



HAL
open science

Automatic Generation of Web Advertising Layouts: A Synthetic Dataset and a Deep Learning Baseline Model

Romain Carletto, Hubert Cardot, Nicolas Ragot

► **To cite this version:**

Romain Carletto, Hubert Cardot, Nicolas Ragot. Automatic Generation of Web Advertising Layouts: A Synthetic Dataset and a Deep Learning Baseline Model. 11th International Conference on Pattern Recognition Systems, Mar 2021, Curicó, Chile. hal-03385229

HAL Id: hal-03385229

<https://hal.science/hal-03385229>

Submitted on 19 Oct 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Automatic Generation of Web Advertising Layouts: A Synthetic Dataset and a Deep Learning Baseline Model

R. Carletto^{*1}, H. Cardot¹, N. Ragot¹

¹ LIFAT, Université de Tours, 37000 Tours, France
* romain.carletto@univ-tours.fr

Keywords: Open source dataset, Automatic document layout, Web Advertising, Deep neural networks, Residual blocks.

Abstract

Automatic generation of advertising layouts shows high economic interest, but as identified with our industrial partner, there is no public document layout dataset that matches this particular application. In this context, we produced two synthetic datasets that allow both the evaluation and training of any learning model on web advertising layout generation, and a small dataset of real cases to demonstrate the contribution of our work. We compared the results obtained by different learning models on the real cases, with and without prior use of our synthetic datasets, and our results show that these datasets allow to build and decisively improve models for the generation of real-world advertising layouts. Our three datasets, as well as useful data processing tools, are available at : https://github.com/romain-rsr/synth_datasets_for_web_advertising_layout/tree/master

1 Introduction

1.1 Automatic advertising layout applications

Laying out advertisements consists essentially of positioning and sizing rectangular bounding boxes containing the graphic elements. It is a long and still very manual job, which may seem simple due to the small number of input parameters, but it is not: the diversity and multiplicity of criteria combinations to be integrated, as well as their subjective nature require a certain level of abstraction, and small variations in a layout parameters can strongly impact the visual rendering.

1.2 Restricted access to real data

Due to advertisement ownership policies, bringing together thousands of advertising layouts within a dataset would require the agreement of too many separate entities. Furthermore, the same information (e.g. the position of an element) can be coded in thousands of different ways, due to the specificity of HTML and CSS languages, and it is currently impossible to retrieve the positions and dimensions of the elements by only parsing the code of the advertisement. The acquisition of a public dataset,

to develop, evaluate and compare models for online advertising layout is therefore a problem in its own right, to which our work seeks to respond.

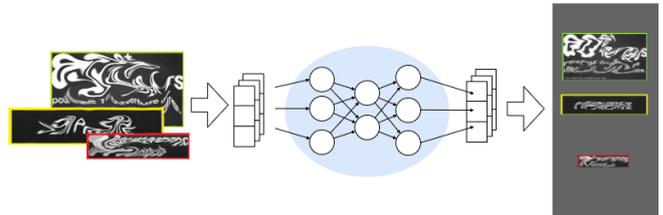


Figure 1. Real case application of an automatic advertising layout. Types and dimensions of input assets are processed by the model to generate dimensions and positions of these assets within a given rectangular area. Assets contents have been warped for privacy matters.

2 Related works

2.1 Former works

Early work on automatic layout problems sought to provide basic tools for designers. Some such as Myers [1] or Lok et al. [2] focus on the simple use of templates, others such as Feiner et al. [3], propose to the user to combine low-level rules. But these first tools are either too limited in terms of functionalities or too complicated to use.

More recent studies aim to bring the layout guidelines together in elaborate, multi-dimensional quantitative functions. Merrell et al. [4] incorporate these guidelines as terms in a density function, where the inputs are the positions and the dimensions of the elements to be arranged within a given space, and then apply a Markov Chain Monte Carlo sampler on this density function to generate layout suggestions. Other authors combine the layout guidelines into an objective function to be optimized. To this nonlinear optimization problem, Lin [5] adapt a linear two-pass method based on the simplex algorithm. He considers images and texts as inputs to its system, and output the distribution of these elements on a rectangular page, ideally seeking to leave no empty space, contrary to what is sought in online advertising.

Lisa Purvis et al. [6] encode various layout criteria, such as the degree of alignment, the balance of empty spaces, the legibility and compactness of the elements on the layout space, and the balance between images and text areas. They generate layouts that optimize the sum of these different encoded criteria by evolving a population of randomly initialized sets of layout parameters via a genetic algorithm. However, the authors put the quality of the examples thus generated into perspective and points out that very close scores at the output of their system can give very different quality renderings.

Therefore, the prioritization of terms within the objective functions defined by these various works remains problematic and these first methods remain on the whole too reductive to integrate the aesthetic canons of online advertising, which are highly complex and constantly evolving.

2.2 Learning methods

More recent work on automatic document layout has made use of deep learning models. These models have the advantage of being able to learn implicit rules of arrangement via very simple objective functions, and the depth of these models makes it possible to achieve abstraction capacities which are very well adapted to the complexity of document layout tasks.

Li et al [7] apply generative networks to different document layout problems. Their architecture is particularly elaborate and integrates several state-of-the-art concepts. They model the elements to be arranged in the form of wireframes, train and apply a Generative Adversarial Network (GAN) model to these inputs, and nevertheless add corrections at the GAN output to ensure, among other things, the alignment of the elements among themselves. Their results are significantly better than those obtained by conventional non-learning methods. However, deep learning models require a lot of training examples, which is not an obstacle in the applications covered by Li et al. article, but which is very problematic in the field of online advertising, where there is no dataset available.

2.3 Related datasets, unsuitable for online advertising

Datasets exist on applications that are partially related to advertising layout, including the Layout Analysis dataset, which presents articles containing mostly text, the RICO dataset, which contains mobile application pages, and the Abstract Scenes dataset, which presents layouts of a small number of different elements: characters, accessories, and background elements. But these datasets do not address the critical application issues of web advertising layout. Logos and call to action (CTA), for example, are two predominant categories of elements in the field of online advertising, and they do not present any similarity with the elements of these related datasets.

Furthermore, we find neither the spacing of the elements, nor the balance of spaces, nor the asymmetry that are systematically observed in online advertising. Finally, the reading order of the elements within an online advertisement is always sequential and circumvolutive and the combination of these two criteria is not found in any of these datasets. Therefore, we

built two synthetic datasets in order to be able to develop and evaluate truly incisive models for online advertising placement.

3 Datasets

3.1 First synthetic dataset : fictitious layouts, of arbitrary complexity

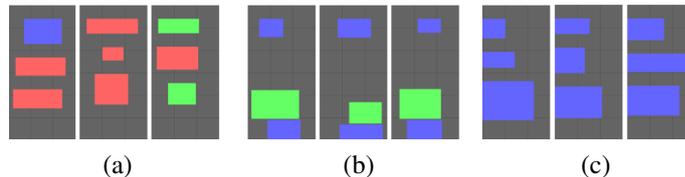


Figure 2. Layouts of our first dataset, created by fictitious layout rules. Layouts (a) have been created from general rules while layouts (b) and (c) have been generated through special rules, activated on layouts where the colors of the elements match specific combinations.

The first synthetic dataset contains 100,000 samples. For each sample, the input variables correspond to the information given to the designers at the start of any layout job : the categories (which in this first dataset are fictitious), the starting dimensions and the sequential order of the assets to be arranged. The corresponding output layout is then generated via a series of two sets of fictitious rules.

General rules, which apply to a large majority of samples, randomly size and position the elements within given constraints. Among these constraints, elements must not overlap, must not exceed the background frame and a minimum space is set between the elements.

Then, specific rules reposition and resize the elements of certain samples via specific and deterministic functions. An example of a specific rule is as follows: if the three elements are of the fictitious "blue" category, they are positioned against the left border. Yet, their dimensions are not affected by this rule and remain randomly defined within general rules constraints presented previously. These specific rules each relate to a particular order and combination of categories, to which a very small minority of the dataset samples correspond: about 3% of the entire dataset for each of the specific rules.

These artificial rules recreate and even increase tenfold the learning difficulties found in the real cases of advertising layout. The precision and diversity of our layout rules, the degree of contradiction between some of them, the fact that some rules concern only a very small minority of elements when others concern almost all the elements: each of these difficulties is inspired by a real problem in advertising layout and is arbitrarily increased within our first dataset.

Finally, while layout rules are only available in their implicit form in real document datasets, our artificial rules can both be presented in their implicit form during training (through the synthetic layouts they have created) and in their explicit form during evaluation, which makes it possible to evaluate any learning layout model with absolute precision.

3.2 Second synthetic dataset : realistic layouts, matching real advertising layout characteristics

The second synthetic dataset aims to allow a model to be pre-trained on it then used on real cases layouts. It contains 100,000 samples and was created through several realistic layout rules, identical to decisive rules in online advertising layout. As in the first dataset, the input elements to be laid out are rectangles of different types and sizes, except that here their categories are real categories used in advertising : "image", "text", "CTA" and "logo" The constraints of dimensions and positions of our explicit layout rules are here specific to these real categories, with for example the "image" elements being often bigger than CTA elements, position of the logo being imperatively in first or last position, etc. Output layouts parameters are then very similar to those of the real advertising layouts.

3.3 Real cases dataset for evaluation

To verify the contribution of these two synthetic datasets, we created a small dataset of a hundred real-life layouts, by manually gathering and segmenting advertisements arranged by our industrial partner. The quantitative and visual results obtained on the real cases presented in this article come from this dataset.

4 Models

Previous research on automatic document layouts jumped from methods without learning to unsupervised deep generative networks, that are particularly heavy to set up and train. Since there is no published evidence that supervised models can't achieve layout generation and since our synthetic datasets aim to enhance the performance of the broadest range of learning models, experiments have been run on our datasets with supervised networks, which are more common and whose training is also more stable.

We conceived and experimented several adaptations of the resnet architecture [8], presented in Figure 3. Model δ , which systematically reached better performances against the other deep residual models, is presented and referenced within our experiment results as our deep residual network. It is composed of 14 residual blocks followed by a fully connected layer of 12 neurons, activated by a sigmoid activation function. Each residual block contains one fully connected layer of 15 neurons, activated by a ReLU activation function.

A simple neural network of three fully connected layers (of respectively 12, 12 and 15 neurons) has also been experimented, aside of our deep residual model, to evaluate the added value of a deeper model on our datasets.

5 Experimental Setup

In the first dataset, samples where only general rules apply and samples concerned by a specific additional rule are shuffled all together. In each dataset, data has been split as usual into a train set, a validation set (to compare training and validation accuracy and assert the absence of over-fitting during training)

and a test set (for post-training performance evaluation). The results presented in this article have been recorded on the test set, which represents 20% of the samples in each dataset.

Since our goal is to integrate layout rules without giving their explicit form to the model during training, a very general learning metric has been used in all experimented models :

$$L(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{N} \sum_{n=0}^N (y_i - \hat{y}_i)^2 . \quad (1)$$

$L(\mathbf{y}, \hat{\mathbf{y}})$ represents the mean squared error between the vectors \mathbf{y} and $\hat{\mathbf{y}}$, that respectively contain the original and predicted parameters of each layout, and where N is the number of layouts to predict. Optimization is run through the Adam algorithm, which adapts the learning rate to each parameter and to the first and second moments of recent gradient magnitudes.

While the only purpose of our fictitious synthetic dataset is to evaluate a model on artificially challenging layouts, our realistic synthetic dataset aims to pre-train any learning model to generate real advertising layouts. Therefore, two training protocols were experimented and compared to assert such contribution. The first protocol is to train and evaluate a model on our small dataset of real cases only. The second protocol consists of training a model on our second synthetic data set, then re-training this model on the real cases train set and finally evaluating the model on the real cases test set.

6 Evaluation and results

6.1 Evaluation metrics

While in real cases layout, training and evaluation have to deal with implicit layout rules, the artificial rules that created our synthetic layouts can be shown implicitly to an experimented model during training (through the synthetic layouts themselves) then used in their explicit form during evaluation of the model results.

Once a model has been trained on a synthetic dataset, the layouts generated by this model must thus comply with a set of explicit layout rules, depending on the category of each element and the order of these elements, vertically and horizontally. These rules are defined exhaustively in our git directory, indicated in the abstract. In the real cases dataset, the layouts have been designed by hand by designers who follow intuitive, implicit layout rules, so our evaluation is limited to critical criteria : assets must not overlap and must not exceed screen borders. For each dataset, the respect of the dataset criteria by the generated layouts is evaluated through two different metrics.

A first metric is the Samples Average Maximum Error. For each sample, we measure the difference between the expected value of each feature, defined by the set of rules impacting this sample, and the value of this feature within the layout generated by the model. We then consider the maximum of these differences for each layout and compute the average of these maximums to get the Samples Average Maximum Error. This value is set in pixels, and has to be put into perspective with the banner frame dimensions, which is 300x600 pixels.

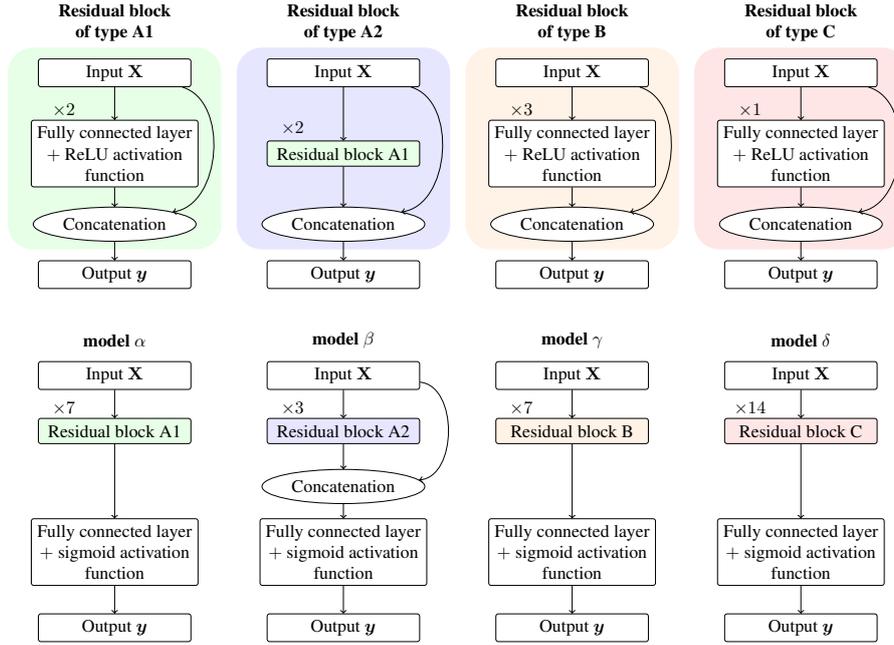


Figure 3. Experimented deep residual models. Best performance is reached by model δ , referenced in this article as our deep residual model.

A second metric is the percentage of Samples Containing Critical Errors, where, as specified, critical errors are those identified as strongly unacceptable in the industry : when one of the layout elements overlap on another or when it exceeds a screen border. Concerning layouts generated after a training on the first synthetic dataset, only layouts where general rules apply are evaluated by this metric. The details and code of our evaluation methods are also available in our git directory.

6.2 Results on our synthetic datasets

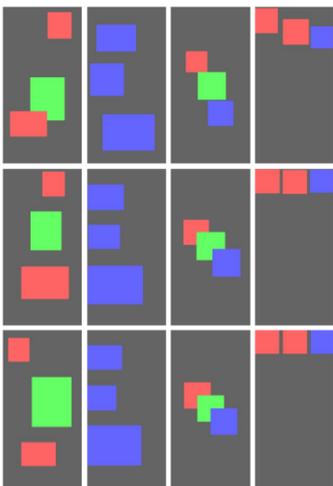


Figure 4. layouts generated by a three-layers fully connected network (first row) and by our deep residual model (middle row), after training on our synthetic dataset. Original synthetic layouts are presented in the lower row.

Evaluation Metric	Three Layers Fully Connected Network	Our Deep Residual Network
Samples Average Maximum Error	17	3
Samples Containing Critical Errors	13%	0.7%

Table 1. Models performances on our fictitious synthetic dataset

Evaluation Metric	Three Layers Fully Connected Network	Our Deep Residual Network
Samples Average Maximum Error	9	2
Samples Containing Critical Errors	8%	0%

Table 2. Models performances on our realistic synthetic dataset

In Figure 4, generated layouts are compared to original synthetic layouts. As indicated in the dataset section, some parameters have been set randomly in each layout, so while some layout parameters are expected to be the same between a generated layout and an original layout, some other parameters are not concerned by this criteria and can be different without impacting the validity of a generated layout, at the evaluation step.

On the left column layout, due to the sequence of asset categories (which here is red-green-red), only general rules apply : a large range of positions and dimensions are valid but no over-

lap is tolerated. On other columns, on the contrary, additional virtual rules apply, which are specific to the related sequence of asset categories and can even override some of the general rules. Therefore on each of these other columns, exact values are expected for some particular parameters, e.g. lefts must all be equal to zero on the second column, precise overlap patterns are expected on the third column (which is deliberately contradictory with the general no-overlap criteria), tops must all be equal to zero on the fourth column.

While the three-layers network results show important issues (prohibited overlaps in the first column and unexpected values in other columns), the visual accuracy of our deep residual model is in line with industrial requirements, in terms of distances between generated parameters and closest valid asset parameters. These visual results are confirmed quantitatively in Table 1, where concerning our deep residual model, the Samples Average Maximum Error is only a few pixels long and the percentage of Samples in Critical Error is very close to zero. Same performance shifts are observed on our realistic synthetic dataset, as quantitatively displayed in Table 2

6.3 Results on real cases

Evaluation Metric	Training on Real Cases	Training on Synthetic Samples Then Training on Real Cases
Samples Average Maximum Error	25	11
Samples Containing Critical Errors	60%	4%

Table 3. Performance gains of our deep residual model on real cases due to pre-training on our realistic synthetic dataset

In Figure 5, we compare real advertising layouts generated by our deep residual network after different training protocols. For now our experiment focus on the layout of assets on a plain background. More sophisticated backgrounds and their graphical content are not considered yet, therefore they are replaced by a plain background in the real layouts generated by our models. Also, assets graphical content have been warped for privacy matters but their categories are indicated by a color frame around each asset : logos, cta, texts and images respectively correspond to red, yellow, green and blue frame colors.

The first column shows poor results, generated by our deep residual model after training on our small real cases dataset only. The second column shows much better results, reached by the same model being pre-trained on our synthetic realistic dataset, then trained again on on our small real cases dataset. These results are confirmed by quantitative results displayed in Table 4.

In Figure 6, we visually compare the results obtained by a simple three-layer model with those achieved by our deep residual model. These visual differences, in addition to the quantitative performance shift seen in Table 4, show that trivial models are not suitable for real advertising layout tasks, while

Evaluation Metric	Three Layers Fully Connected Network	Our Deep Residual Network
Samples Average Maximum Error	38	11
Samples Containing Critical Errors	16%	4%

Table 4. Models performances on real cases. Each model has been pre-trained on our realistic synthetic dataset

our own deep residual model matches reliability expectations on critical layout criteria.

Finally, we see through both visual and quantitative evaluations that the performances shifts (between two different models) are positively correlated between our synthetic and our real-cases datasets, showing that our synthetic datasets match similar problematic with the real cases dataset.

7 Conclusion

7.1 Identified contributions

We created two synthetic datasets that allow for strong quantitative evaluation in document layout generation, which is sorely lacking in current state of the art, where only user studies are available. These datasets also allow for the training of deep learning models on advertising layout generation, in a context where real web advertising layouts are highly restricted and where related document layout datasets do not match decisive characteristics of advertising layout.

Our fictitious synthetic dataset allows to evaluate the capacity of a given model to integrate layout rules that match and even surpasses specific challenges of advertising layouts generation. Our results also show that the complexity of these virtual layout rules is not accessible to trivial neural network models, and can be addressed only by deeper models. Our realistic synthetic dataset makes it possible to pre-train a model to generate real advertising layouts that match industry critical criteria, even when only a few real training samples are available.

Finally, the tools we built to create our synthetic datasets can easily be handled by designers to teach learning models any creative bias, in the form of new explicit rules of their choice, in addition to implicit rules that those models directly learn on real data. This combines the advantage of both learning and non-learning approaches that were opposed so far, as described in our related works section.

7.2 Future works

Since our work aimed to enhance the evaluation and the training of main deep learning models, we focused our experimentation on supervised models which are widespread, stable to train and allow, as our results show, to generate real advertising layouts that match industry decisive requirements. However, generative unsupervised models are often experimented in related works on layout generation and they are never compared



Figure 5. Visual comparison between real advertising layouts generated with and without pre-training (resp. middle and left columns) on our second synthetic dataset. Ground-truth is presented in the right column. Several results in the first column show critical overlapping issues while results of the middle column match all decisive criteria.

with supervised models. Therefore, it could be very interesting to compare unsupervised generative models with our supervised model. This comparison can be run on our synthetic datasets but also on related datasets, which do not match advertising layout requirements but address other valuable layout problems.

Another interesting approach would be to replace dimensional inputs by actual images content and analyze the graphical and semantic content of input elements in the process of generating layouts. In addition, adapting saliency maps and segmentation networks to integrate the background graphical content and then position elements between important areas of the background could be of great interest.

Acknowledgements

This work is financed by Centre Val de Loire Region (France) and by Madmix Digital, a creative studio based in Paris and New-York, which has enabled us to identify and scientifically match the major challenges of layout generation in the online advertising industry.

References

[1] B. A. Myers, "User interface software tools," *ACM Transactions on Computer-Human Interaction*, 1995.



Figure 6. Visual comparison between layouts generated by a three-layers fully connected network (left column) which show critical sizing and overlapping issues, and layouts generated by our best performing model (middle column) which show satisfying results. Ground-truth is presented in the right column.

[2] S. Lok, S. Feiner, and G. Ngai, "Evaluation of visual balance for automated layout," in *Proceedings of the 9th international conference on Intelligent user interfaces*, 2004.

[3] S. Feiner, S. Nagy, and A. Van Dam, "An Experimental System for Creating and Presenting Interactive Graphical Documents," *ACM Transactions on Graphics (TOG)*, 1982.

[4] P. Merrell, E. Schkufza, Z. Li, V. Koltun, and M. Agrawala, "Interactive Furniture Layout Using Interior Design Guidelines," *ACM Trans. Graph*, 2011.

[5] X. Lin, "Active Layout Engine: Algorithms and Applications in Variable Data Printing," *Computer-Aided Design, Volume 38, Issue 5*, 2005.

[6] L. Purvis, S. Harrington, B. O'Sullivan, and E. C. Freuder, "Creating Personalized Documents: An Optimization Approach," *ACM symposium on Document engineering*, 2003.

[7] J. Li, J. Yang, A. Hertzmann, J. Zhang, and T. Xu, "Layoutgan: generating graphic layouts with wiredframe discriminators," in *ICLR*, (New Orleans (USA)), 2019.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, (Las Vegas (USA)), 2016.