boldfacing to recognize a section title, while they might use spatial location to recognize a footnote. Based on the intuition that layout aids in document understanding, recent work introduced layout-infused language models (LMs). To improve document processing, these models incorporate layout features such as the styling, size, and spatial configuration of document text. However, these features often change between documents – are layout-infused LMs robust to shifts in layout distribution?

With rising interest in processing visually-rich documents, some language models have been augmented with components specifically designed to process layout features (e.g., Xu et al., 2020). Layout-infused models accurately process documents with similar layouts to those seen during training (Shen et al., 2022; Huang et al., 2022b), and can leverage visual information to better understand long-range dependencies (Nguyen et al., 2023). But in practice, models often encounter documents with different layouts – for instance, pages with a different number of columns, a different density of words on the page, and different locations of textual elements. In order to realistically evaluate model performance, we study model performance under layout distribution shifts. Although robustness to text-distribution shifts has been relatively

*Work primarily done during internship at AI2.

[1]Our code and evaluation suite are available at https://github.com/cchen23/layout_distribution_shift.

Findings of the Association for Computational Linguistics: ACL 2023, pages 13345–13360
July 9-14, 2023

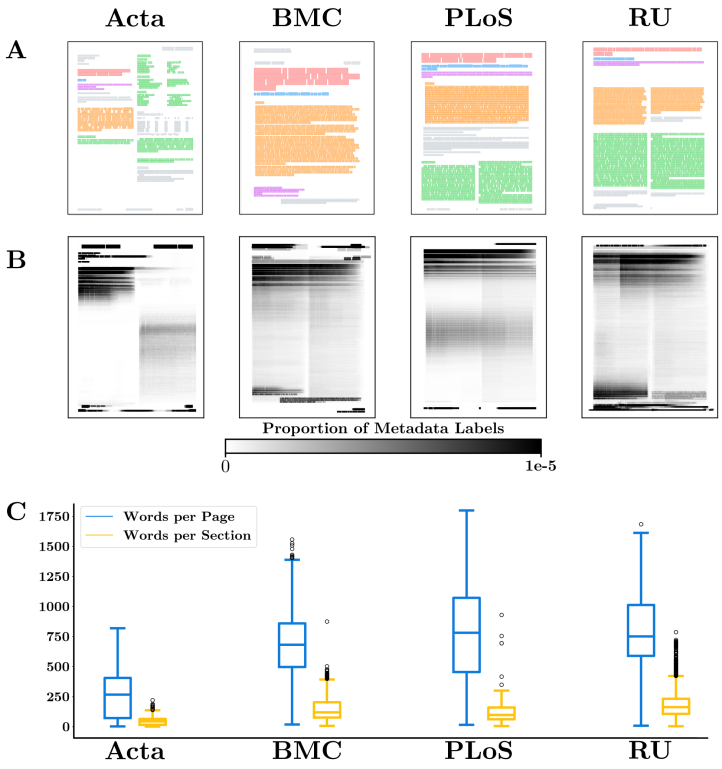

Figure 2: Differences in layout between publishers (ACTA, BMC, PLOS, and RU). **A**. One representative page from each publisher. Each word is colored according to structural category. Abstracts (orange) are structured in one (ACTA, BMC, PLOS) or two (RU) columns, spanning half (ACTA) or full (BMC, PLOS, RU) page widths. Author affiliations (pink) may appear at the top (ACTA, PLOS, RU) or bottom (BMC) of the page. **B**. Spatial distribution of metadata categories (e.g., AUTHOR_TITLE, PAGE_NUM). **C**. Boxplots showing number of words per page and section, for each publisher.

well-studied, layout distribution shifts pose unique challenges and may require unique solutions; therefore, we focus specifically on robustness to layout distribution shifts.

Here we present a case study to evaluate model robustness to layout distribution shifts. Our case study focuses on the task of segmenting a scientific paper into its structural categories. For example, a scientific paper might be segmented into categories such as TITLE, AUTHORS, CAPTION, PARAGRAPH, and REFERENCE. We refer to this task as *document structure recovery*.[2] We chose to focus on this task because it serves as a testbed to determine whether layout-infusion remains beneficial under layout distribution shifts. Document structure recovery requires grasping the organizational structure of a document, a key piece of information that layout conveys. Moreover, layout-infused models have been shown to reach state-of-the-art performance on this task, in settings where the layout distribution is the same between training and testing (Shen et al., 2022; Huang et al., 2022b).

To test model robustness, evaluations must train and test on examples drawn from different layout distributions. However, existing datasets for document structure recovery use random train-test splits, and thus are ill-suited for evaluating model robustness. In this work we leverage publisher metadata to construct train-test splits that reflect layout distribution shifts. Publisher metadata is a proxy for layout distribution shifts because publication venue is a key driver of layout differences – different publishers adhere to different style guides and templates (e.g., Figure 2). Moreover, layout differences across publishers reflect layout distribution shifts faced in practice as new publishers, templates, and style guides arise. We use publisher metadata to propose new train-test splits of an existing dataset (GROTOAP2, Tkaczyk et al. (2014)) for scientific document structure recovery. These splits are designed to reflect **L**ayout **D**istribution **S**hifts; hence we refer to the splits as GROTOAP2-LDS.

[2]Other names for this task include layout analysis (Li et al., 2020; Zhong et al., 2019), logical structure recovery (Huang et al., 2022a), zone classification (Tkaczyk et al., 2014), and text classification (Shen et al., 2022).

Using GROTOAP2-LDS, we evaluate a set of layout-infused LMs and find that model performance degrades by up to 20 F1 under layout distribution shifts (Figure 1). We show that layout-infused models can quickly adapt to new distributions, and that increasing training diversity can improve model robustness. However, even with diverse training sets and fewshot fine-tuning, performance on out-of-distribution layouts remains more than 2 F1 below in-distribution performance. Although layout-infusion aids in processing documents with in-distribution layouts, layout-infused LMs may overfit on features seen during training. We release our code and evaluation suite to enable future evaluations and to facilitate expansions of our evaluation suite.

2 Background and Related Work

Recent work established that models are often sensitive to distribution shifts. Shifts in the distribution of text or image statistics have been shown to substantially degrade model performance (e.g., Geirhos et al., 2020; Bai et al., 2021; Ye et al., 2021; Miller et al., 2020; Koh et al., 2021), even in cases when human performance is robust to these distribution shifts (Miller et al., 2020). For example, question-answering models struggle to generalize from text in Wikipedia to text in newspaper articles (Miller et al., 2020), and image classification models struggle to generalize between images taken from different cameras (Koh et al., 2021). We extend this line of research to study robustness to shifts in the distribution of layout features.

Document structure recovery provides an opportune setting for evaluating robustness to layout distribution shifts. Solving this task requires understanding how the text and visual layout of a page convey the organizational structure of the document, and layout-infused LMs have been shown to reach near-human performance on this task (Tkaczyk et al., 2014; Shen et al., 2022). Prior work has shown that models transfer poorly across different document types (e.g., from scientific papers to financial documents) (Pfitzmann et al., 2022). Although different document types exhibit differences in layout, they also exhibit large differences in other features, such as the textual domain and the distribution of structural categories. It is therefore unclear whether poor transfer across document types is due to layout distribution shifts or other factors. In this work, we experiment using train-test splits exhibiting different layout distributions but with documents of the same type (i.e., scientific papers from biomedical journals).[3]

Existing evaluation datasets for document structure recovery include many document types, such as scientific papers (Tkaczyk et al., 2014; Li et al., 2020; Zhong et al., 2019), forms (Jaume et al., 2019), receipts (Park et al., 2019), and long-form business documents (Grali'nski et al., 2020; Pfitzmann et al., 2022). We focus on scientific papers, where layout distribution shifts are prevalent (Figure 2). Although existing datasets for scientific document structure recovery contain documents with different layouts, existing train-test splits do not reflect layout distribution shifts.

3 Evaluation Methodology

To evaluate model robustness, we propose a set of new train-test splits of GROTOAP2. These splits reflect layout distribution shifts that occur in practice, and we refer to this set of splits as GROTOAP2-LDS. In this section we formally define our task (Section 3.1), describe our procedure for partitioning data into splits that emulate layout distribution shifts (Section 3.2), and present a specific benchmark for evaluating robustness to layout distribution shifts (GROTOAP2-LDS, Section 3.3).

3.1 Task Definition

For each page, a model receives N words $w_0, ..., w_N$ in detected reading order. Layout-infused models receive additional page features, such as the x- and y- coordinates of the bounding box of each word or an image of the page. Given these inputs, the model must predict category labels $y_0, ..., y_N$, one for each word, where y_i is selected from a set of structural page categories (e.g., TITLE, CAPTION, AUTHORS)

3.2 Dataset Construction Procedure

We focus on layout distribution shifts within scientific papers, but our data partitioning procedure is agnostic to the particular document type. In the future, this procedure could be used to evaluate

[3]Concurrent work (Wang et al., 2022) evaluates how well models extract information on unseen form types (e.g., training on "Amendment" and "Short Form" foreign agent registration forms from the US government, and testing on "Dissemination Report" forms). In contrast, our experiments evaluate transfer on documents of the same type (scientific articles from biomedical journals), and experiment on a wider variety of layout transfer settings (e.g., varying the amount of layout diversity seen during training) (Section 5.4).

model robustness with other types of documents, such as receipts from different vendors or articles from different newspapers.

Document-Level Layout Assignments Previous evaluation setups assigned dataset splits at the page level, sometimes placing different pages of the same document in both the train and the test set (Tkaczyk et al., 2014; Li et al., 2020). However, layout formatting decisions are often made at the document-level, and the layout of different pages in a multi-page document are often highly dependent on each other. We therefore consider layout in terms of whole documents, and assign dataset splits at the document level.

Provenance Metadata as a Proxy for Layout Distribution Shifts For scientific papers, different publishers format papers with different layouts (Tkaczyk et al., 2014), and layout differences across publishers reflect distribution shifts that may occur in practice. We therefore use publisher metadata to partition documents into different dataset splits. Existing datasets often do not preserve provenance metadata, instead including only the content and task labels for each document (e.g., Tkaczyk et al., 2014; Li et al., 2020; Zhong et al., 2019). Fortunately, scientific literature citation tools provide a way to recover publisher metadata for scientific papers. To link each paper to its associated publisher, we access the Semantic Scholar database to obtain the journal based on the title of each publication in GROTOAP2, and then map from each journal to the corresponding publisher.

3.3 The GROTOAP2-LDS Benchmark

We use the procedure described in Section 3.2 to construct GROTOAP2-LDS, a set of train-test splits that evaluate model performance under different training conditions.

Test splits We construct four test sets, each of which contains papers from a held-out publisher (ACTA, BMC, PLOS, RU). GROTOAP2 contains a large number of papers from each of these publishers (at least 300 papers per publisher, or about 1 million words), ensuring enough data to compute reliable estimates of in-distribution and out-of-distribution performance.[4] Each of these four publishers contains papers from a qualitatively distinct layout distribution (Figure 2). The same four test sets are used to evaluate models under each train condition.

Test publisher	Train papers	Test papers
ACTA	2039	44
BMC	1886	63
RU	1893	62
PLOS	1349	130

Table 1: GROTOAP2-LDS split sizes.

For each publisher, 20% of papers were used as a held-out test set, and the remaining 80% of papers were used in certain training conditions (e.g., to compute an estimate of in-distribution performance). The test sets contain an average of 75 papers (≈500,000 words) each.

GROTOAP2-LDS includes 12 label categories (ABSTRACT, ACKNOWLEDGEMENTS, AFFILIATION, AUTHOR_TITLE, BIBLIOGRAPHIC_INFO, BODY, DATES, FIGURE, PAGE_NUM, REFERENCES, TABLE, UNKNOWN). Train/test split sizes are included in Table 1.

In-Distribution (ID) Training For each of the four held-out test publishers, we construct a training set with papers from the same publisher. Papers from the same publisher exhibit different layouts, but layout differences between papers within the same publisher are small relative to differences between papers from different publishers. We therefore refer to settings in which models are trained and tested on papers from the same publisher as the "in-distribution" setting, and settings involving transfer across publishers as the "out-of-distribution" setting. Model performance in this setting is used to estimate the performance drop between in-distribution and out-of-distribution layouts.

Out-of-Distribution (OOD) Training We construct training sets that evaluate model performance under layout distribution shift. The number of train papers is matched between training sets. Each training set contains roughly 2,000 papers (≈10,000,000 words). We construct training sets reflecting different levels of layout diversity. Our default training approach ("LIMITEDPUBLISHER") is a leave-one-publisher-out setting in which each model is trained on three publishers and tested on the held-out fourth publisher. To evaluate the impact of training set diversity on robustness to layout

[4] Other publishers also met the minimum number of papers/words (e.g., Nucleic Acids Research). We focused on four publishers to keep the number of experiments tractable.

Data split	B (%)
In-Distribution	50.13
LIMITEDPUBLISHER	64.54
LIMITEDPUBLISHER+	75.47
LIMITEDPUBLISHER++	78.46

Table 2: Breadth of layouts in each training set. B denotes the percentage of spatial page locations covered by each structural category, averaged over categories.

distribution shifts, we construct datasets with 25 publishers ("LIMITEDPUBLISHER+") or 125 publishers ("LIMITEDPUBLISHER++").

To quantify the diversity of spatial configurations in each training set, we measure the breadth (B) of spatial locations covered by each structural category. To compute B, for each structural category we count the proportion of spatial x-y positions where that category occurs,[5] and then compute the mean across categories. The value of B for each data split is included in Table 2.

Few-shot Adaptation In practice, it may be possible to cheaply annotate a few papers from a new layout distribution (e.g., when a trained model is applied to papers from a new publisher.) To test how quickly models can adapt to a new layout distribution, we additionally evaluate models in settings in which models are first trained on an out-of-distribution training set, and are then fine-tuned on a small amount of in-distribution data. Specifically, before testing models on each of the test sets, we perform few-shot fine-tuning with a few annotated examples (10 papers, $\approx 50,000$ words) from the held-out test publisher.

4 Experiment Details

4.1 Models

We evaluate on BERT, LayoutLM, LayoutLMv2, LayoutLMv3, and SciBERT (we use the base uncased version of each model).[6] The three layout-infused models (LayoutLM, LayoutLMv2, LayoutLMv3) share the same model size and underlying architecture as BERT (Devlin et al., 2019). The equivalence in model size facilitates direct comparisons between different methods of incorporating layout features. Each layout-infused model is adapted to use layout features such as text position or page image embeddings on top of the standard BERT architecture. These layout-infused models have previously been shown to achieve state-of-the-art performance for processing visually-rich text documents with in-distribution layouts (Xu et al., 2020, 2021; Huang et al., 2022b; Shen et al., 2022). We briefly describe these layout-infused models, and defer to the original papers for more specific details about model architecture and training.

LayoutLM (Xu et al., 2020): LayoutLM is initialized from BERT, and then adapted to incorporate information about spatial text position. Masked visual-language modeling and multi-label document classification are used to adapt the model to incorporate the layout-specific components.

LayoutLMv2 (Xu et al., 2021): LayoutLMv2 is initialized from BERT, and then adapted to incorporate spatial text position as well as image embeddings of page regions. Masked visual-language modeling and text-image alignment are used to adapt the model to incorporate the layout-specific components.

LayoutLMv3 (Huang et al., 2022b): LayoutLMv3 is initialized from RoBERTa, and then adapted to incorporate spatial text position as well as image embeddings of page patches. Masked language modeling, masked image modeling, and word-patch alignment are used to adapt the model to incorporate the layout-specific components.

We additionally evaluate on SciBERT (Beltagy et al., 2019), which is pretrained with the same pretraining tasks as BERT, but instead with data from scientific texts. SciBERT allows us to compare the benefit of layout-infusion with the benefit of simply using a model pretrained on in-domain text.

I-VILA tokens, which provide a textual indication of visual group boundaries as part of model input, have been shown to improve performance on document structure recovery (Shen et al., 2022). Our preliminary experiments showed that I-VILA tokens improve performance across all experimental settings. Therefore for all reported experiments, we use block-level I-VILA tokens provided by Shen et al. (2022).

4.2 Implementation Details

We implemented experiments in PyTorch, using the transformers library to access pretrained models (Paszke et al., 2019; Wolf et al., 2020). The learning rate for each model was selected by train-

[5] x-y positions are determined using pixel locations in images of each page.

[6] Because of computational constraints we evaluate on a subset of all existing layout-infused models. We release our code and train-test splits to aid evaluations of other models.

ing each model with a learning rate of 1e-04, 1e-05, and 1e-06, and selecting the learning rate with the best dev set performance. This learning rate sweep was done separately for the initial training phase and for few-shot fine-tuning (see Appendix for details). The initial training stage included a linear warmup schedule with 2000 steps. The adamW optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$ was used during training. During each episode of few-shot fine-tuning, eight papers were used as the train set and two papers were used as the dev set. No warmup was used during few-shot fine-tuning. Batch size four was used throughout training. Models were trained for a maximum of 10 epochs during the initial training phase and a maximum of 250 epochs during few-shot fine-tuning. Dev set performance was used for early stopping. For each model, the initial training phase was performed over three random seeds. For each random seed, few-shot fine-tuning was performed over three different episodes. Tables in Section 5 show the mean and standard deviation over random seeds and episodes. Full experimental results are included in the Appendix.

Test split	Base model	ID
SPLIT AVG	BERT	93.81 ±0.42
	LayoutLM	94.33 ±0.18
	LayoutLMv2	94.28 ±0.57
	LayoutLMv3	**94.58** ±0.32
	SciBERT	94.24 ±0.52
ACTA	BERT	86.78 ±0.84
	LayoutLM	86.71 ±0.23
	LayoutLMv2	87.40 ±0.69
	LayoutLMv3	**88.67** ±0.34
	SciBERT	87.58 ±1.12
BMC	BERT	95.82 ±0.41
	LayoutLM	96.14 ±0.09
	LayoutLMv2	95.77 ±1.09
	LayoutLMv3	95.79 ±0.63
	SciBERT	**96.23** ±0.19
RU	BERT	95.38 ±0.32
	LayoutLM	**96.64**±0.20
	LayoutLMv2	96.55 ±0.17
	LayoutLMv3	96.36 ±0.09
	SciBERT	95.85 ±0.37
PLOS	BERT	97.24 ±0.12
	LayoutLM	**97.84** ±0.19
	LayoutLMv2	97.38 ±0.32
	LayoutLMv3	97.48 ±0.22
	SciBERT	97.30 ±0.42

Table 3: **In-distribution performance.** Test macro-F1 on document structure recovery. Mean and standard deviation over trials is reported. The best performance is highlighted in blue. Layout-infused models achieve the highest in-distribution test performance.

5 Results

We use GROTOAP2-LDS (Section 3.3) to evaluate robustness to layout distribution shifts. For each experimental condition, models are evaluated on four test sets, each containing papers from a held-out layout distribution. Unless indicated otherwise, model performance is reported as the average across these four test sets.

5.1 Layout-infused LMs Perform Best on In-Distribution Layouts

In-distribution performance of each model is shown in Table 3. Consistent with prior work, we find that layout-infused LMs reach the highest performance for documents with in-distribution layouts.[7] In subsequent sections, we use Δ_{ID} to refer to the difference between model performance on this in-distribution training condition, and on out-of-distribution training conditions.

5.2 Models Overfit to Layout Distributions Seen During Training

To evaluate model robustness to layout distribution shifts, we train models on papers from three publishers (LIMITEDPUBLISHER), and then test on papers from a held-out test publisher. Model performance for each of the test sets is shown in Table 4. Compared to in-distribution performance (Table 3), out-of-distribution performance drops between 15.38 and 20.22 F1 (Δ_{ID}). Layout-infused models perform worse than SciBERT, a model not pretrained with layout-specific components. Although layout-infused models achieve the highest performance for in-distribution layouts, these models overfit to layout distributions seen during training. In settings in which models need to generalize to out-of-distribution layouts, models with in-distribution text pretraining (as with SciBERT) may be more effective.

5.3 Models Can Quickly Adapt to Layout Distribution Shifts

In practice, it may sometimes be possible to cheaply annotate a few papers from a target distribution (e.g., when a system ingests papers from a new publisher). To test how well models can quickly adapt to a new layout distribution, we first train models on out-of-distribution layouts (LIMITEDPUBLISHER). For each of the four test

[7] Note that the inference-time costs of LayoutLMv2 and LayoutLMv3 are around 10× more than other tested models.

Test split	Base model	OOD	Δ_{ID}
SPLIT AVG	BERT	74.52 ±1.22	−19.29
	LayoutLM	74.11 ±1.30	−20.22
	LayoutLMv2	74.65 ±2.09	−19.63
	LayoutLMv3	76.50 ±1.39	−18.07
	SciBERT	**78.86** ±0.74	**−15.38**
ACTA	BERT	51.89 ±1.20	−34.89
	LayoutLM	51.15 ±1.20	−35.56
	LayoutLMv2	55.83 ±0.98	−31.57
	LayoutLMv3	55.83 ±3.21	−32.84
	SciBERT	**60.66** ±0.28	**−26.92**
BMC	BERT	72.74 ±1.19	−23.09
	LayoutLM	74.84 ±0.50	−21.3
	LayoutLMv2	73.26 ±1.86	−22.51
	LayoutLMv3	74.83 ±0.86	−20.96
	SciBERT	**78.10** ±1.13	**−18.13**
RU	BERT	83.68 ±2.02	−11.7
	LayoutLM	82.91 ±1.52	−13.73
	LayoutLMv2	81.62 ±3.49	−14.93
	LayoutLMv3	84.13 ±1.24	−12.23
	SciBERT	**87.36** ±0.79	**−8.49**
PLOS	BERT	89.76 ±0.48	−7.48
	LayoutLM	87.55 ±1.96	−10.29
	LayoutLMv2	87.87 ±2.03	−9.51
	LayoutLMv3	**91.23** ±0.23	**−6.25**
	SciBERT	89.32 ±0.76	−7.98

Table 4: **Out-of-distribution performance.** Test macro-F1 on document structure recovery. Layout distribution shift substantially degrades performance of all models, with SciBERT achieving the best out-of-distribution test performance. For generalization to new layouts, in-domain text pretraining may be more effective than layout-infusion.

Test split	Base model	OOD	Δ_{ID}
SPLIT AVG	BERT	89.64 ±0.67	−4.16
	LayoutLM	90.14 ±0.69	−4.19
	LayoutLMv2	**90.95** ±0.65	**−3.32**
	LayoutLMv3	90.28 ±0.63	−4.3
	SciBERT	90.50 ±0.58	−3.73
ACTA	BERT	79.03 ±0.93	−7.75
	LayoutLM	80.49 ±1.11	−6.22
	LayoutLMv2	**81.45** ±0.86	**−5.95**
	LayoutLMv3	79.89 ±0.85	−8.78
	SciBERT	80.30 ±0.73	−7.28
BMC	BERT	92.10 ±1.02	−3.73
	LayoutLM	93.32 ±0.73	−2.82
	LayoutLMv2	**94.22** ±0.70	**−1.55**
	LayoutLMv3	93.19 ±0.57	−2.6
	SciBERT	93.74 ±0.59	−2.49
RU	BERT	91.57 ±0.19	**−3.81**
	LayoutLM	90.72 ±0.41	−5.92
	LayoutLMv2	**91.99** ±0.47	−4.56
	LayoutLMv3	91.64 ±0.75	−4.72
	SciBERT	91.79 ±0.44	−4.06
PLOS	BERT	95.88 ±0.54	−1.36
	LayoutLM	96.04 ±0.52	−1.8
	LayoutLMv2	96.15 ±0.58	−1.23
	LayoutLMv3	**96.38** ±0.36	**−1.1**
	SciBERT	96.19 ±0.56	−1.11

Table 5: **Out-of-distribution performance (test macro-F1), after few-shot adaptation.** Performance on OOD layouts falls below ID performance, but few-shot fine-tuning reduces the performance drop by up to 80%. LayoutLMv2 achieves the best out-of-distribution test performance. Layout infusion may facilitate adaptation to new layout distributions.

splits, we perform few-shot fine-tuning with ten papers from the held-out test publisher, and then evaluate on the test set for that publisher.

Table 5 shows model performance in this setting. From the in-distribution to out-of-distribution settings, model performance drops between 3.3 and 4.3 F1 (Δ_{ID}). Although model performance falls substantially below in-distribution performance, few-shot adaptation to the target distribution reduces the performance drop by over 80% compared to settings in which models must directly generalize to the new distribution (Table 4). After few-shot adaptation, LayoutLMv2 achieves the highest out-of-distribution test performance, suggesting that layout-infusion may help models adapt more quickly to new layout distributions.

5.4 Increasing Layout Diversity Observed During Training Can Improve Robustness

To determine whether layout-diverse training can improve model robustness, we train models on papers from more publishers while holding the total number of papers constant (the LIMITEDPUBLISHER+ and LIMITEDPUBLISHER++ training sets described in Section 3.3). Model performance for each training diversity condition is shown in Figure 3. Performance is shown separately for settings in which models must generalize directly to papers from a different layout distribution (as in Section 5.2), and for settings in which models are fine-tuned on a few annotated examples from the target distribution (as in Section 5.3).

When models must generalize directly to papers from a different layout distribution, a change from training on LIMITEDPUBLISHER to LIMITEDPUBLISHER+ increases test performance on out-of-distribution layouts by a mean of 9.91 F1 over models. A further increase in diversity from LIMITEDPUBLISHER+ to LIMITEDPUBLISHER++ increases performance by an additional 0.28 F1. In settings where models receive a few annotated examples to adapt to the target distribution (e.g., Section 5.3), training on LIMITEDPUBLISHER+ rather than LIMITEDPUBLISHER yields a much smaller performance gain (1.53 F1). In few-shot adaptation settings, a further increase from training on LIM-

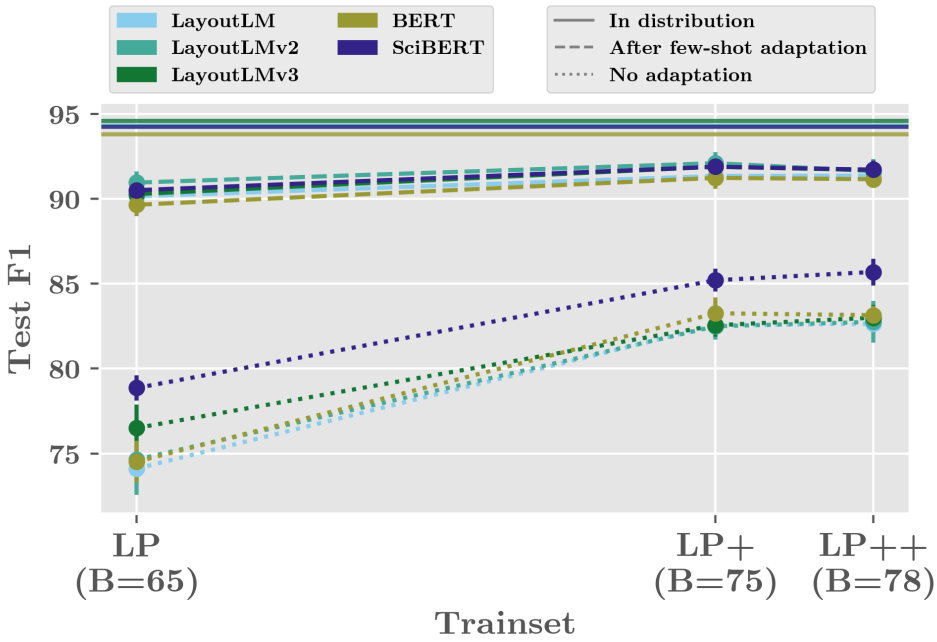

Figure 3: **Out-of-distribution performance vs training diversity.** Test macro-F1 on document structure recovery. LP=LIMITEDPUBLISHER. Error bars reflect standard deviation over trials. (1) Increasing training diversity improves robustness to layout distribution shifts, but even the highest training diversity condition does not reach ID performance. (2) Increasing training diversity provides diminishing benefits. (3) Benefits of training diversity overlap with benefits from few-shot adaptation.

ITEDPUBLISHER+ to LIMITEDPUBLISHER++ results in a -0.19 F1 drop in performance.

These results suggest that increasing the diversity of layouts observed during training can improve model robustness, but that this strategy provides diminishing returns as training diversity continues to increase. Furthermore, the benefits of increasing training diversity may largely overlap with the benefits of few-shot adaptation to the target distribution. Even in the most favorable out-of-distribution setting, in which models are trained on the most beneficial training diversity condition and then fine-tuned on a few papers from the target layout distribution, model performance is at least 2 F1 below in-distribution performance.

5.5 Error Analysis

In practice, shifts in layout and text distribution are highly correlated. For instance, papers written for different scientific communities differ in both textual content and visual layout. To understand whether performance drops are driven by changes in layout, we analyzed model performance in the most difficult generalization setting (LIMITEDPUBLISHER with no few-shot adaptation). We examined whether generalization errors typically occurred for categories for which layout changes the most. Figure 4 shows the performance drop between in-distribution and out-of-distribution settings for each structural category. Categories with the largest performance drops are those which are often characterized by spatial location, such as PAGE_NUM (-51.9 F1), BIBLIOGRAPHIC_INFO (-25.0 F1), and ACKNOWLEDGEMENTS (-22.7 F1). In contrast, much smaller performance drops occurred in categories containing the main textual content of the paper, such as BODY (-9.2 F1) and ABSTRACT (-14.1 F1).

6 Conclusion

This work studies whether layout-infused models are robust to layout distribution shift. We present a method for evaluating robustness to layout distribution shift, and construct GROTOAP2-LDS, a new set of splits for the GROTOAP2 dataset that evaluate model robustness to layout distribution shifts. We use GROTOAP2-LDS to evaluate a set of existing layout-infused models (LayoutLM, LayoutLMv2, and LayoutLMv3), and compare against two text-only models (BERT, SciBERT).

Layout-infused models perform most accurately on documents with familiar layouts (Table 3), but

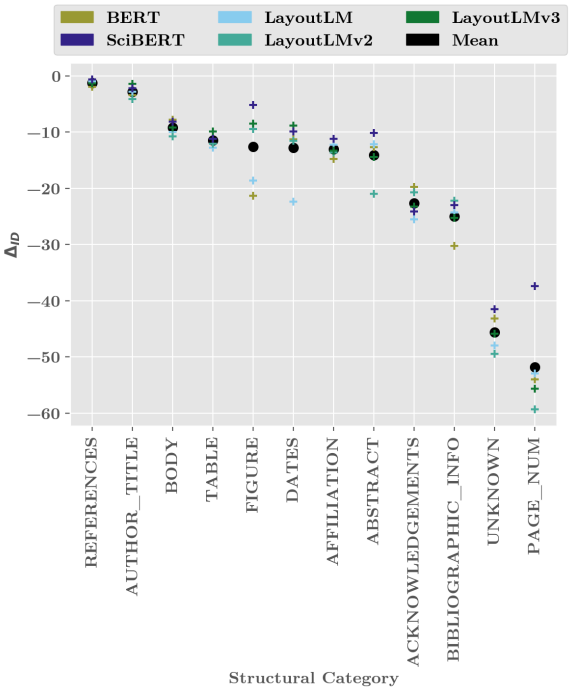

Figure 4: **Performance drop per category, low-diversity training.** Performance drop (Δ_{ID}) between in-distribution and out-of-distribution layouts for each structural category. Largest performance drops occur for categories characterized by spatial page location (e.g., PAGE_NUM, BIBLIOGRAPHIC_INFO). In contrast, much smaller performance drops occured in categories that contain the main textual content of the paper (e.g., BODY, ABSTRACT)

in settings where models must generalize to documents with unfamiliar layouts, layout-infused models underperform text-only models such as SciBERT (Table 4). In such settings, models with in-domain text pretraining both provide more accurate results, and obviate the inference time cost of processing visual layout features (e.g., image embeddings LayoutLMv3 increase inference time by $\approx 10\times$).

We hypothesize that layout-specific components overfit more because they receive less pretraining data compared to text-only components, or because they increase total model parameter count (e.g., LayoutLM, LayoutLMv2, and LayoutLMv3 contain 20-45% more parameters than BERT and SciBERT).[8] Future work could test whether larger-scale pretraining improves robustness of layout-specific components.

[8]We note that the discrepancy in generalization performance is not driven by the proportion of UNK tokens. All tokenizers produced fewer than 0.3% UNK tokens, and the tokenizer for SciBERT in fact had more UNK tokens than the tokenizer for BERT and the LayoutLM models.

We show that training strategies such as increasing training diversity or few-shot adaptation to the target layout distribution can mitigate the performance drop across layout distribution shifts. These results provide guidance for curating training data and highlight the importance during data collection of curating examples that reflect variation in document provenance. In situations with a known change in layout distribution (e.g., if a system trained on papers from one publisher is re-used to process papers from a new publisher), the cost of annotating a few examples from the target distribution may be highly effective, resulting in a large improvement in out-of-distribution model performance.

This work highlights the importance of considering layout distribution shifts when evaluating models on tasks involving visually-rich documents such as scientific papers. We hope that our study and evaluation methodology facilitate the development of layout-infused models that can generalize across layout distribution shifts.

7 Limitations

We use scientific papers as a first testbed for evaluating model robustness to layout distribution shifts. Many different layouts exist among scientific papers, and the existence of metadata databases facilitated the construction of train-test splits with layout distribution shifts. However, scientific papers are only one domain in which layout distribution shifts occur. Layouts also vary for many other visually-rich documents, such as business forms, receipts, webpages, and newspapers. We hope our evaluation methodology engenders evaluations on a wider range of document types.

Our experiments involve a subset of the many layout-infused models proposed in recent work (e.g., Peng et al., 2022; Kim et al., 2021; Li et al., 2021). The models in our experiments were chosen because they share a similar model size and underlying architecture, facilitating comparisons between different methods of layout-infusion. We release our evaluation suite to enable more comprehensive evaluations in the future.

Performance drops occur both for layout-infused and, to a lesser extent, text-only models. The performance drops from text-only models may be due to layout information conveyed via word order and visual section boundary markers, but may also reflect shifts in text distribution. Our error analy-

ses suggest that generalization errors are driven by shifts in layout rather than content (Section 5.5). In the future, synthetic experiments (e.g., with LaTeX-based manipulations) would help to fully disentangle the effects of layout and content distribution shifts, provided that large-scale synthetic manipulations can be constrained to produce realistic layouts.

8 Potential Risks

Although we do not foresee direct harms from this work, our work is related to automated processing of scientific documents. This line of study carries the risk of inaccurately processing documents and propagating false information about scientific findings.

9 Acknowledgements

This work was supported in part by NSF Grant 2033558. CC was supported in part by an IBM PhD Fellowship.

References

Fan Bai, Alan Ritter, and Wei Xu. 2021. Pre-train or annotate? domain adaptation with a constrained budget. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5002–5015, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: Pretrained language model for scientific text. volume abs/1903.10676.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Filip Grali'nski, Tomasz Stanislawek, Anna Wr'oblewska, Dawid Lipi'nski, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and P. Biecek. 2020. Kleister: A novel task for information extraction involving long documents with complex layout. *ArXiv*, abs/2003.02356.

Po-Wei Huang, Abhinav Ramesh Kashyap, Yanxia Qin, Yajing Yang, and Min-Yen Kan. 2022a. Lightweight contextual logical structure recovery. In *Proceedings of the Third Workshop on Scholarly Document Processing*, pages 37–48, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022b. Layoutlmv3: Pre-training for document ai with unified text and image masking. *ArXiv preprint*, abs/2204.08387.

Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 2, pages 1–6. IEEE.

Geewook Kim, Teakgyu Hong, Moonbin Yim, Jeongyeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. 2021. Ocr-free document understanding transformer. In *European Conference on Computer Vision*.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran S. Haque, Sara M. Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. 2021. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 5637–5664. PMLR.

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

Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I. Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. 2021. Selfdoc: Self-supervised document representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 5652–5660. Computer Vision Foundation / IEEE.

John Miller, Karl Krauth, Benjamin Recht, and Ludwig Schmidt. 2020. The effect of natural distribution shift on question answering models. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6905–6916. PMLR.

Laura Nguyen, Thomas Scialom, Benjamin Piwowarski, and Jacopo Staiano. 2023. Loralay: A multilingual

and multimodal dataset for long range and layout-aware summarization. In *Conference of the European Chapter of the Association for Computational Linguistics*.

Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. 2019. Cord: a consolidated receipt dataset for post-ocr parsing. In *Workshop on Document Intelligence at NeurIPS 2019*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Qiming Peng, Yinxu Pan, Wenjin Wang, Bin Luo, Zhenyu Zhang, Zhengjie Huang, Teng Hu, Weichong Yin, Yongfeng Chen, Yin Zhang, Shi Feng, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2022. Ernie-layout: Layout knowledge enhanced pre-training for visually-rich document understanding. *ArXiv*, abs/2210.06155.

Birgit Pfitzmann, Christoph Auer, Michele Dolfi, Ahmed S Nassar, and Peter WJ Staar. 2022. Doclaynet: A large human-annotated dataset for document-layout analysis. *ArXiv preprint*, abs/2206.01062.

Zejiang Shen, Kyle Lo, Lucy Lu Wang, Bailey Kuehl, Daniel S. Weld, and Doug Downey. 2022. VILA: Improving structured content extraction from scientific PDFs using visual layout groups. *Transactions of the Association for Computational Linguistics*, 10:376–392.

Dominika Tkaczyk, Pawel Szostek, and Lukasz Bolikowski. 2014. Grotoap2—the methodology of creating a large ground truth dataset of scientific articles. *D-Lib Magazine*, 20(11/12).

Zilong Wang, Yichao Zhou, Wei Wei, Chen-Yu Lee, and Sandeep Tata. 2022. A benchmark for structured extractions from complex documents. *ArXiv*, abs/2211.15421.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

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

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. Layoutlm: Pre-training of text and layout for document image understanding. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1192–1200. ACM.

Qinyuan Ye, Bill Yuchen Lin, and Xiang Ren. 2021. CrossFit: A few-shot learning challenge for cross-task generalization in NLP. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7163–7189, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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)*, pages 1015–1022. IEEE.

Appendix

Experiment Compute Details

All experiments were run on NVIDIA RTX A6000 GPUs. For each training condition, model training took around one day for LayoutLMv2 and LayoutLMv3, and took a couple hours for the other three models (LayoutLM, BERT, and SciBERT).

Learning Rate Selection

To select the learning rate for each model, models were trained on each of three learning rates (1e-04, 1e-05, 1e-06), and the learning rate that produced the best performance on the dev set was selected. Learning rate selection was performed separately for each of the training stages: the initial stage of training on the larger set of out-of-distribution layouts, and few-shot fine-tuning on examples from the target distribution. Dev performance for each model is shown for the initial training stage (Table 6) and few-shot fine-tuning (Table 7).

Base Model	$1e-04$	$1e-05$	$1e-06$
BERT	97.067	97.817	96.942
LayoutLM	93.964	98.066	97.308
LayoutLMv2	6.326	98.006	97.171
LayoutLMv3	6.326	98.314	97.794
SciBERT	96.644	98.184	97.902

Table 6: Learning rate selection for the initial training stage. Dev performance for each learning rate

Base Model	$1e-04$	$1e-05$	$1e-06$
BERT	78.408	97.301	97.395
LayoutLM	79.399	97.685	97.839
LayoutLMv2	79.358	97.869	97.772
LayoutLMv3	80.731	97.5	97.813
SciBERT	77.417	97.733	98.01

Table 7: Learning rate selection for few-shot fine-tuning. Dev performance for each learning rate

Dataset Details

Our set of new train-test splits, GROTOAP2-LDS, is an adaptation of data released in the GROTOAP2 dataset. GROTOAP2 is distributed under the CC-BY license. Further use of GROTOAP2-LDSshould attribute original dataset collection to (Tkaczyk et al., 2014). We refer the reader to (Tkaczyk et al., 2014) for details of the original data collection procedure.

Label remapping

Although a shared annotation procedure was used to label all papers in the GROTOAP2 dataset (Tkaczyk et al., 2014), some differences XML formatting for different publishers resulted in discrepancies between structural category labels used for different publishers. For instance, in the original GROTOAP2 dataset TITLE_AUTHOR labels are used for some publishers, whereas separate TITLE and AUTHOR labels are used for other publishers). To account for minor annotation discrepancies between publishers as well as insufficient support for certain category labels in our dataset splits, we re-map the structural category tagset used in the original GROTOAP2 dataset. Our label re-maping is shown in Table 8.

New label	Original labels
BIB_INFO	BIB_INFO, COPYRIGHT
TITLE_AUTHOR	AUTHOR, TITLE, TITLE_AUTHOR
AFFILIATION	AFFILIATION, CORRESPONDENCE
UNKNOWN	KEYWORDS, GLOSSARY, EQUATION, TYPE, EDITOR, CONFLICT_STATEMENT, UNKNOWN

Table 8: Label remapping for the GROTOAP2 tagset.

Full results

We provide the test performance for each trial and episode in Tables 9, 10, 13, 11, 14, 12, and 15.

Test Publisher	Base Model	Seed 0	Seed 1	Seed 2
ACTA	BERT	87.96	86.05	86.34
	LayoutLM	86.38	86.82	86.92
	LayoutLMv2	86.44	88.02	87.75
	LayoutLMv3	88.26	88.63	89.1
	SciBERT	86.02	88.11	88.59
BMC	BERT	95.66	96.4	95.44
	LayoutLM	96.13	96.25	96.03
	LayoutLMv2	96.5	96.58	94.24
	LayoutLMv3	94.94	95.94	96.48
	SciBERT	96.39	96.34	95.97
RU	BERT	95.49	95.71	94.94
	LayoutLM	96.66	96.87	96.39
	LayoutLMv2	96.68	96.66	96.3
	LayoutLMv3	96.29	96.48	96.3
	SciBERT	95.45	95.75	96.34
PLOS	BERT	97.34	97.07	97.31
	LayoutLM	98.01	97.92	97.58
	LayoutLMv2	97.44	96.97	97.74
	LayoutLMv3	97.77	97.42	97.24
	SciBERT	97.38	97.77	96.76

Table 9: Model performance for each random seed, in-distribution training

Test Publisher	Base Model	Seed 0	Seed 1	Seed 2
ACTA	BERT	51.46	53.52	50.69
	LayoutLM	52.85	50.25	50.36
	LayoutLMv2	57.09	54.69	55.7
	LayoutLMv3	59.87	55.58	52.02
	SciBERT	60.37	60.59	61.03
BMC	BERT	71.55	74.37	72.31
	LayoutLM	75.54	74.42	74.55
	LayoutLMv2	70.65	74.28	74.85
	LayoutLMv3	73.63	75.23	75.62
	SciBERT	78.47	76.58	79.27
RU	BERT	81.03	85.93	84.07
	LayoutLM	84.97	81.34	82.42
	LayoutLMv2	82.49	85.4	76.98
	LayoutLMv3	83.04	85.87	83.49
	SciBERT	88.36	87.3	86.44
PLOS	BERT	89.09	90.2	90.0
	LayoutLM	84.9	88.17	89.58
	LayoutLMv2	90.4	87.76	85.43
	LayoutLMv3	91.56	91.09	91.04
	SciBERT	88.25	89.9	89.8

Table 10: Model performance for each random seed, LIMITEDPUBLISHER training

Test Publisher	Base Model	Seed 0	Seed 1	Seed 2
ACTA	BERT	61.6	57.36	59.53
	LayoutLM	58.8	62.31	59.99
	LayoutLMv2	60.4	59.15	58.22
	LayoutLMv3	60.01	55.43	57.11
	SciBERT	62.62	62.94	64.25
BMC	BERT	94.1	94.14	95.18
	LayoutLM	93.53	93.71	94.42
	LayoutLMv2	94.01	93.26	93.9
	LayoutLMv3	95.01	95.32	94.86
	SciBERT	94.67	94.56	93.81
RU	BERT	92.15	90.85	92.06
	LayoutLM	91.43	91.32	91.28
	LayoutLMv2	92.65	90.96	90.29
	LayoutLMv3	90.87	91.08	90.49
	SciBERT	92.52	92.68	92.26
PLOS	BERT	88.0	87.94	86.17
	LayoutLM	84.52	83.59	85.66
	LayoutLMv2	86.01	84.43	86.59
	LayoutLMv3	86.61	86.52	87.33
	SciBERT	89.09	92.61	90.44

Table 11: Model performance for each random seed, LIMITEDPUBLISHER+ training

Test Publisher	Base Model	Seed 0	Seed 1	Seed 2
ACTA	BERT	59.59	57.42	59.48
	LayoutLM	60.48	58.04	57.37
	LayoutLMv2	61.31	55.37	56.47
	LayoutLMv3	54.78	59.63	56.45
	SciBERT	61.26	64.0	65.55
BMC	BERT	94.82	94.64	95.16
	LayoutLM	94.12	94.51	93.7
	LayoutLMv2	93.28	94.4	93.65
	LayoutLMv3	94.44	94.4	94.03
	SciBERT	93.54	94.75	93.68
RU	BERT	92.44	92.37	92.57
	LayoutLM	93.09	92.29	92.67
	LayoutLMv2	92.87	92.15	91.68
	LayoutLMv3	93.31	93.49	93.06
	SciBERT	91.68	91.91	92.57
PLOS	BERT	86.62	87.41	85.24
	LayoutLM	85.13	85.54	84.8
	LayoutLMv2	88.57	87.84	85.45
	LayoutLMv3	87.95	87.84	86.62
	SciBERT	92.54	93.69	93.1

Table 12: Model performance for each random seed, LIMITEDPUBLISHER++ training

Test Publisher	Base Model	Seed 0 Ep 0	Seed 0 Ep 1	Seed 0 Ep 2	Seed 1 Ep 0	Seed 1 Ep 1	Seed 1 Ep 2	Seed 2 Ep 0	Seed 2 Ep 1	Seed 2 Ep 2
	BERT	79.22	79.93	80.12	79.48	77.56	80.07	78.95	78.0	77.9
	LayoutLM	82.48	81.22	80.35	80.44	79.23	78.49	81.48	80.36	80.35
ACTA	LayoutLMv2	81.46	82.2	80.92	81.55	80.87	79.63	82.11	82.75	81.52
	LayoutLMv3	81.49	79.56	80.22	80.23	79.77	79.98	79.96	77.99	79.76
	SciBERT	80.78	81.44	79.62	79.6	80.5	79.88	81.05	79.14	80.73
	BERT	90.14	90.78	92.54	92.84	91.44	93.11	92.89	92.06	93.09
	LayoutLM	92.46	93.07	93.6	94.51	93.76	94.15	92.09	92.91	93.35
BMC	LayoutLMv2	93.35	93.21	93.81	95.58	93.97	94.71	94.77	94.2	94.41
	LayoutLMv3	93.15	92.53	93.73	93.48	93.48	94.13	92.37	92.49	93.38
	SciBERT	94.36	93.38	94.44	93.76	93.06	94.28	93.44	92.73	94.23
	BERT	91.74	91.77	91.5	91.29	91.66	91.62	91.66	91.69	91.22
	LayoutLM	90.86	90.0	90.56	91.28	91.12	91.01	90.93	90.18	90.49
RU	LayoutLMv2	92.08	92.62	91.5	91.96	91.96	92.44	91.78	92.48	91.06
	LayoutLMv3	91.98	92.01	91.88	90.96	90.79	90.28	92.85	92.02	91.99
	SciBERT	92.02	90.99	91.18	92.28	91.89	91.63	92.44	91.74	91.92
	BERT	95.94	95.42	96.69	96.15	95.41	96.52	96.07	94.86	95.88
	LayoutLM	96.91	96.18	95.92	96.08	95.43	96.05	96.81	95.56	95.38
PLOS	LayoutLMv2	97.12	96.07	95.93	96.63	95.35	95.61	96.86	95.6	96.18
	LayoutLMv3	96.91	96.23	96.5	96.7	96.43	96.15	96.74	95.76	95.96
	SciBERT	96.12	94.96	96.69	96.57	95.54	96.55	96.74	96.13	96.4

Table 13: Model performance for each random seed and few-shot episode (Ep), LIMITEDPUBLISHER training, after few-shot fine-tuning

Test Publisher	Base Model	Seed 0 Ep 0	Seed 0 Ep 1	Seed 0 Ep 2	Seed 1 Ep 0	Seed 1 Ep 1	Seed 1 Ep 2	Seed 2 Ep 0	Seed 2 Ep 1	Seed 2 Ep 2
	BERT	82.88	84.03	81.67	80.18	79.9	80.53	80.28	79.06	79.37
	LayoutLM	81.29	79.49	79.44	82.36	80.96	80.92	81.07	78.28	80.12
ACTA	LayoutLMv2	84.51	84.27	84.23	81.58	83.99	83.47	81.52	84.46	84.03
	LayoutLMv3	81.39	81.96	83.05	81.47	81.41	81.27	81.85	84.68	80.46
	SciBERT	83.78	82.18	83.42	83.32	82.14	82.1	80.78	80.63	79.7
	BERT	94.76	94.67	94.72	94.67	94.38	94.23	95.44	94.79	95.14
	LayoutLM	94.83	94.79	94.43	95.24	94.8	95.46	95.05	95.08	95.19
BMC	LayoutLMv2	95.19	95.24	95.54	95.64	95.14	95.58	95.41	94.02	94.99
	LayoutLMv3	94.84	95.1	95.24	95.25	94.91	95.74	94.76	94.22	95.34
	SciBERT	95.3	94.12	95.12	95.27	94.52	95.52	94.63	94.37	95.33
	BERT	92.92	93.29	92.81	92.92	92.99	93.02	93.38	93.41	92.42
	LayoutLM	93.05	93.72	92.92	93.85	94.35	93.36	93.28	93.8	91.86
RU	LayoutLMv2	94.1	93.92	93.61	93.49	94.04	93.5	93.26	93.71	92.9
	LayoutLMv3	93.34	93.83	93.25	93.8	93.58	93.31	93.75	94.02	93.85
	SciBERT	94.37	93.95	93.51	94.0	93.2	93.8	93.72	94.06	93.72
	BERT	96.51	96.18	96.49	96.34	95.9	95.47	96.47	96.43	96.52
	LayoutLM	96.94	96.09	96.11	97.03	96.89	96.29	96.75	96.4	96.85
PLOS	LayoutLMv2	96.77	96.08	95.4	97.02	95.86	95.25	96.77	96.03	94.93
	LayoutLMv3	97.61	96.6	97.06	97.1	96.76	97.05	97.42	96.43	96.22
	SciBERT	97.09	96.36	96.97	96.88	96.73	96.97	96.88	96.53	96.84

Table 14: Model performance for each random seed and few-shot episode (Ep), LIMITEDPUBLISHER+ training, after few-shot fine-tuning

Test Publisher	Base Model	Seed 0 Ep 0	Seed 0 Ep 1	Seed 0 Ep 2	Seed 1 Ep 0	Seed 1 Ep 1	Seed 1 Ep 2	Seed 2 Ep 0	Seed 2 Ep 1	Seed 2 Ep 2
	BERT	79.79	80.22	81.14	79.9	79.41	78.29	80.48	81.22	79.48
	LayoutLM	82.53	80.18	80.65	80.6	79.25	80.95	80.87	79.53	79.16
ACTA	LayoutLMv2	81.27	82.57	83.29	81.24	82.54	80.56	81.2	83.69	83.68
	LayoutLMv3	81.53	82.74	81.06	82.73	82.67	80.9	81.44	82.71	81.14
	SciBERT	81.81	82.98	82.37	81.8	82.17	82.74	80.64	82.03	81.26
	BERT	95.21	94.97	94.97	95.58	94.88	95.2	95.65	95.08	95.33
	LayoutLM	94.93	95.34	95.38	95.11	94.55	94.86	93.82	94.12	94.78
BMC	LayoutLMv2	95.04	93.33	95.47	95.09	94.94	94.82	94.63	94.68	94.66
	LayoutLMv3	95.57	94.54	94.58	95.22	93.77	94.67	95.03	93.2	94.02
	SciBERT	94.86	94.15	94.56	95.32	94.82	95.07	94.98	93.34	94.66
	BERT	93.36	93.27	92.78	93.39	93.55	93.03	93.48	93.67	93.65
	LayoutLM	94.23	93.75	92.89	93.96	93.8	92.71	93.86	93.97	93.06
RU	LayoutLMv2	93.95	93.56	92.86	94.31	93.68	93.18	94.04	93.67	93.33
	LayoutLMv3	93.94	94.73	93.8	94.04	94.07	93.89	94.4	93.97	93.74
	SciBERT	94.03	93.09	92.12	93.74	93.2	93.48	94.0	93.97	93.35
	BERT	96.2	95.5	95.81	96.41	95.66	95.36	96.54	96.6	96.47
	LayoutLM	96.92	97.02	97.07	96.57	96.34	96.22	97.2	96.52	96.51
PLOS	LayoutLMv2	96.42	95.01	95.13	96.7	96.19	95.71	97.19	95.84	95.01
	LayoutLMv3	97.03	96.27	96.25	96.53	96.06	96.37	96.79	95.7	95.66
	SciBERT	97.06	96.3	96.58	97.3	96.76	97.12	96.95	96.42	96.76

Table 15: Model performance for each random seed and few-shot episode (Ep), LIMITEDPUBLISHER++ training, after few-shot fine-tuning

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 7

☑ A2. Did you discuss any potential risks of your work?
Section 8

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

We present a new set of train-set splits for Grotoap2. We describe these splits in Section 3.3.

☑ B1. Did you cite the creators of artifacts you used?
Yes, we cite Grotoap2 throughout the paper, including in Section 3 (Evaluation Methodology).

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Yes, we discuss the Grotoap2 license in the Appendix.

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Yes, we discuss the Grotoap2 license in the Appendix.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
The examples in Grotoap2 are from scientific papers in the public domain; we did not perform further anonymization steps.

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
We did not provide documentation of the original Grotoap2 dataset, but we refer the reader to Tkaczyk et al., 2014 for details of the original data collection procedure.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
We describe statistics of train-test splits in Section 3.3.

C ☑ Did you run computational experiments?

Section 4, Section 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
We report the computational cost and infrastructure in the Appendix.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
We discuss the experimental setup and hyperparameter search procedure in Section 3 and in the Appendix.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
We describe descriptive statistics in each of our tables and figures.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.