A. Ethics Statement

This work does not collect or release any new data resource. Moreover, all three datasets used in experiments (Hierarchical Forms, RICO, and FUNSD) are publicly available and free to use, hence do not intrude user privacy. During the course of this work, no human judgements were exploited nor any user-level data was collected, stored or processed. Our methods do not add to any pre-existing data biases. Potential applications of this work include extracting document layout structure from forms, invoices, receipts, official documents, contractual documents, and digitizing scanned documents. Other uses of our contribution include form authoring, re-flowability across devices [14, 22], adaptive editing of semi-structured documents and user-interfaces [37], and improving accessibility for user-interactions [54]. We acknowledge that information extraction from documents is finds applications in finance, health, record-keeping, commercial businesses, and can have multiple downstream NLP applications. None of our methods generate, store, or harness any private identity information. We utilize forms that are in public domain and in English language.

B. Dataset Statistics

We train and test the `LayerDoc` model on three datasets - Hierarchical Forms, RICO and FUNSD. The data statistics are given in Table 6; class labels and label distributions are given in Table 7.

Hierarchical Forms: [1] provides a rich corpus of scanned form documents from diverse domains like automobile, insurance, finance, medical, and government agencies. The documents are human-annotated with labelled bounding boxes, element type, and element relations for a diverse set of 14 constituent elements such as Text Fields, Checkboxes, Choice Groups, Widgets, Tables, Image, Header, Footer, etc.

RICO[10] is one of the largest repositories of mobile app screens with more than 66k layout hierarchies augmented with semantic annotations of UI components. The bounding boxes, element labels and nested hierarchies are from the Android app source code.

FUNSD is a dataset of noisy scanned forms from research, marketing, and advertising documents that vary widely in their structure and appearance. The dataset consists of 199 annotated forms with 9,707 elements and 31,485 word-level annotations for 4 element types: header, question, answer, and other. This dataset differs from the rest due to shallow 2-layer hierarchies and filled form fields.

C. Reading Order Metrics

- **Average Page-level BLEU:** BLEU is a popular metric that measure the n-gram overlaps between the hypothesis and reference. Inspired by [46], we report Average Page-level BLEU (p-BLEU) which refers to the micro-average precision of n-gram overlaps within a page.
- **Average Relative Distance:** [46] introduced Average Relative Distance (ARD) metric to measure the relative distance between different sequences along with incorporating a penalty for omitted tokens. ARD provides a quantitative estimate about how far the scrambled tokens are located in the hypothesis with respect to their correct position in the reference sequence.

D. Word Grouping Metric

Word grouping is the task of aggregating words that belong to the same element, realized by link prediction. Following [24, 46], we use F1, precision and recall to evaluate the intervals in the predicted word sequence belonging to the elements as compared to the ground truth sequence.

E. Hierarchy Reconstruction Metric

Adjusted Rand-index: We utilize the Adjusted Rand-index [40], adjusted for the chance grouping of elements, to measure the similarity between two hierarchies. We consider the child-boxes in a given layer linked to the same parent-box as one cluster and consider the predicted parent-boxes to match if the ground truth if IoU > 0.5. We compute the rand-index score for each layer as well as for the whole layout hierarchy in aggregate.

F. Experimental Settings

LayoutLMv2 Settings: LayoutLMv2 applied for hierarchy reconstruction follows the DocStruct baseline by replacing the feature extraction modules with LayoutLM. We use our box proposal strategy and combine it with DocStruct methodology to perform parent-child linking one pair at a time. In order to infer word grouping, we use the constructed hierarchy to group together OCR text in the same box. Traversing the extracted hierarchy, we obtain the reading order of text tokens.

Ground Truth Labels: During training, LayerDoc evaluates the links between the proposed parent box and the child boxes in the present layer to select if the parent box is the ground truth or not. In case the parent box is the ground truth, it should group one or more child boxes. The link prediction should be positive for each corresponding parent-child pair and negative for the remaining child boxes. In case the parent box proposal is not a ground truth parent box in that layer, the link prediction should be negative for all parent-child box links. For cases where the proposed parent box is a ground truth box, the model should predict the correct type class of parent box. In all other cases it should predict the “None” class.

Dataset	Train	Validation	Test
Hierarchical Forms [1]	14780	2546	263
RICO [10]	53007	6626	6626
FUNSD [21]	100	50	50

Table 6: Train/Val/Test documents distribution for Hierarchical Forms, RICO, and FUNSD datasets.)

Hyperparameter: Hyper-parameters for `LayerDoc` were tuned on the respective validation set to find the best configurations for different datasets. We summarize the range of our model's hyper parameters such as: number of hidden layers in dense layer for semantic encoder $\{1, 2, 3\}$, multi-task weighting factor $\in$ $[0.1, 1]$, hidden state of SBERT (768), LayoutLM/ LayoutLMv2 embedding size (768), dropout $\delta \in \{0.2, 0.3, 0.4, 0.5.0.6\}$, learning rate $\lambda \in \{1e-5, 2e-5, 5e-5, 1e-4, 1e-3\}$, weight decay $\omega \in \{1e-6, 1e-5, 1e-4, 1e-3\}$, batch size $b \in \{4, 8, 16, 32, 64\}$ and epochs (≤ 20).
Semantic Cues: We use SentenceBERT[1] [41] to extract the feature representation of the text present inside an input box. SentenceBERT generates token embedding of size 1x 768.
Visual Cues: We use Detectron2[2] to extract visual cues from the input document image. Detectron2 expects the image to be resized to shape (224, 224).
Multimodal Contextual Encoder: We used LayoutLM[3] [51] or LayoutLMv2[4] [52] as backbone of the multimodal context encoder. The input to LayoutLM model comprises of bounding box co-ordinates and element type labels of the input sequence. LayoutLMv2 model additionally takes input image to extract visual features. Both models provide respective tokenizers for pre-processing input sequence. The output from the Contextual Encoder is a 1-D vector of size 768 per input box.
OCR Extraction: OCR text and bounding boxes are extracted using Tesseract OCR [44]. We use Detectron2[5] object detector for extracting the embedded icons and widgets. OCR text boxes for FUNSD dataset were obtained using Google Vision[6] as described by [21].
Loss Function and Inference: `LayerDoc` is trained end to end using (i) Categorical Cross Entropy loss with Adam optimizer for element classification, and (ii) Weighted Binary Cross Entropy with Adam for link prediction. Across all three datasets, we found the best results correspond with the use of Adam optimiser set with default values $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e-8$, weight-decay of $5e-4$ and an initial learning rate of 0.001.
Computing Infrastructue: `LayerDoc` is written in PyTorch library and was trained on multiple Nvidia GeForce RTX 2080 GPU.
Average Runtime: Each model takes a maximum of approximately 15 hours to train on either of the three datasets.
Dataset Access Links to download Hierarchical Forms dataset: `https://github.com/forms-data-structures/forms-data`
Link to download RICO dataset: `https://interactionmining.org/rico` Link to download FUNSD dataset: `https://guillaumejaume.github.io/FUNSD/`

[1] `https://www.sbert.net/`
[2] `https://github.com/facebookresearch/detectron2`
[3] `https://huggingface.co/transformers/model_doc/layoutlm.html`
[4] `https://huggingface.co/transformers/model_doc/layoutlmv2.html`
[5] `https://github.com/facebookresearch/detectron2`
[6] `https://cloud.google.com/vision/docs/handwriting`

G. Additional Results

Figure 4 in Appendix shows an aggregate layer-wise element grouping performance where the F1 score reduces as we move higher up the predicted hierarchy. Elements detected in the given layer are used for predicting elements in the next level of the hierarchy, causing error propagation to deteriorate the predictions at higher layers of the hierarchy. We acknowledge this as a drawback of our bottom-up approach.

H. Computational cost of `LayerDoc`

On an average, `LayerDoc` performs approximately $1e+4$ forward predictions per document at test time to generate the complete layout hierarchy for a form document. This is ≈10 times less than the DocStruct/LayoutLMv2 baseline which does link prediction for each possible pair in the document. `LayerDoc` takes advantage of improved contextual modeling and the natural geometric constraints of the document layout to optimize time complexity for document inference.

I. More Related Work

Reading Order and Word Grouping: Current 1D methods in information extraction [51, 52] directly utilize the output from off-the-shelf Optical Character Recognition (OCR) systems [7, 24]. Errors in OCR reading order can lead to scrambled text which can cause contiguous text fragments to be located far apart in the extracted token sequence (see Fig.1), thereby affecting the performance of downstream

Dataset	Label
FUNSD	header
	question
	answer
	other
Hierarchical Forms	header
	Widget
	TableRow
	ChoiceGroup
	ListItem
	Table
	TextRun
	TableCell
	TextBlock
	Image
	Field
	Header
	Form
RICO	header
	List Item
	Background Image
	Text
	Checkbox
	Video
	Radio Button
	Text Button
	Modal
	Toolbar
	Date Picker
	Card
	On/Off Switch
	Number Stepper
	Map View
	Bottom Navigation
	Drawer
	Icon
	Multi-Tab
	Web View
	Input
	Button Bar
	Pager Indicator
	Slider
	Advertisement
	Image
	Screen

Table 7: Label distribution for FUNSD, Hierarchical Forms, and RICO datasets.

tasks relying on the correct reading order. Errors in reading order detection also impedes grouping of word words belonging to the same semantic element in complex nested layouts. [24] explored a GCN-based approach to provide reading order encoding for text tokens. However, this approach lacked global document context and can unscramble the local reading order only for short text sequences. [46]

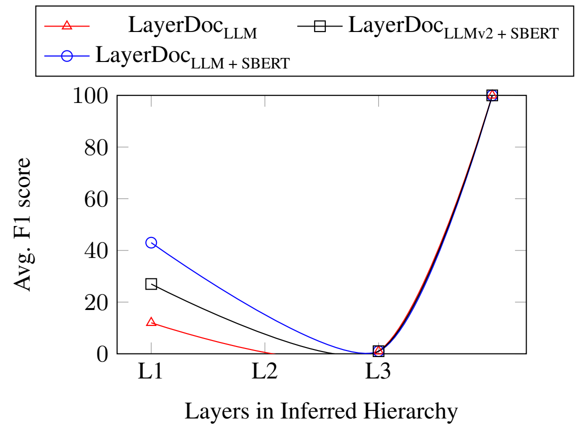


Figure 4: F1 score of **layer-wise rand-index test** to evaluate `LayerDoc` and its ablations on **Hierarchical Forms** dataset. Elements detected in the given layer are used for predicting elements in the next level of the hierarchy. F1 score reduces as we move higher up the predicted hierarchy due to recursive error propagation.

used a LayoutLM based encoder-decoder model to sequentially rearrange text tokens into reading order. However, it struggles to provide accurate reading order for visually-rich documents. [48] showed that link prediction between related structural elements can reinforce word grouping. In this work, we decompose a document into logical structural elements that group words into semantic elements to solve both reading order and grouping tasks simultaneously.

J. Reproducibility

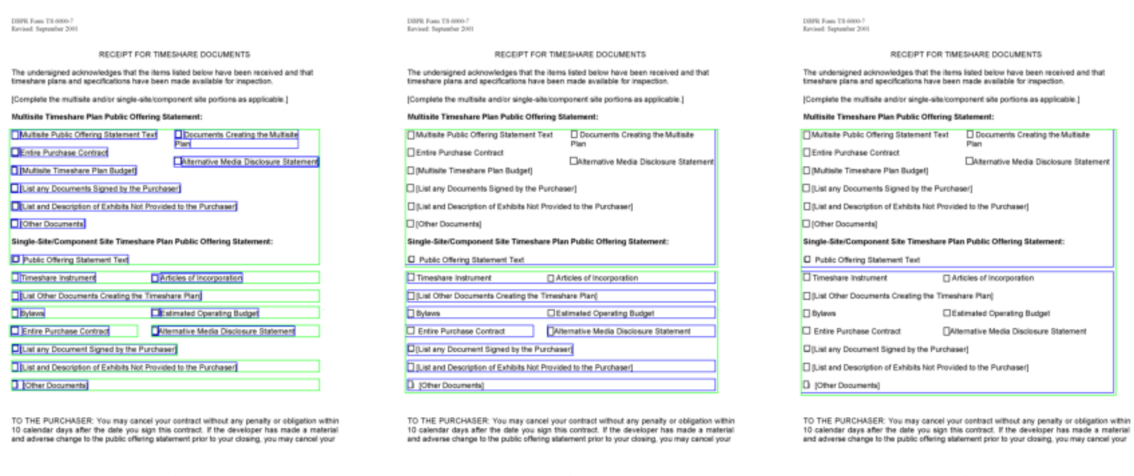
Table 8 lists the best values of the hyperparameters used in `LayerDoc` model for different data settings. We used manual search to choose the best set of training configurations across each dataset.

K. Illustrated Examples of Hierarchy Predictions

Figures 5 - 43 present some examples of the generated layer-wise hierarchies in form documents by `LayerDoc`. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it. Successive subfigures (a $\rightarrow$ d) represent higher layers in the layout hierarchy tree.

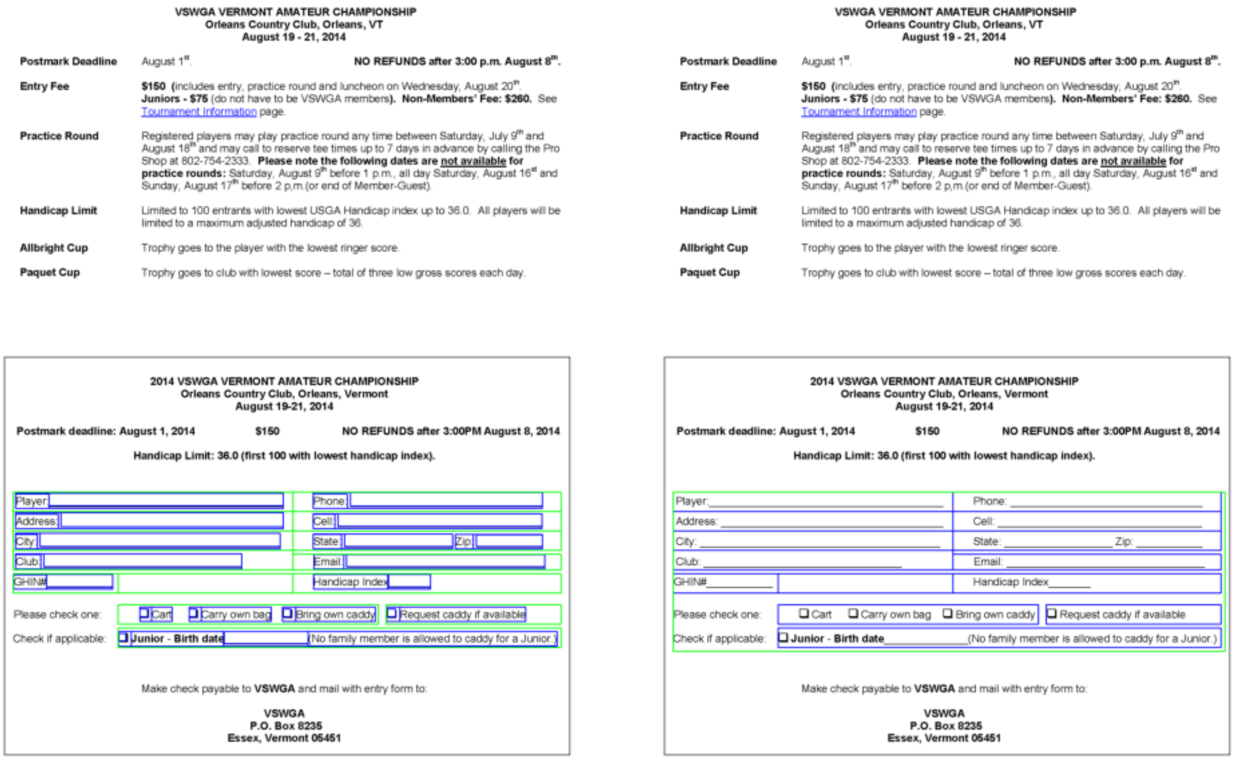
	Dataset		
Hyperparameters	Hierarchical Forms	RICO	FUNSD
Dropout Ratio	0.5	0.5	0.5
Optimizer	Adam	Adam	Adam
Output Dimension (Context Encoder)	(n,768)	(n,768).	(n,768)
Output Dimension (Semantic Encoder)	(n,768)	(n,768)	(n,768)
Input Dimension (Visual Encoder)	(n,3,224,224)	(n,3,224,224)	(n,3,224,224)
λ - LayerDoc$_{LLMv}$	0.4	0.4	0.5
λ - LayerDoc$_{LLMv+SBERT}$	0.2	0.25	0.5
λ - LayerDoc$_{LLMv2}$	0.3	0.3	0.5
λ - LayerDoc$_{LLMv2+SBERT}$	0.1	0.1	0.5
Learning Rate - LayerDoc$_{LLMv}$	1e-5	1e-5	1e-5
Learning Rate - LayerDoc$_{LLMv+SBERT}$	1e-5	1e-5	1e-5
Learning Rate - LayerDoc$_{LLMv2}$	1e-5	1e-5	1e-5
Learning Rate - LayerDoc$_{LLMv2+SBERT}$	1e-5	1e-5	1e-5
Hidden Layer Dimension (Semantic Encoder)	768x1	768x1	768x1
Epochs	10	10	10
Batch Size	16	16	16
Activation Function of Linear layers	ReLU	ReLU	ReLU
IoU match threshold	0.5	0.5	0.5
Output Classes	17	28	4
Number of parameters	200M	200M	200M
Training Samples**	100k	100k	100k
Validation Samples**	100k	100k	100k
Testing Samples**	800k	800k	800k

Table 8: **Hyperparameters Details:** Training hyperparameters of best performing `LayerDoc` model for Hierarchical Forms, RICO, and FUNSD datasets. n refers to the number of input samples; r refers to the number of total element classes. ** refers to element classification and group identification task.



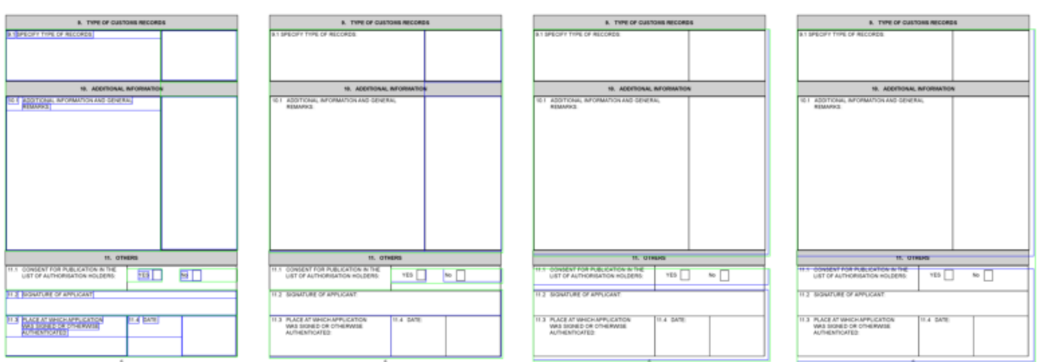
(a) (b) (c)

Figure 5: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



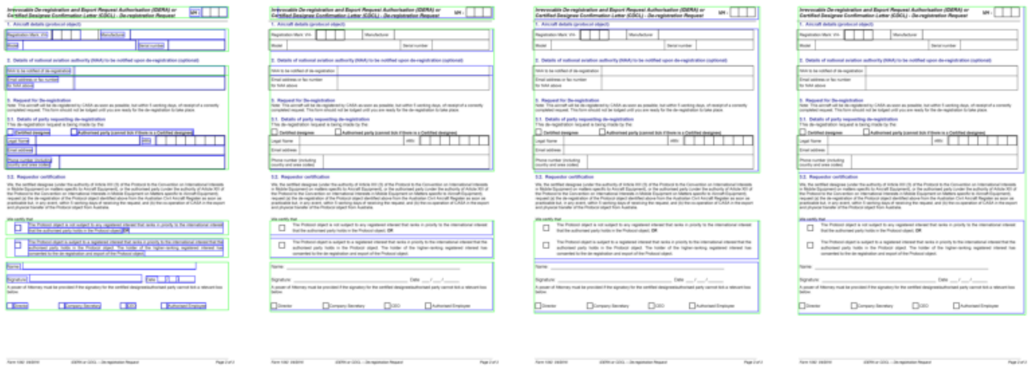
(a) (b)

Figure 6: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c) (d)

Figure 7: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

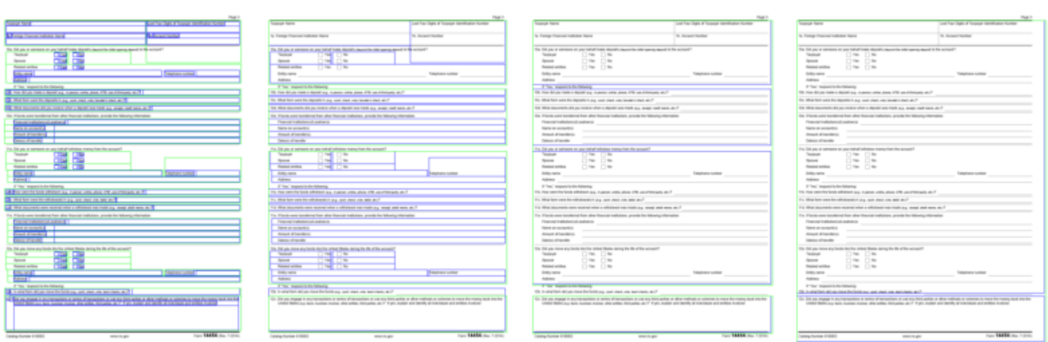


(a) (b) (c) (d)

Figure 8: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

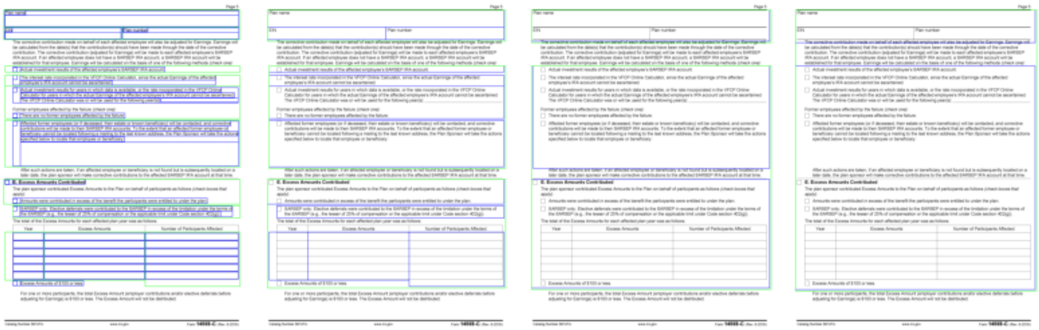
(a) (b) (c) (d)

Figure 9: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c) (d)

Figure 10: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

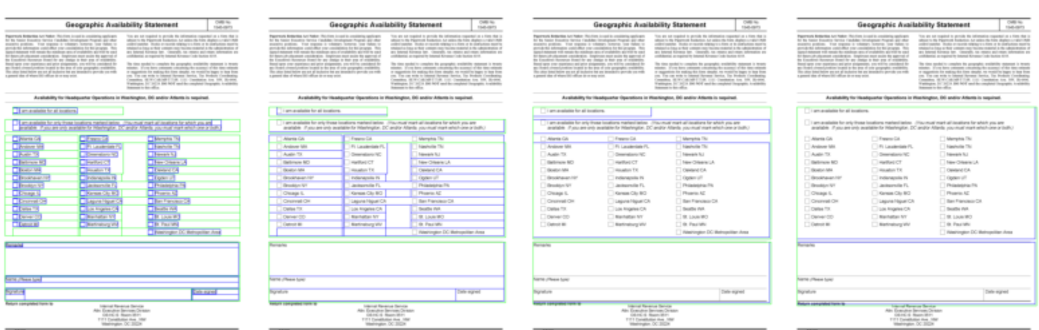


(a) (b) (c) (d)

Figure 11: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

(a) (b) (c) (d)

Figure 12: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

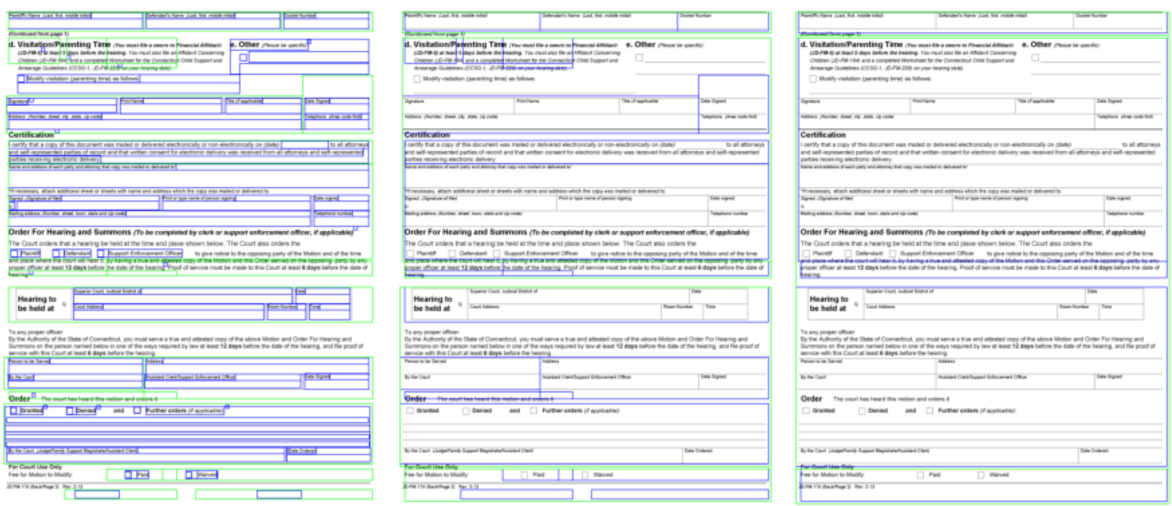


(a) (b) (c) (d)

Figure 13: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

(a) (b) (c) (d)

Figure 14: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

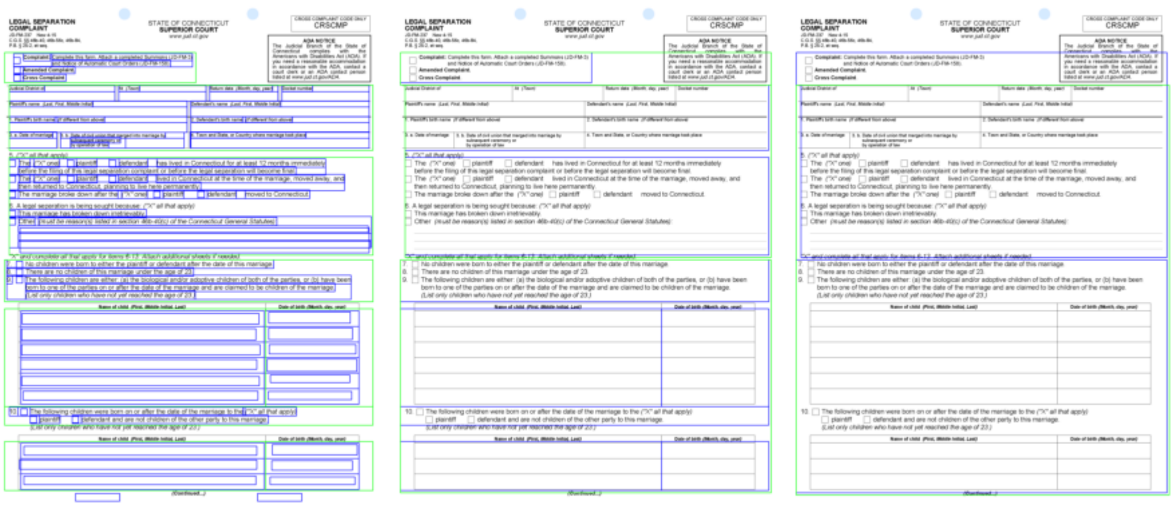


(a) (b) (c)

Figure 15: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

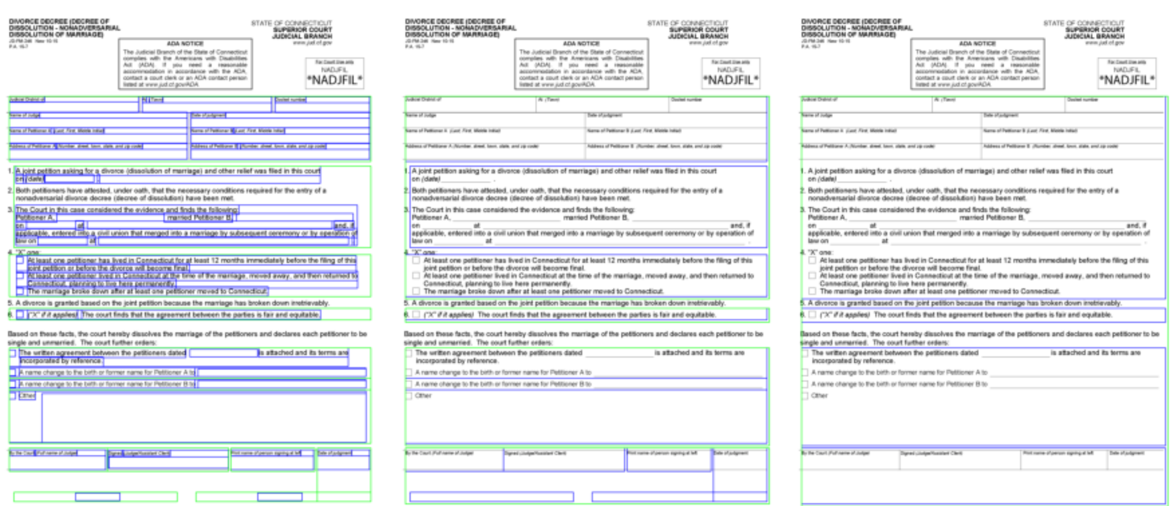
(a) (b) (c) (d)

Figure 16: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



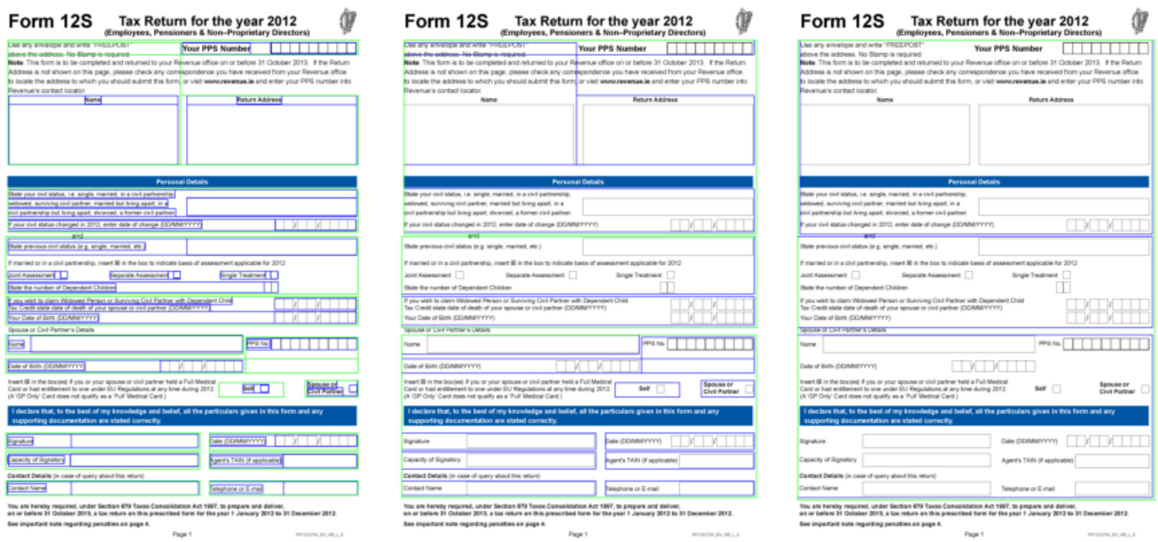
(a) (b) (c)

Figure 17: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



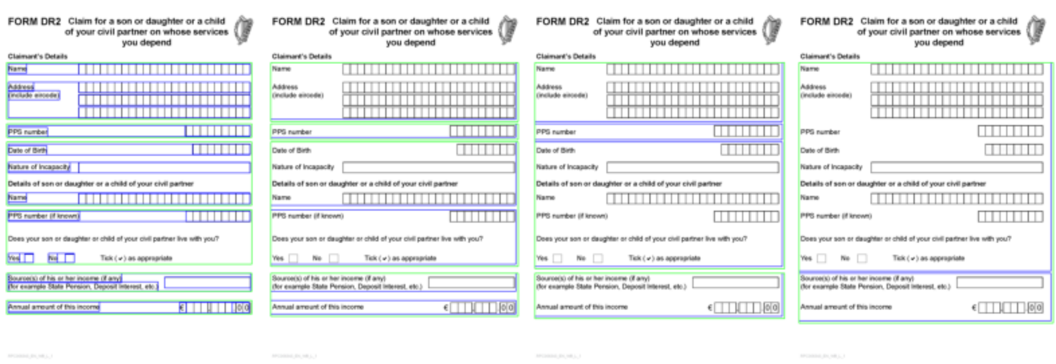
(a) (b) (c)

Figure 18: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



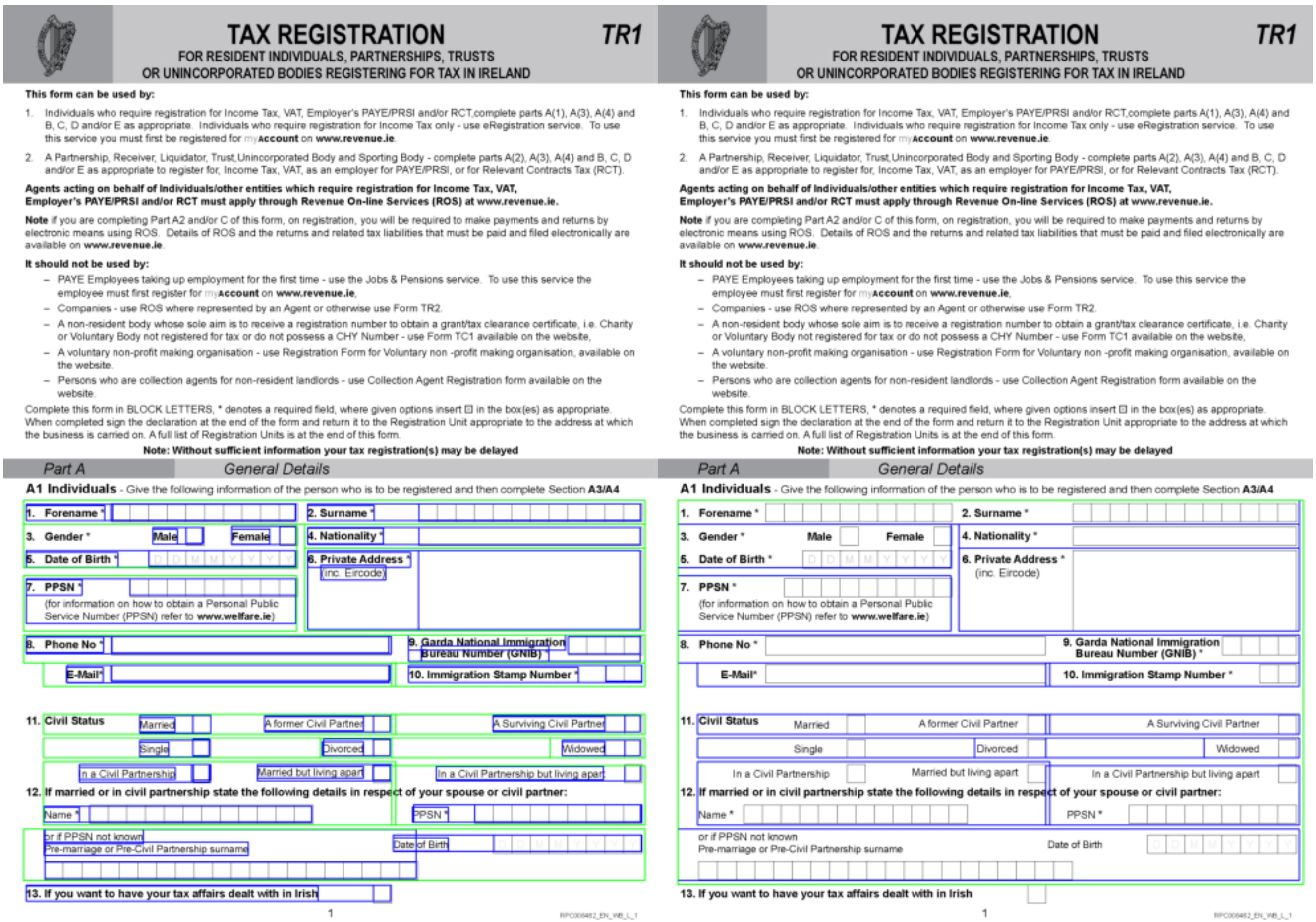
(a) (b) (c)

Figure 19: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c) (d)

Figure 20: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

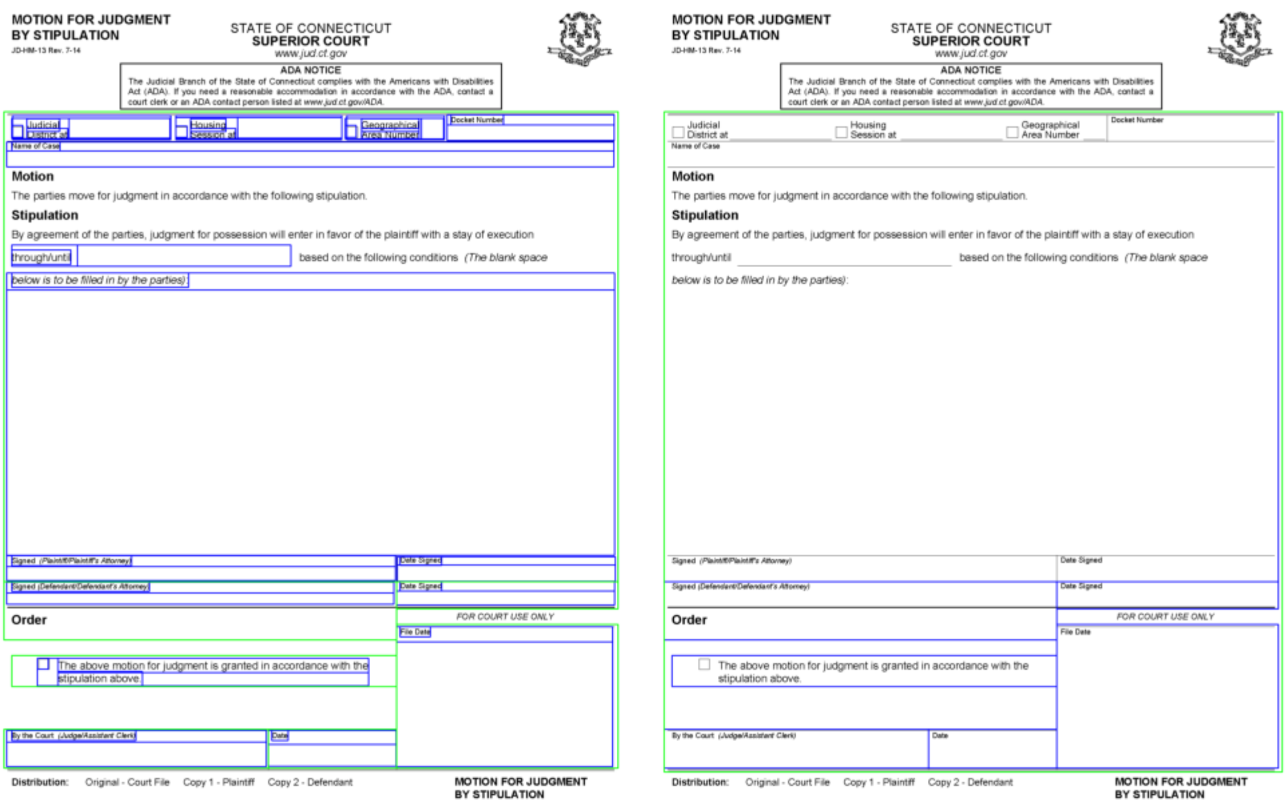


(a) (b)

Figure 21: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

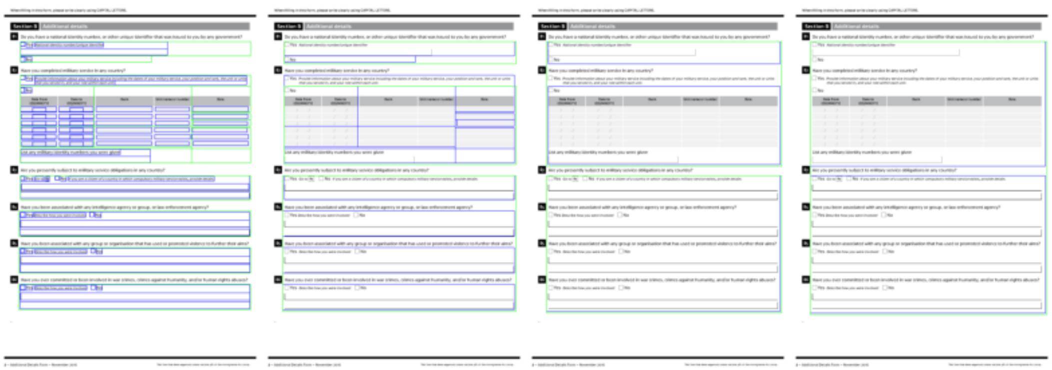
(a) (b) (c)

Figure 22: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b)

Figure 23: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c) (d)

Figure 24: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

(a) (b) (c) (d)

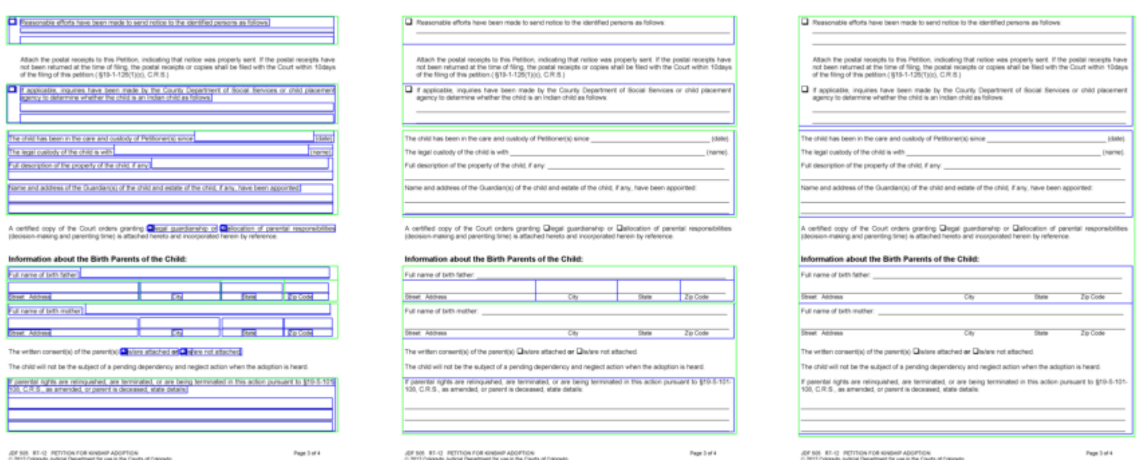
Figure 25: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

(a) (b) (c)

Figure 26: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

(a) (b) (c)

Figure 27: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c)

Figure 28: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

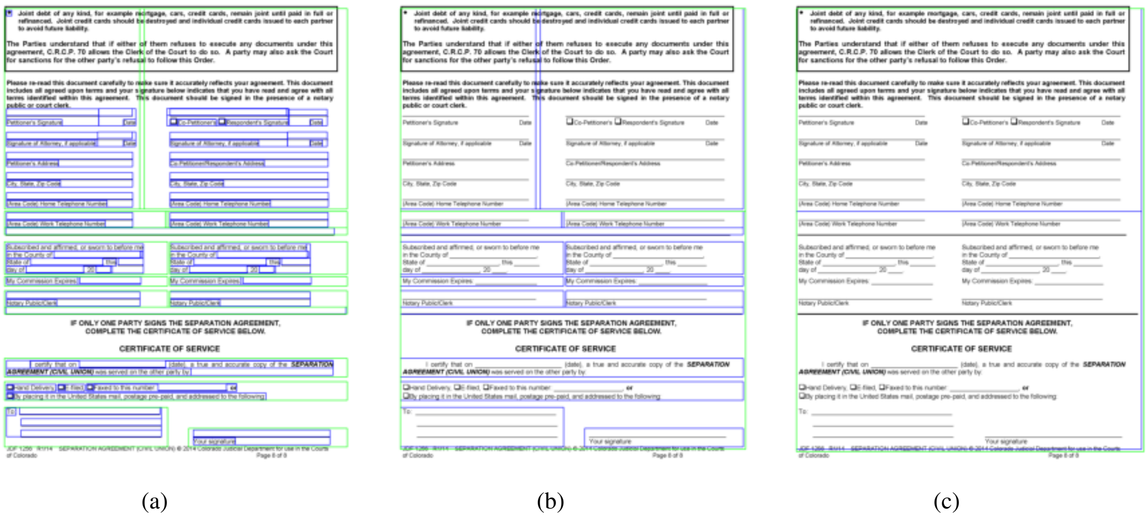


Figure 29: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

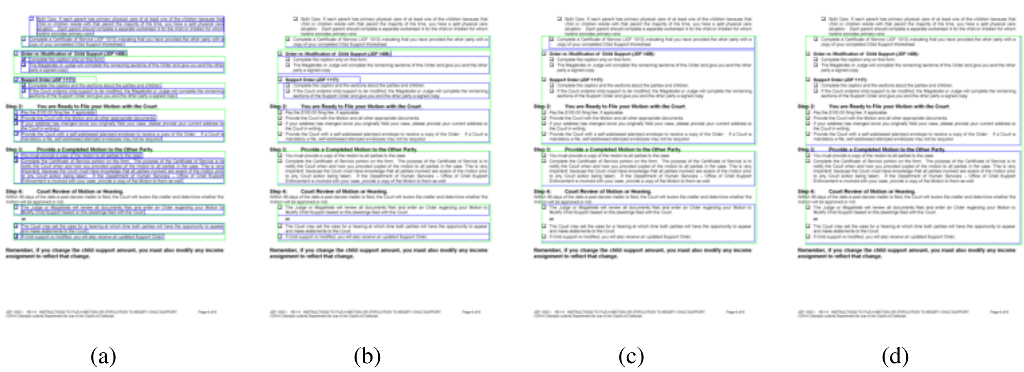
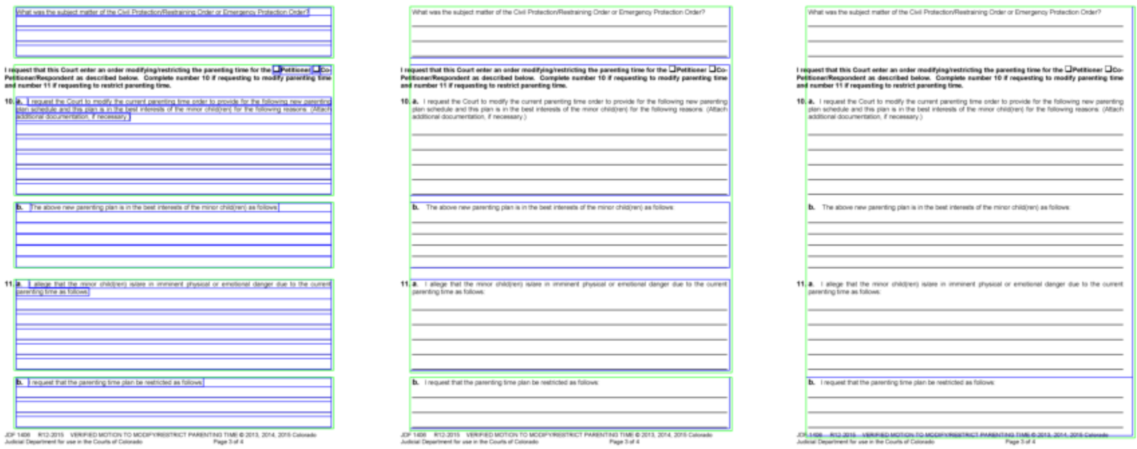


Figure 30: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

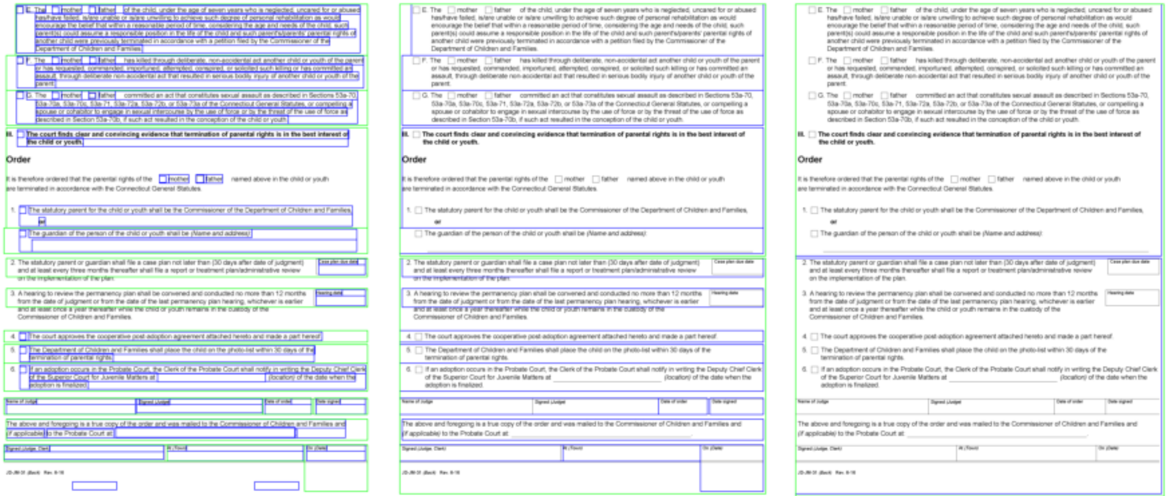


(a) (b) (c)

Figure 31: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

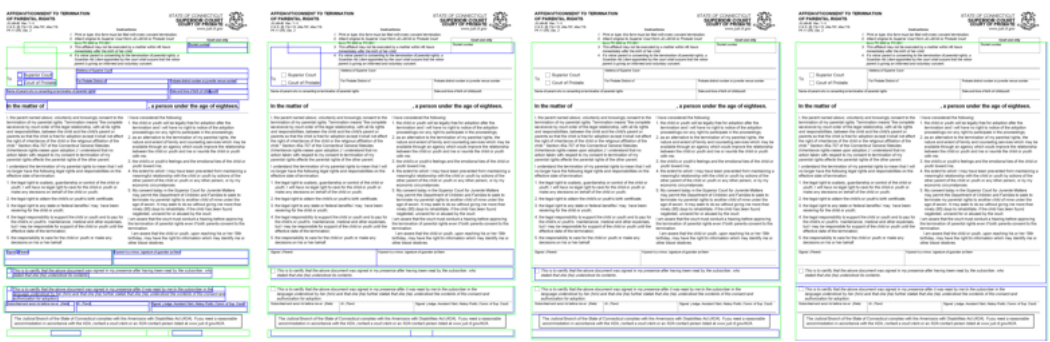
(a) (b) (c) (d)

Figure 32: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c)

Figure 33: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

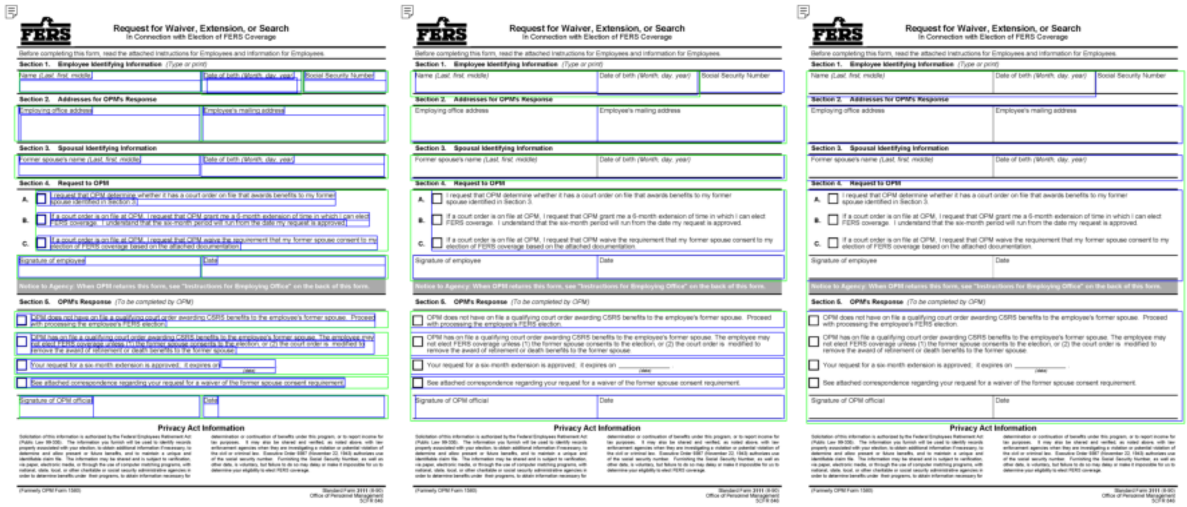


(a) (b) (c) (d)

Figure 34: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

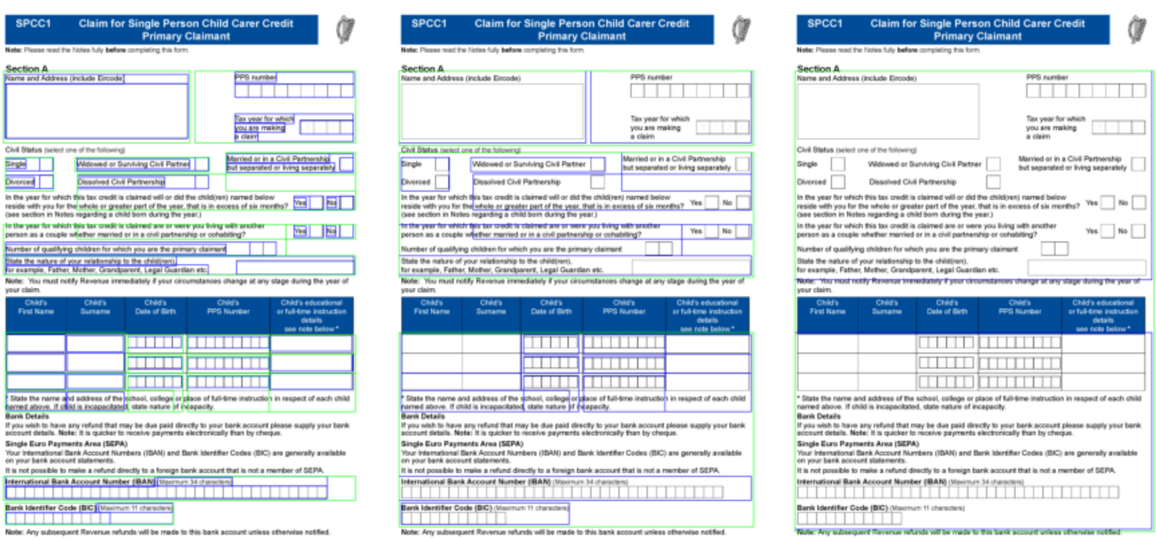
(a) (b) (c) (d)

Figure 35: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



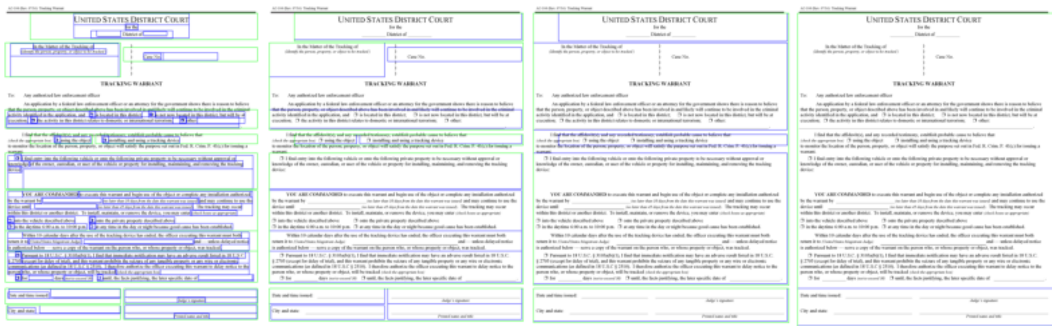
(a) (b) (c)

Figure 36: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



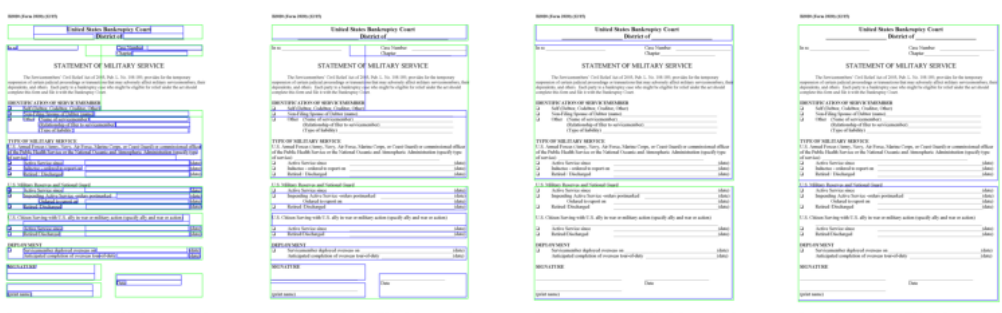
(a) (b) (c)

Figure 37: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c) (d)

Figure 38: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c) (d)

Figure 39: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

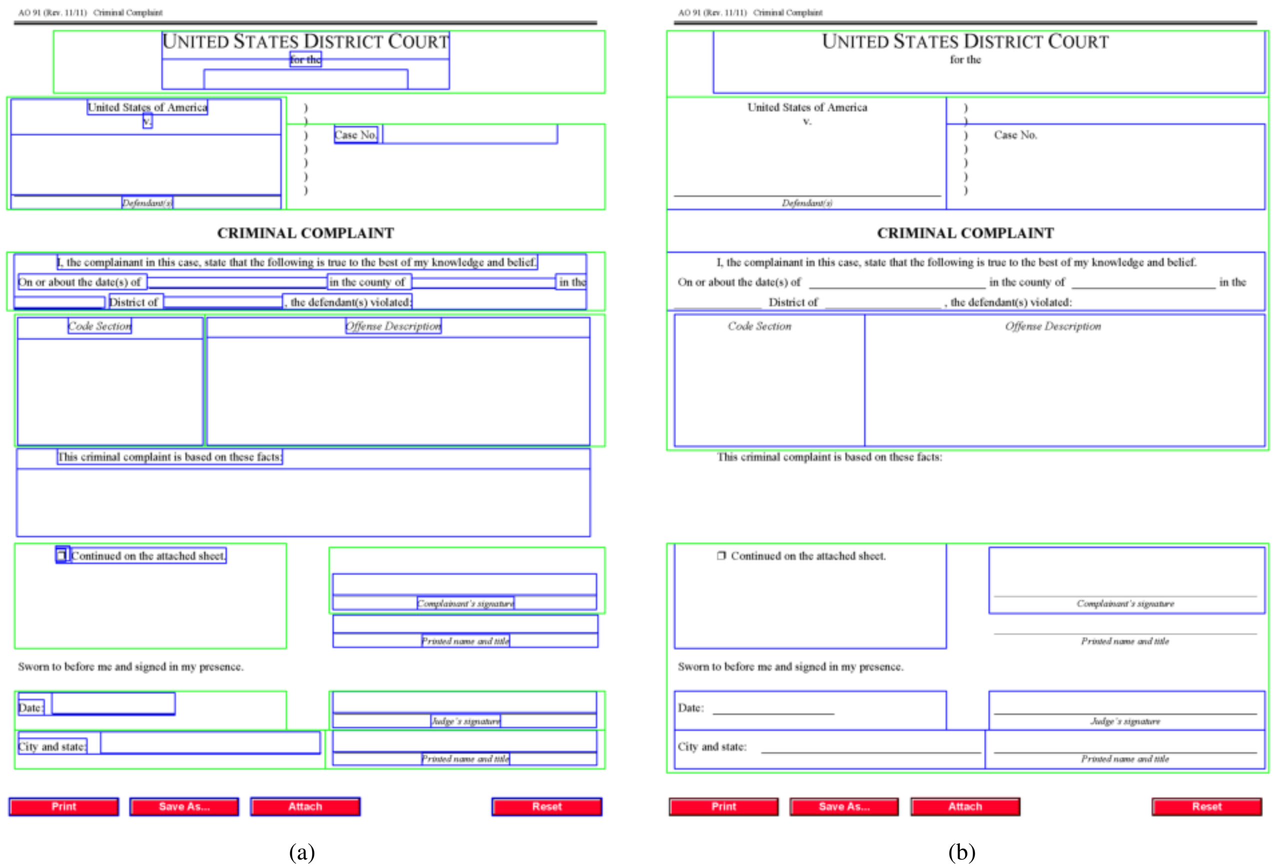


Figure 40: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.

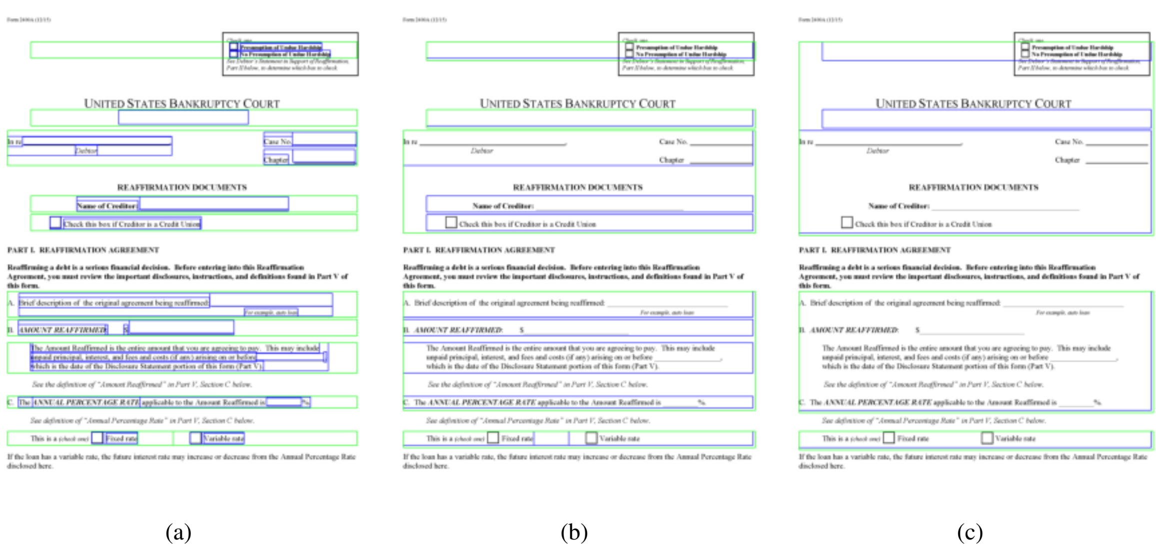
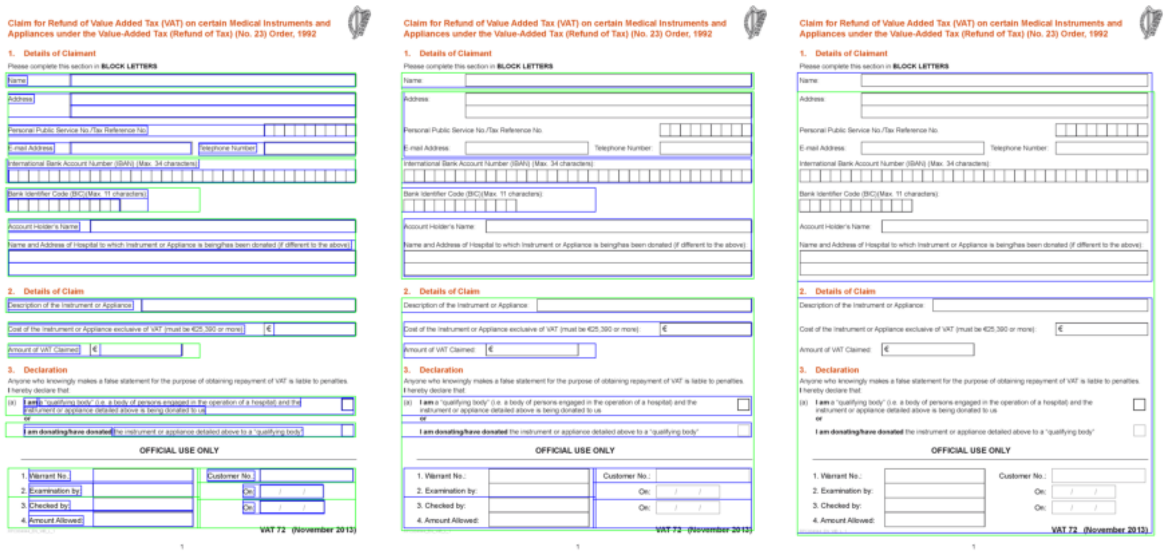
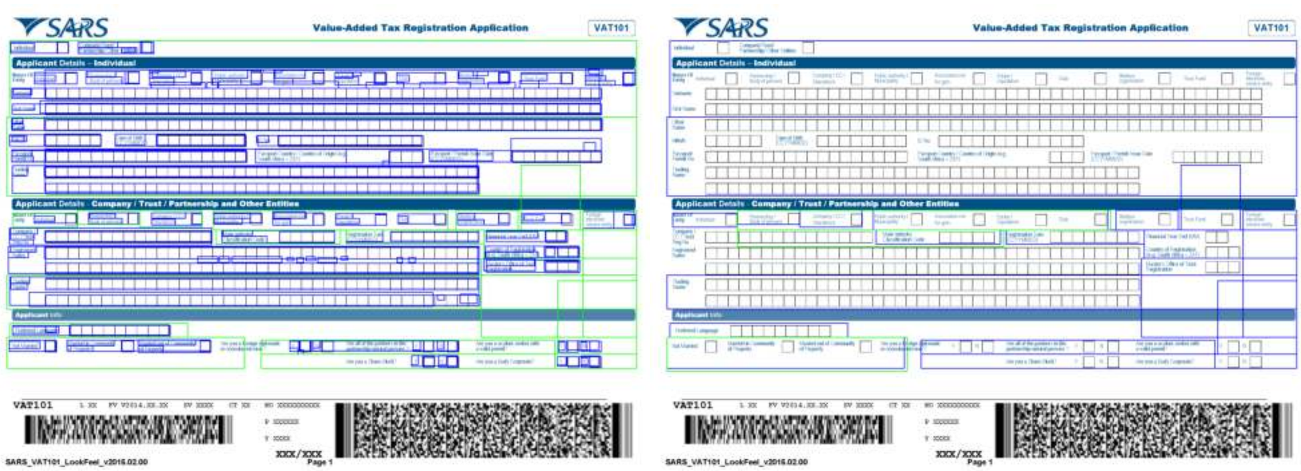


Figure 41: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b) (c)

Figure 42: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.



(a) (b)

Figure 43: Example illustrations of successive layer-wise predictions by LayerDoc model on the test set of the Hierarchical Forms dataset. Blue boxes denote the lowest layer of generated hierarchy while green boxes indicate the layer just above it.