Layout Generation as Intermediate Action Sequence Prediction -Supplement Material

Huiting Yang , Danqing Huang , Chin-Yew Lin , Shengfeng He*

In this supplement material, we first show our model's generalization capability on the unconditional generation and layout completion (Li et al. 2020). Next, we detail the network architectures for the FID ablation study, as well as the datasets and human evaluation in the main experiments.

Unconditional Layout Generation

The main paper has discussed the results on conditional layout generation where the input is a set of categorical labels. For unconditional layout generation, we only feed the first start token $< bos >$ as input, and our network will autoregressively generate the remaining sequence. Similar to the model analyses of conditional layout generation, here we show the results for the unconditional setting[1].

Model Analyse

Table 1 shows the statistics on the action sequence from both real data derivation and our model outputs. We obtain similar observations with the conditional setting: the rate of actions is in similar distribution with the real data. Moreover, the grammar correctness indicator shows higher accuracy than conditional generation, which is somehow reasonable since unconditional generation has been less intervened with respect to the input constraint.

Qualitative Results

In Figure 1, we show some unconditional generated cases, with two additional layouts with maximum IoU from the training set (2nd row) and our generated set (3rd row). We can see that our generated layouts are well-structured with proper alignments and margins. Meanwhile, high diversity property is preserved since the generated layouts are not identical to the most similar layouts in the training set (i.e. not memorizing the training data) and the generated set. Compared to mobile UI and scientific document, layouts in slide domain are more artistic with more elements on average. Our method can generate reasonable slide layouts with repetitive groups of elements, which indicates that the intermediate actions can help the model better generalize the structure features.

[1]Please note that most baselines (LayoutVAE (BBoxVAE) and LayoutGAN++) only target for the conditional input and thus we cannot show the quantitative comparison under this setting.

Layout Completion

In addition to the conditional layout generation with category label constraints, our model also supports applications such as layout completion (Li et al. 2020). Given a partial layout, our model learns to complete the layout by predicting the position and size of the remaining elements. Figure 2 shows some cases. Given the initial element in each example (1st column), we show three different completed layouts. This further demonstrates the flexibility and diversity of our model.

Details of the FID network

Architecture Details

Figure 3 illustrates the architectures of four FID feature extractor networks. Layout embeddings $\mathbf{h}$ (the purple blocks) in each of the four architecture diagrams encode the overall layout that are used to calculate the FID score in the evaluation. Transformer (Dec) is the same as the LayoutTransformer backbone except that we additionally add a layout embedding (the purple block) for evaluating the realness of the input layout. Transformer (Enc-Dec) is the discriminator used by the LayoutGAN++, which is also the FID feature extractor network used in Kotaro *et al.* (Kikuchi et al. 2021). We also follow the corresponding input format used in their paper. The bounding box input for Transformer (Dec) is represented with discrete variables while the input for Transformer (Enc-Dec) is represented with normalized continuous variables. The input for Layout Matching Network (GMN) is a graph where node features contain semantic and geometric information and edge features capture relative difference between nodes. Different from the above networks, the Resnet-18 takes the layout images as input rather than the bounding box.

Contrastive Learning Objective

Based on the observation that human rank Gaussian negatives consistently higher than shuffle negatives, we introduce the contrastive loss to constraint the learned distribution:

$$L(r, g, s) = max\{||r_i - g_i||_2 - ||r_i - s_i||_2 + margin, 0\}, \quad (1)$$

where r, g, s are the real sample features, Gaussian negative sample features and in-batch shuffle negative sample

Table 1: Statistics on the action sequence, including grammar correctness and trigger rate of different actions.

	Rico		PubLayNet		InfoPPT	
	Real data	ours	Real data	ours	Real data	ours
grammar correctness (%)	100	99.997	100	100	100	99.994
generate (%)	54.253	51.470	62.981	62.988	40.674	43.240
copy (%)	33.625	35.102	26.059	26.000	49.346	46.914
margin (%)	22.244	26.811	24.920	22.023	19.958	16.679

features, respectively. The margin value is set to 1 in our experiment.

Details of Human Evaluation

This section shows the layout ranking criteria and some cases presented to the human annotators. The annotators are first introduced with some basic design principles such as balance, alignment, repetition, and so on. Here are some detailed guidelines presented to the annotators:

1. Balance: the generated layout should have a balanced visual weight (i.e. visual elements should be distributed evenly in the layout). A large empty area is not desired.
2. Non-Overlap: elements should not be overlapped with each other, except for some categories such as background image and toolbar which might contain hierarchical child elements.
3. Alignment: elements should be properly aligned with the same x-/y- axis or same width/height when they are lined up in composition.
4. Overall Aesthetic: to consider the pleasing qualities of the overall layout.

Figure 4 $\sim$ Figure 6 show some cases presented in the human evaluation. Since the Shuffle Neg. layouts are generated by randomly interchanging elements of the in-batch layouts, the element category labels might seem different from other systems. Given the cases, annotators are asked to rank the samples in each group from the best to the worst. The final rankings are calculated by summing the score in each group per annotator. Figure 7 shows the summed ranking scores statistics, where we can see our model outperforms other systems by a large margin.

Details of Dataset

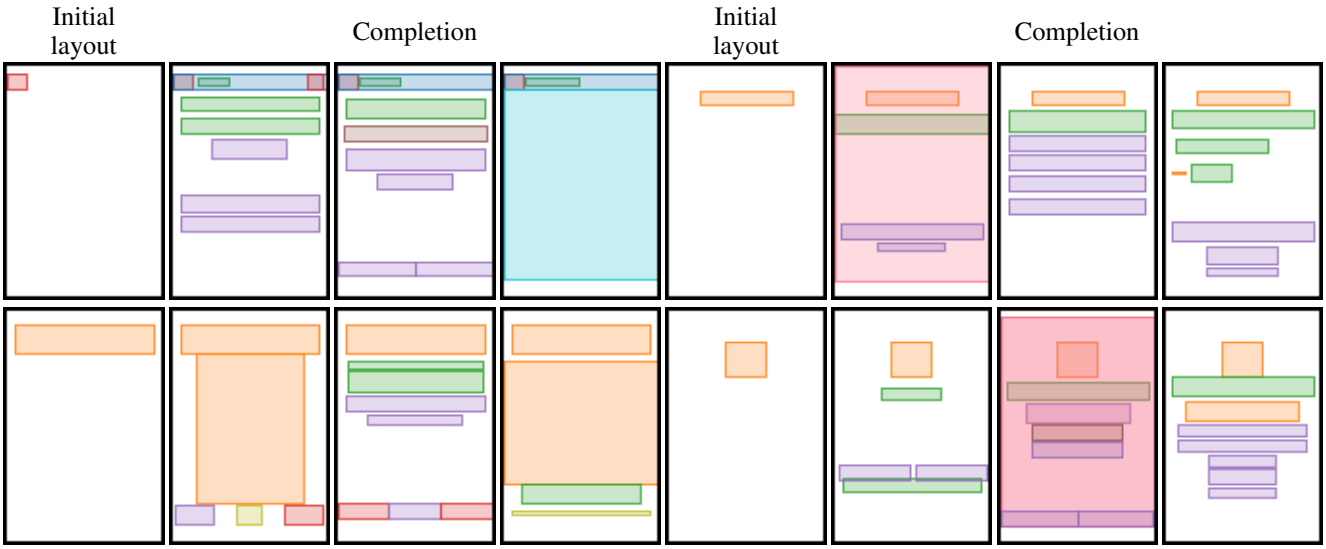
Figure 8 shows the category label sets and the corresponding number of labels for three datasets including Rico, PubLayNet, and InfoPPT.

References

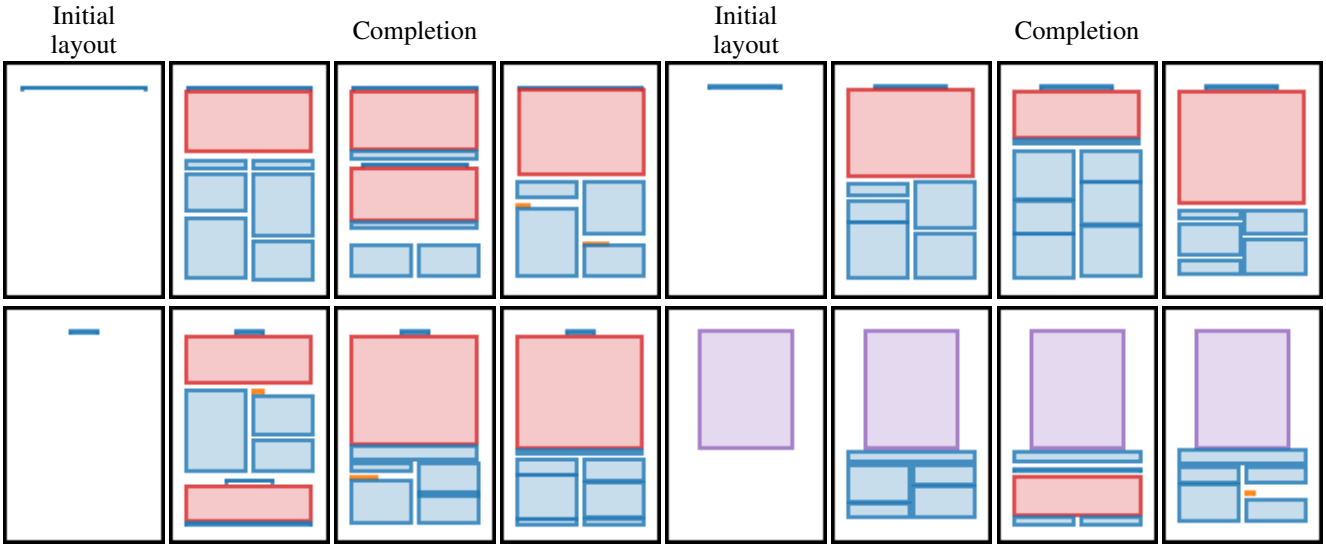
Kikuchi, K.; Simo-Serra, E.; Otani, M.; and Yamaguchi, K. 2021. Constrained graphic layout generation via latent optimization. In *Proceedings of the 29th ACM International Conference on Multimedia*, 88–96.

Li, Y.; Amelot, J.; Zhou, X.; Bengio, S.; and Si, S. 2020. Auto completion of user interface layout design using transformer-based tree decoders. *arXiv preprint arXiv:2001.05308*.

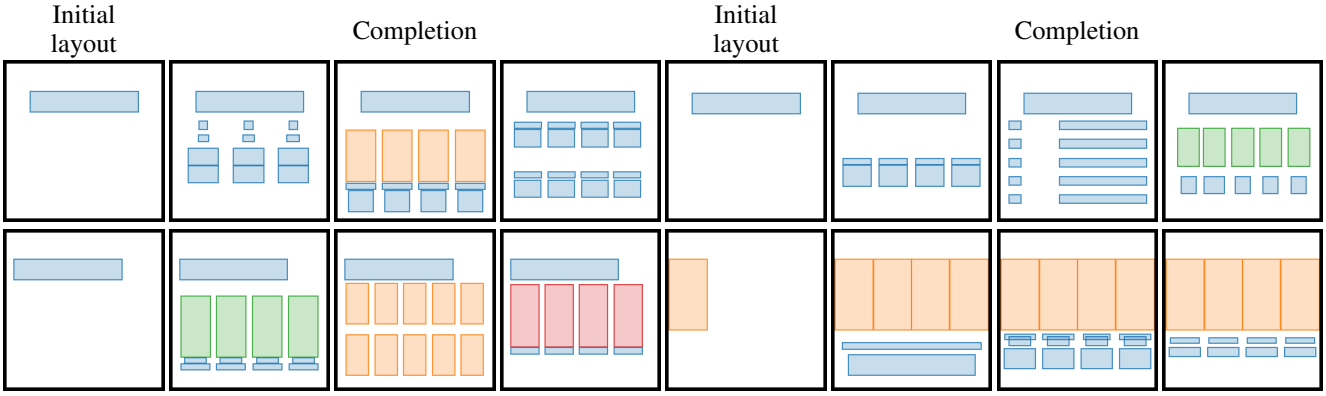
Figure 1: Unconditional layout generation samples for mobile UI, scientific article and slide. For each sample, we also retrieve two layouts from the training set (labeled as IoU (train)) and our generated set (labeled as IoU (generated)) with maximum IoU respectively.



(a) Rico

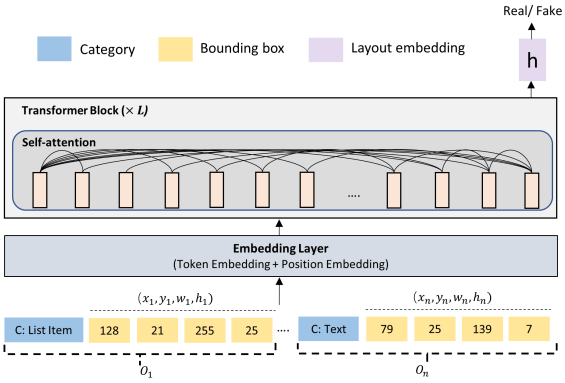


(b) PubLayNet

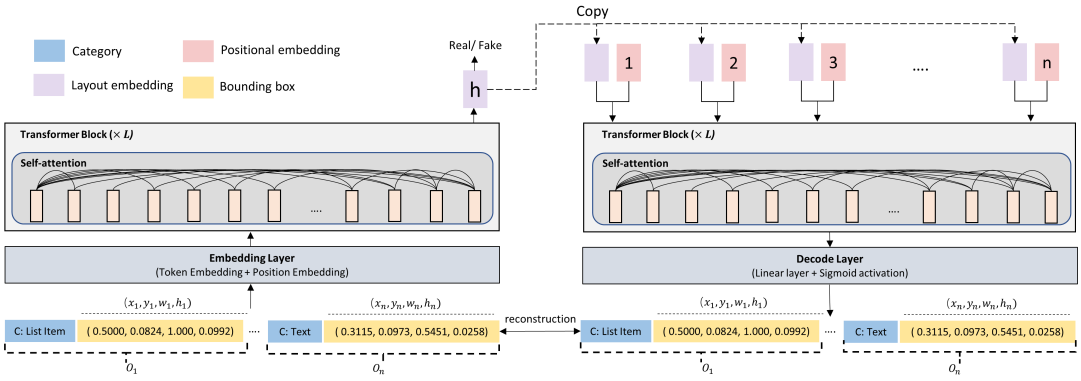


(c) InfoPPT

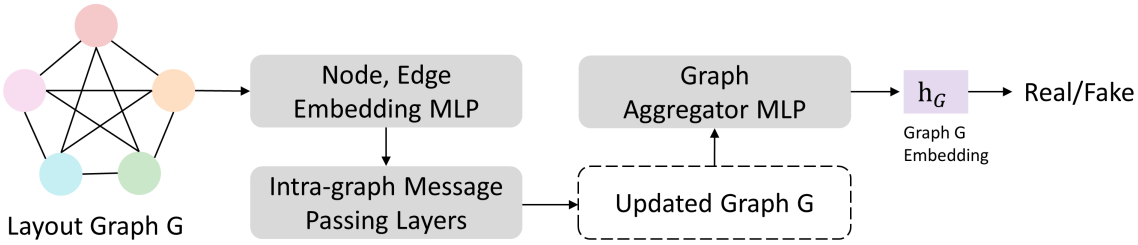
Figure 2: Multiple completions from same initial element.



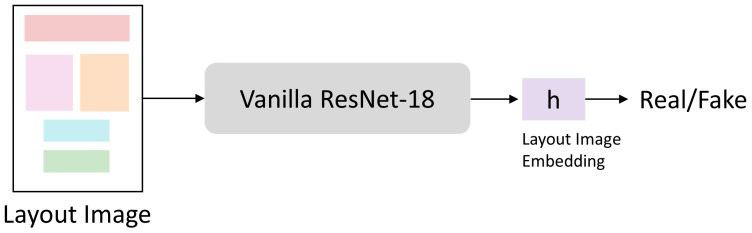
(a) Transformer(Dec)



(b) Transformer(Enc-Dec)



(c) GMN



(d) Resnet-18

Figure 3: Architecture details of four FID networks.

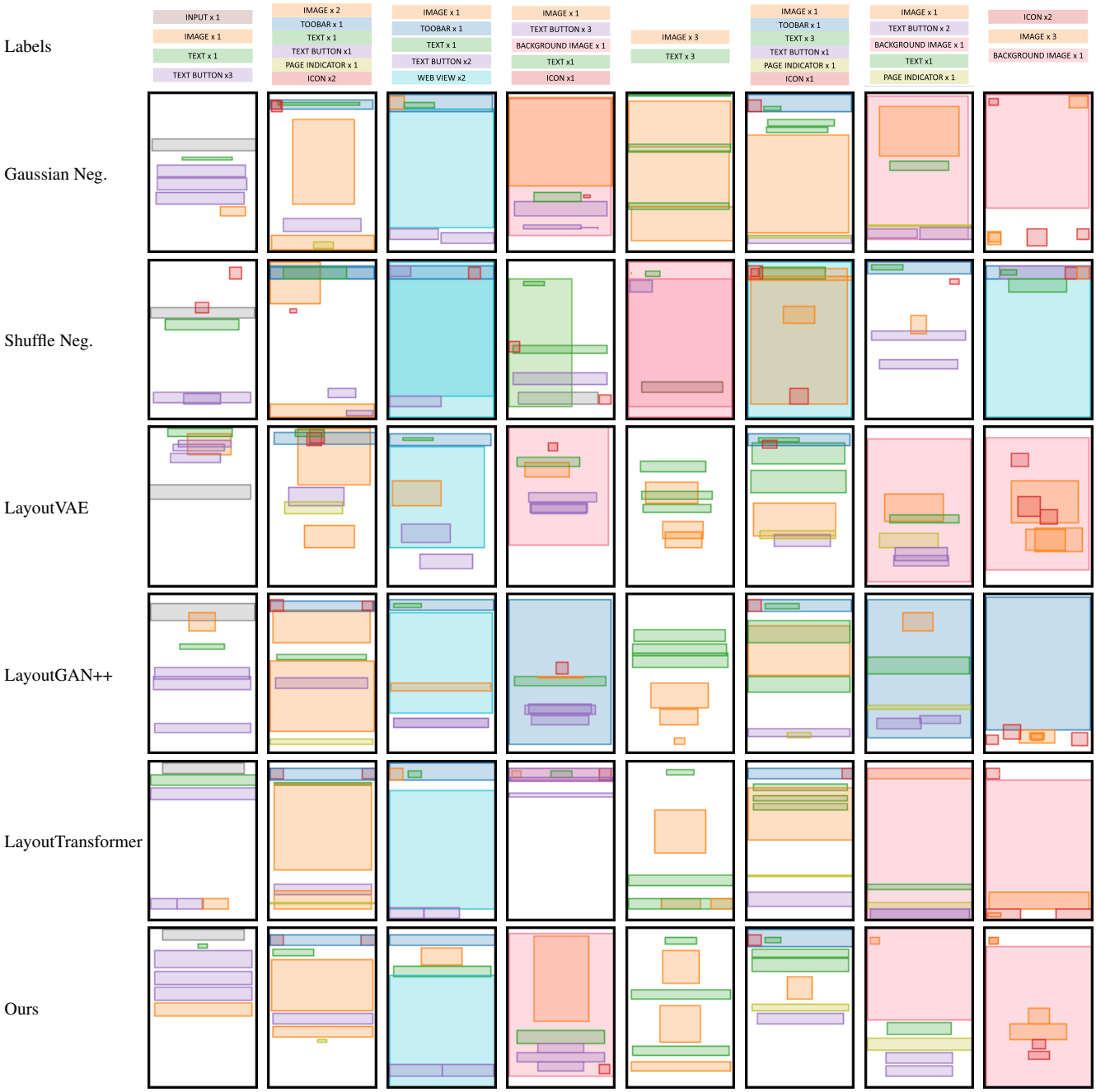


Figure 4: Some Rico cases presented in the human evaluation. Each group (per column) contains 6 systems for ranking.

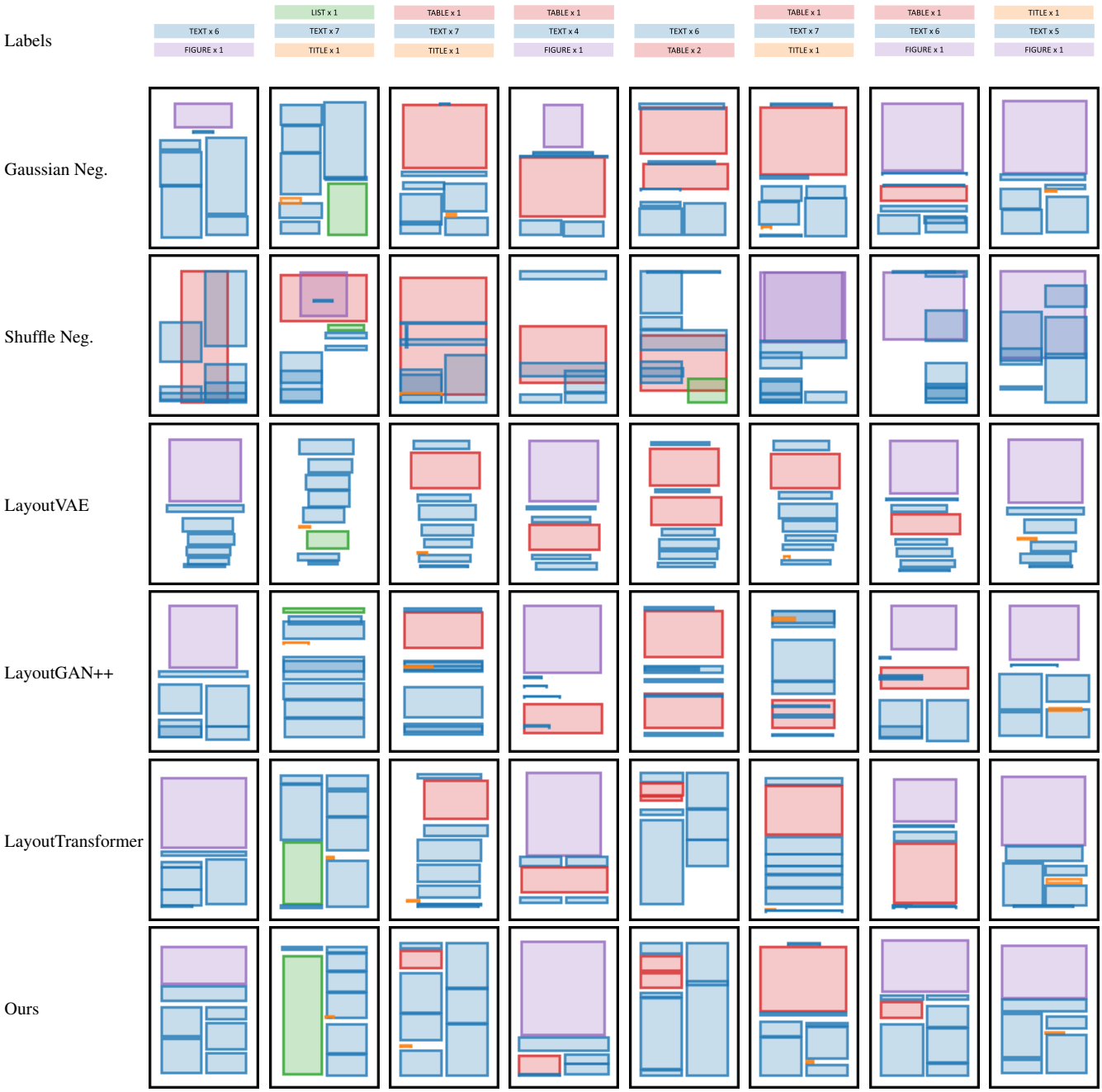


Figure 5: Some PubLayNet cases presented in the human evaluation. Each group (per column) contains 6 systems for ranking.

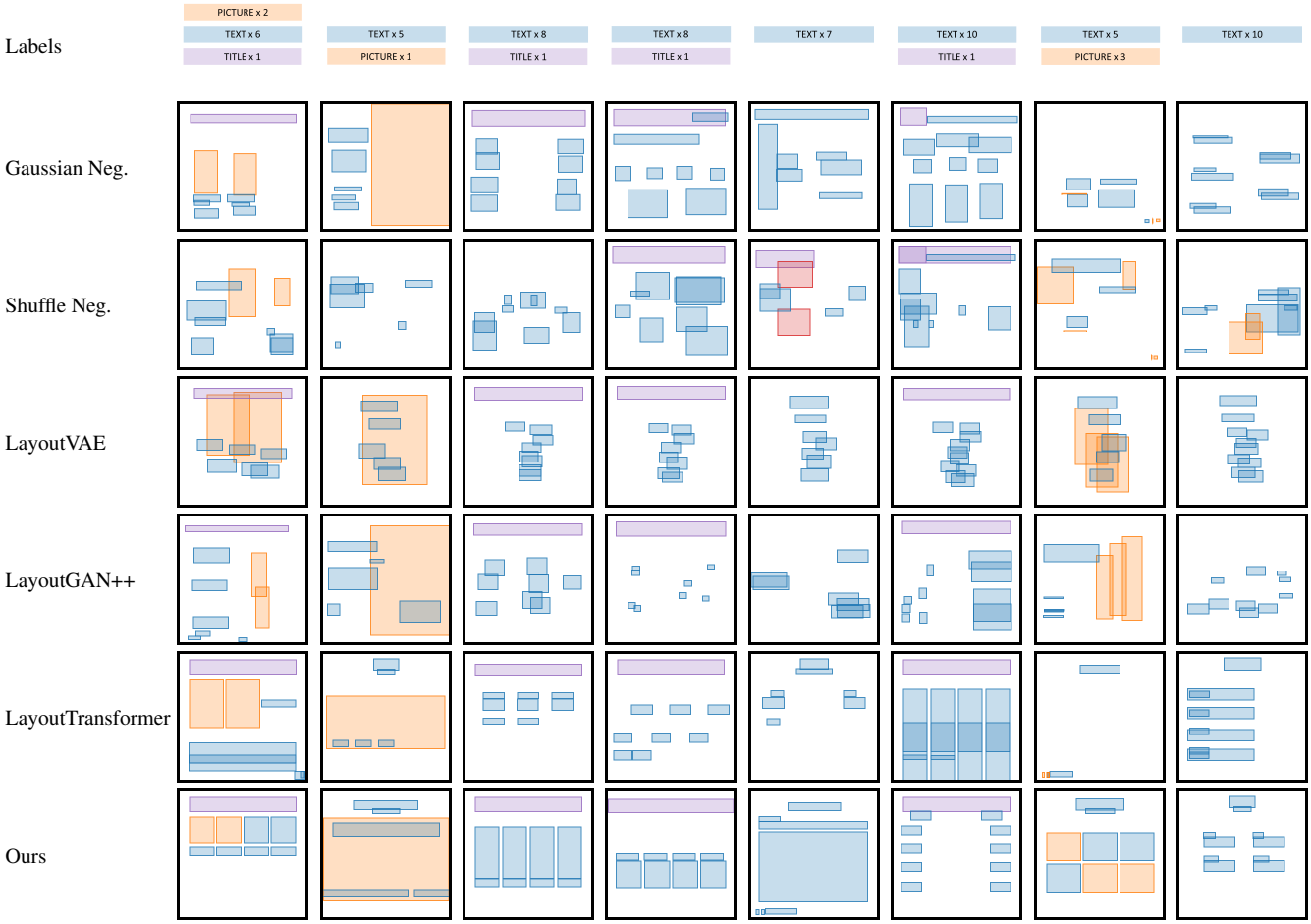


Figure 6: Some InfoPPT cases presented in the human evaluation. Each group (per column) contains 6 systems for ranking.

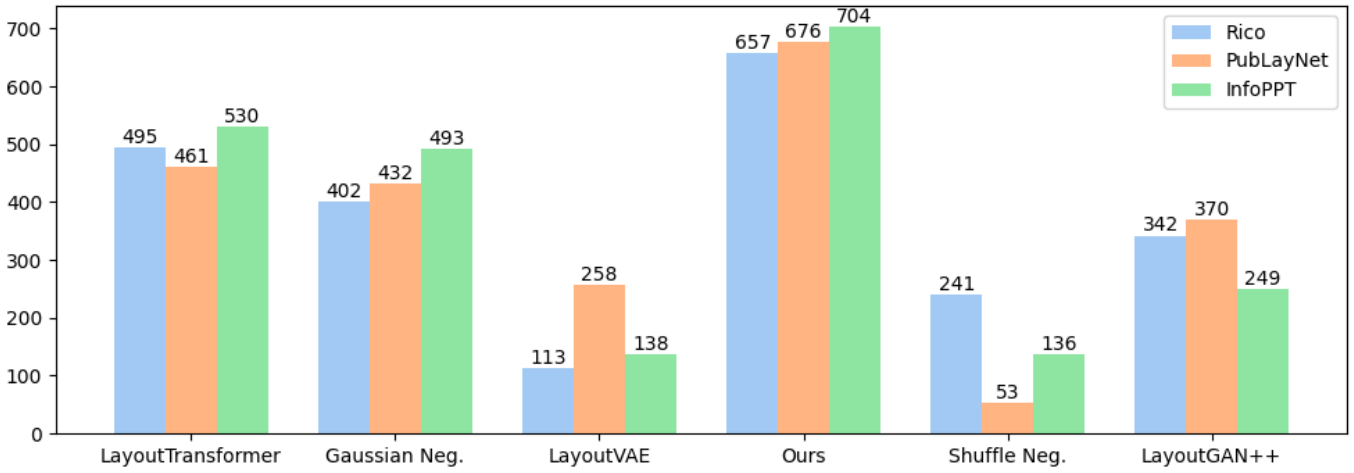
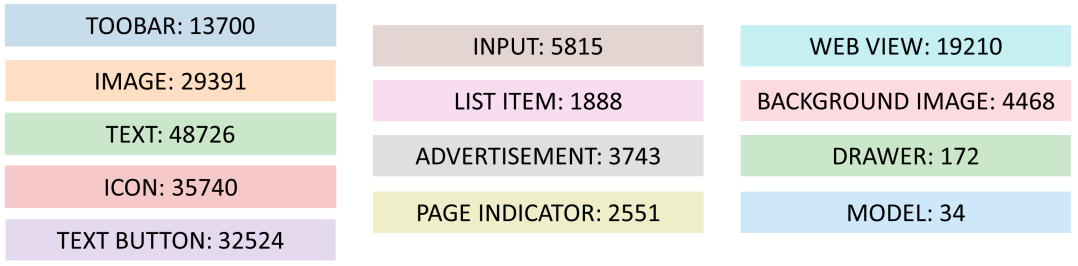
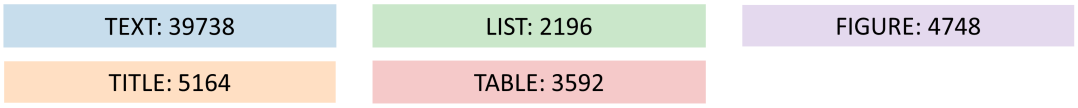


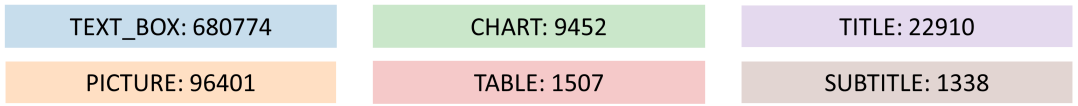
Figure 7: Ranking statistics of the human evaluation. The larger score indicates higher ranking.



(a) Rico



(b) PubLayNet



(c) InfoPPT

Figure 8: Category labels in three datasets, including mobile UI, scientific article and slide. Each block is annotated with "Label name: number" form.