Previous datasets for evaluating PDF content extraction rely on machine-generated labels of imperfect quality, and comprise papers from a limited range of scientific disciplines. To better evaluate our proposed methods, we design a new benchmark suite, Semantic Scholar Visual Layout-enhanced Scientific Text Understanding Evaluation (**S2-VLUE**). The benchmark extends two existing resources (Tkaczyk et al., 2015; Li et al., 2020) and introduces a newly curated dataset, S2-VL, which contains high-quality human annotations for papers across 19 disciplines.

Our contributions are as follows:

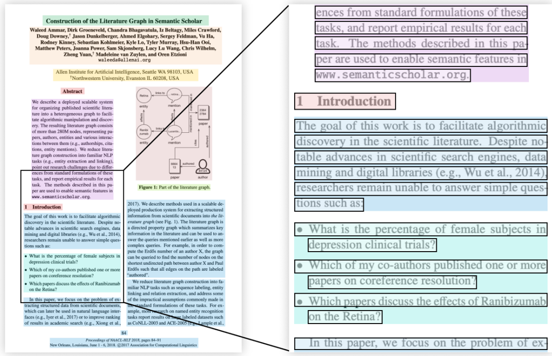
1. We introduce a new strategy for PDF content extraction that uses VILA structures to inject layout information into language models, and show that this improves accuracy *without* the expensive pretraining required by existing methods, and generalizes to different language models.
2. We design two models that incorporate VILA features differently. The I-VILA model injects layout indicator tokens into the input texts and improves prediction accuracy (up to +1.9% Macro F1) and consistency compared to the previous layout-augmented language model LayoutLM (Xu et al., 2020). The H-VILA model performs group-level predictions and can reduce model inference time by 47% with less than 0.8% loss in Macro F1.
3. We construct a unified benchmark suite S2-VLUE which enhances existing datasets with VILA structures, and introduce a novel dataset S2-VL that addresses gaps in existing resources. S2-VL contains hand-annotated gold labels for 15 token categories on papers spanning 19 disciplines.

The benchmark datasets, modeling code, and trained weights are available at `https://github.com/allenai/VILA`.

2 Related Work

2.1 Structured Content Extraction for Scientific Documents

Prior work on structured content extraction for scientific documents usually relies on textual or visual features. Text-based methods like ScienceParse (Ammar et al., 2018), GROBID (GRO, 2008–2021) or Corpus Conversion Service (Staar et al., 2018) combine PDF-to-text parsing engines like CERMINE (Tkaczyk et al., 2015) or pdfalto,[1] which output a sequence of tokens extracted from a PDF, with machine learning models like RNN (Hochreiter and Schmidhuber, 1997), CRF (Lafferty et al., 2001), or Random Forest (Breiman, 2001) trained to classify the token categories of the sequence. Though these models are practical and fairly efficient, they fall short in prediction accuracy or generalize poorly to out-of-domain documents. Vision-based Approaches (Zhong et al., 2019; He et al., 2017; Siegel et al., 2018), on the other hand, treat the parsing task as an image object detection problem: given document images, the models predict rectangular bounding boxes, segmenting the page into individual components of different categories. These models excel at capturing complex visual layout structures like figures or tables, but because they operate only on visual signals without textual information, they cannot accurately predict fine-grained semantic categories like title, author, or abstract, which are of central importance for parsing scientific documents.



(a) Paper PDF Screenshot
The text blocks are highlighted in colored boxes.

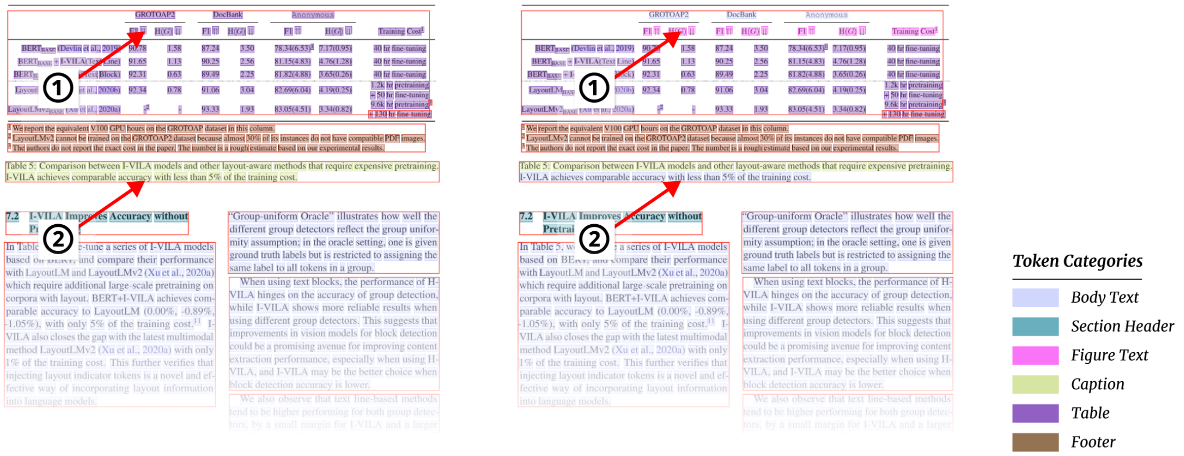
(b) Examples of VIsual LAyout Groups
Tokens in the same text lines or blocks usually have the same semantic category.

Figure 1: (a) Real-world scientific documents often have intricate layout structures, so analyzing only flattened raw text forfeits valuable information, yielding sub-optimal results. (b) The complex structures can be broken down into groups (text blocks or lines) that are composed of tokens with the same semantic category.

2.2 Layout-aware Language Models

Recent methods on layout-aware language models improve prediction accuracy by jointly modeling documents' textual and visual signals. LayoutLM (Xu et al., 2020) learns a set of novel posi-

[1] `https://github.com/kermitt2/pdfalto` (last accessed Jan. 1, 2022).



(a) Ground Truth/VILA Model Predictions
Micro F1: 1.00, H(G): 0.00

(b) Baseline LayoutLM Predictions
Micro F1: 0.94, H(G): 0.11

Figure 4: Model predictions for the 10th page of our paper draft. We present the token category and text block bounding boxes (highlighted in red rectangles) based on the (a) ground-truth annotations and model predictions from both I-VILA and H-VILA models (the three results happen to be identical) and (b) model predictions from the LayoutLM model. When VILA is injected, the model achieves more consistent predictions for the example, as indicated by arrows (1) and (2) in the figure. Best view in color.

3.3 H-VILA: Visual Layout-guided Hierarchical Model

The uniformity of group token categories also suggests the possibility of building a group-level classifier. Inspired by recent advances in modeling long documents, hierarchical structures (Yang et al., 2020; Zhang et al., 2019) provide an ideal architecture for the end task while optimizing for computational cost. Illustrated in Figure 3, our hierarchical approach uses two transformer-based models, one to encode each group in terms of its words, and another modeling the whole document in terms of the groups. We provide the details below.

The Group Encoder is a l_{g}-layer transformer that converts each group g_i into a hidden vector $\mathbf{h}_i$. Following the typical transformer model setting (Vaswani et al., 2017), the model takes a sequence of tokens $T^{(j)}$ within a group as input, and maps each token $T_i^{(j)}$ into a dense vector $\mathbf{e}_i^{(j)}$ of dimension d. Subsequently, a group vector aggregation function $f : R^{n_j \times d} \to R^d$ is applied that projects the token representations $\left(\mathbf{e}_1^{(j)}, \ldots, \mathbf{e}_{n_j}^{(j)}\right)$ to a single vector $\tilde{\mathbf{h}}_j$ that represents the group's textual information. A group's 2D spatial information is incorporated in the form of positional embeddings, and the final group representation $\mathbf{h}_j$ can be calculated as:

$$\mathbf{h}_j = \tilde{\mathbf{h}}_j + p(b_j). \tag{1}$$

where p is the 2D positional embedding similar to the one used in LayoutLM:

$$\begin{aligned} p(b) =& E_x(x_0) + E_x(x_1) + E_w(x_1 - x_0) + \\ & E_y(y_0) + E_y(y_1) + E_h(y_1 - y_0), \end{aligned} \tag{2}$$

where E_x, E_x, E_w, E_h are the embedding matrices for x, y coordinates and width and height. In practice, we find that injecting positional information using the bounding box of the first token within the group leads to better results, and we choose group vector aggregation function f to be the average over all tokens representations.

The Page Encoder is another stacked transformer model of l_{p} layers that operates on the group representation $\mathbf{h}_j$ generated by the group encoder. It generates a final group representation $\mathbf{s}_j$ for downstream classification. A MLP-based linear classifier is attached thereafter, and is trained to generate the group-level category probability p_{jc}.

Different from previous work (Yang et al., 2020), we restrict the choice of l_{g} and l_{p} to $\{1, 12\}$ such that we can load pre-trained weights from BERT base models. Therefore, no additional pretraining is required, and the H-VILA model can be fine-tuned directly for the downstream classification task. Specifically, we set $l_{\text{g}} = 1$ and initialize the group encoder from the first-layer transformer weights of BERT. The page encoder is configured as either a one-layer transformer or a 12-layer transformer that resembles a full LayoutLM model. Weights are initialized from the first-layer or full 12 layers of the LayoutLM model, which