Object Detection Pipeline Using YOLOv8 for Document Information Extraction

Notebook for the DocILE Lab at CLEF 2023

Jakub Straka[1], Ivan Gruber[1]

[1]*Department of Cybernetics and New Technologies for the Information Society, Technická 8, 301 00 Plzeň, Czech Republic*

Abstract

The extraction of information from semi-structured documents is an ongoing problem. This task is often approached from the perspective of NLP and large transformer-based models are employed. In our work, we successfully demonstrated that the Key Information Localization and Extraction (KILE) and Line Item Recognition (LIR) tasks can be effectively addressed as object detection problems using a convolutional neural network (CNN) model. We utilized a relatively small and fast YOLOv8 model for which we conducted a series of experiments to explore the impact of different factors on model training. With YOLOv8, we were able to achieve AP 0.716 on the KILE task and 0.638 on the LIR task. Our code is available at https://github.com/strakaj/YOLOv8-for-document-understanding.git.

Keywords

Document Understanding, Document Information Extraction, Object Detection, YOLOv8, Line Item Recognition, Key Information Localization and Extraction

1. Introduction

Companies engage in daily communication often through semi-structured documents. Upon receiving such documents, the initial step involves manually extracting the data before it can be processed. This manual process is repetitive and time-consuming, leading to the question of whether the information from these documents could be obtained automatically.

However, solving this problem automatically proves to be a complex task for several reasons. Although invoices may contain similar information, their layouts and appearance can vary dramatically across different companies. Another challenge stems from the need to understand the text and the relationship between the individual parts of the document. In some cases, the context, necessary to understand the document, may be missing, for example, when a multi-page table lacks the header on subsequent pages, it may be difficult to interpret values on those pages.

Document understanding includes various tasks, including Line Item Recognition (LIR) and Key Information Localization and Extraction (KILE) as defined in [1]. LIR specifically focuses on processing information from tables. While a typical table assumes that each row corresponds to a single item, this assumption is often oversimplified. In reality, business documents frequently

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

✉ strakajk@kky.zcu.cz (J. Straka); grubiv@ntis.zcu.cz (I. Gruber)

iD 0000-0002-9981-1326 (J. Straka); 0000-0003-2333-433X (I. Gruber)

contain tables where items can span multiple lines. The objective of LIR is to detect and classify all the information in the table and group related information into line items. KILE is a relatively simpler task that only aims to detect information and classify them into a predefined set of classes.

2. Data

The DocILE dataset [1, 2] is composed of three subsets. The main subset is a set of around 6700 annotated documents. All documents were chosen to have an invoice-like structure. The second subset is a set of around 100 000 synthetically generated documents and the last is a set of around 900 000 unlabeled documents. In our work, we primarily focused on the annotated subset.

Annotated subset contains 5180 training documents, 500 validation documents, and 1000 test documents. Each document can have multiple pages resulting in 6759 training pages, 635 validation pages, and 1321 test pages. The KILE task contains 36 classes while the LIR task contains 19 classes. In both tasks, information that needs to be extracted is referred to as fields. These fields contain text and a bounding box that represents their location. Additionally, in the LIR task, they also contain the id of the line item to which they belong. The documents in the dataset are divided into layout clusters based on the types of fields in the document and their position. This means that all documents in one layout cluster have fields with the same types on the same positions.

The dataset exhibits a significant diversity of layouts, as demonstrated by the validation set consisting of 204 distinct layout clusters and the training set containing 1063 clusters. This is one of the main challenges of the document understanding task. In Figure 1 are shown two documents from two layout clusters with similar classes but very different layouts.

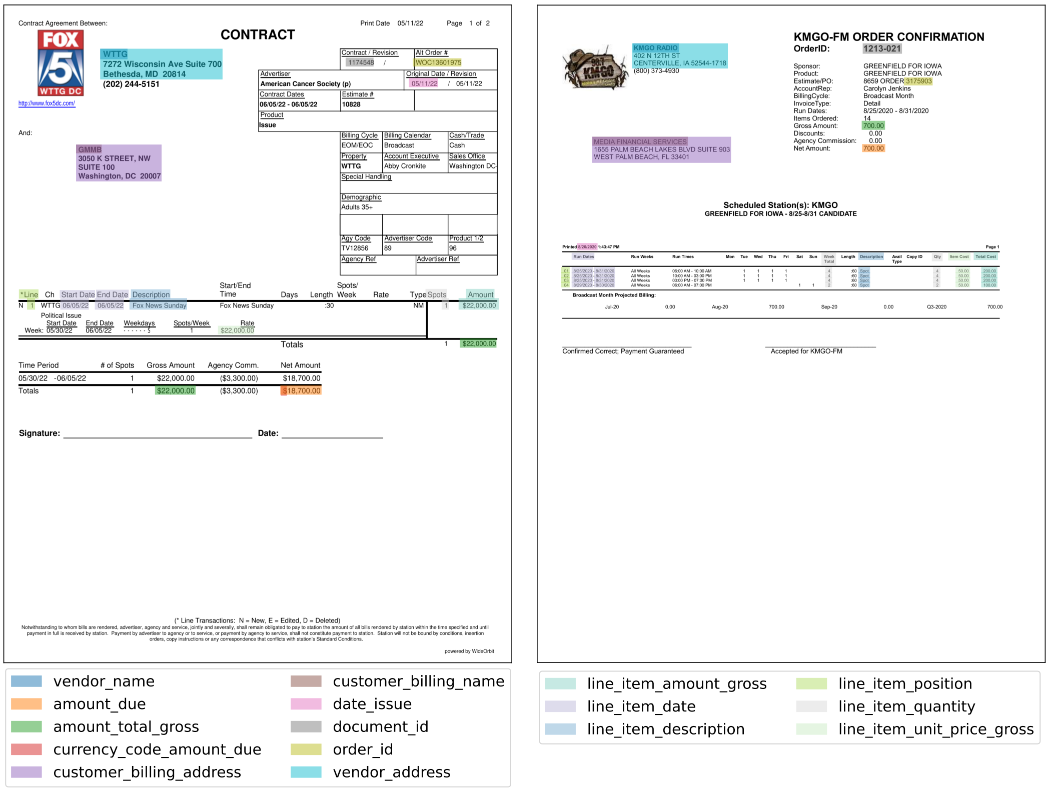
2.1. Evaluation metric

For both tasks, it is only necessary to detect the position of all fields, their class (field type), and in the case of the LIR task, to assign the field to the correct line item. Since the task is specified as a detection task, the authors of the competition chose a standard metric for object detection. The primary metric used for the KILE task is average precision (AP), and for the LIR task is used micro F1 score (F1). To make the metric more suitable for the tasks, Pseudo-Character Centers (PCC) of letters are used instead of Intersection-over-Union to determine true positives. This choice was inspired by [3]. PCCs are generated by splitting the field uniformly by the number of its character. The detected field should contain only PCCs corresponding to that field.

3. Model

In this work, we approached both tasks as object detection tasks. Because of its speed and small size, we decided to utilize the YOLOv8[1] [4] model. YOLOv8 is a one-stage, anchor-free detector based on a convolutional neural network (CNN).

[1] Implementation: https://github.com/ultralytics/ultralytics



Contract Agreement Between:

CONTRACT

Print Date 05/11/22 Page 1 of 2

WTTG
7272 Wisconsin Ave Suite 700
Bethesda, MD 20814
(202) 244-5151

http://www.fox5dc.com/

And:

GMMB
3050 K STREET, NW
SUITE 100
Washington, DC 20007

Contract / Revision 1174548 /	Alt Order # WOC13601975
Advertiser **American Cancer Society (p)**	Original Date / Revision 05/11/22 / 05/11/22
Contract Dates **06/05/22 - 06/05/22**	Estimate # **10828**
Product **Issue**	

Billing Cycle EOM/EOC	Billing Calendar Broadcast	Cash/Trade Cash
Property **WTTG**	Account Executive Abby Cronkite	Sales Office Washington DC
Special Handling		
Demographic Adults 35+		
Agy Code TV12856	Advertiser Code 89	Product 1/2 96
Agency Ref	Advertiser Ref	

	*Line	Ch	Start Date	End Date	Description	Start/End Time	Days	Length	Spots/ Week	Rate	Type	Spots	Amount
N	1	WTTG	06/05/22	06/05/22	Fox News Sunday	Fox News Sunday		:30			NM	1	$22,000.00

Political Issue

	Start Date	End Date	Weekdays	Spots/Week	Rate
Week:	05/30/22	06/05/22	- - - - - - S	1	$22,000.00

Totals 1 $22,000.00

Time Period	# of Spots	Gross Amount	Agency Comm.	Net Amount
05/30/22 -06/05/22	1	$22,000.00	($3,300.00)	$18,700.00
Totals	1	$22,000.00	($3,300.00)	$18,700.00

Signature: ____________________ **Date:** ____________

(* Line Transactions: N = New, E = Edited, D = Deleted)
Notwithstanding to whom bills are rendered, advertiser, agency and service, jointly and severally, shall remain obligated to pay to station the amount of all bills rendered by station within the time specified and until payment in full is received by station. Payment by advertiser to agency or to service, or payment by agency to service, shall not constitute payment to station. Station will not be bound by conditions, insertion orders, copy instructions or any correspondence that conflicts with station's Standard Conditions.

powered by WideOrbit

KMGO RADIO
402 N 12TH ST
CENTERVILLE, IA 52544-1718
(800) 373-4930

KMGO-FM ORDER CONFIRMATION
OrderID: 1213-021

Sponsor:	GREENFIELD FOR IOWA
Product:	GREENFIELD FOR IOWA
Estimate/PO:	8659 ORDER 3175903
AccountRep:	Carolyn Jenkins
BillingCycle:	Broadcast Month
InvoiceType:	Detail
Run Dates:	8/25/2020 - 8/31/2020
Items Ordered:	14
Gross Amount:	700.00
Discounts:	0.00
Agency Commission:	0.00
Net Amount:	700.00

MEDIA FINANCIAL SERVICES
1655 PALM BEACH LAKES BLVD SUITE 903
WEST PALM BEACH, FL 33401

Scheduled Station(s): KMGO
GREENFIELD FOR IOWA - 8/25-8/31 CANDIDATE

Printed 8/20/2020 1:43:47 PM Page 1

	Run Dates	Run Weeks	Run Times	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Week Total	Length	Description	Avail Type	Copy ID	Qty	Item Cost	Total Cost
01	8/25/2020 - 8/31/2020	All Weeks	06:00 AM - 10:00 AM		1	1	1	1			4	:60	Spot			4	50.00	200.00
02	8/25/2020 - 8/31/2020	All Weeks	10:00 AM - 03:00 PM		1	1	1	1			4	:60	Spot			4	50.00	200.00
03	8/25/2020 - 8/31/2020	All Weeks	03:00 PM - 07:00 PM		1	1	1	1			4	:60	Spot			4	50.00	200.00
04	8/29/2020 - 8/30/2020	All Weeks	06:00 AM - 07:00 PM						1	1	2	:60	Spot			2	50.00	100.00

Broadcast Month Projected Billing:

Jul-20	0.00	Aug-20	700.00	Sep-20	0.00	Q3-2020	700.00

Confirmed Correct; Payment Guaranteed

Accepted for KMGO-FM

Figure 1: Example of two documents with similar information but different layouts.

In the KILE task, the bounding box of the field can be directly detected without requiring any additional processing. In the LIR task, field bounding boxes are also detected but an additional clustering of the detected fields into line items is necessary. This clustering process was achieved by adding a new class called *line_item*. The bounding boxes for this class are generated during training. The coordinates of the bounding box are determined based on all the fields corresponding to one line item. During the prediction phase, bounding boxes of this class indicates which LIR fields should be grouped together.

3.1. LIR post-processing

Post-processing of detected LIR fields involves only a grouping procedure. Objects are grouped into line items based on detected *line_item* fields. To determine the line item to which each LIR field belongs, we find the *line_item* field that has the largest overlap in the y-axis with the LIR field. The overlap must exceed a chosen threshold, which, based on experiments, see Table 1, we set at 20% of the height of the *line_item* field. Tables usually span across the entire width of the page, therefore we used only overlap in the y-axis and omitted the x-axis completely. The grouping can be also approached in different ways, it may be beneficial to explore different approaches in the future.

Table 1
Results of a model for different thresholds between line item field and LIR fields.

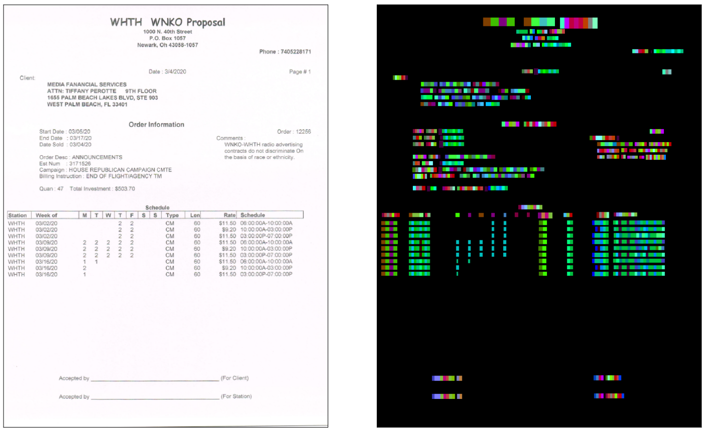
Threshold	0.0	0.1	**0.2**	0.3	0.4	0.5	0.6	0.7	0.8	0.9
F1	0.598	0.601	**0.602**	0.595	0.565	0.559	0.560	0.560	0.558	0.538

3.2. Chargrid

Chargrid was presented in [5] as a novel representation of the documents. *Chargrid* is constructed from character bounding boxes. In the first step, a new image is initialized with the same dimensions as the original document. In the second step, each character is assigned a unique numeric value. In the last step, the area of the box corresponding to a character is filled in the new image by the value assigned to this character.

We adopted this procedure but with some modifications. Instead of replacing the original image, we concatenated *chargrid* representation with the original image. Since we did not have character bounding boxes, we used word bounding boxes from OCR [6], which we divided uniformly by the number of characters. Furthermore, instead of encoding characters into one numeric value, we used three numbers for each character, which led to slightly better results than using one number only. To encode characters into 3 channels, we first select a number base for which there exists a mapping between characters and numbers in this base, such that each character has a value with a maximum of 3 digits. We obtain the base using $\left\lceil \sqrt[3]{len(\text{characters}+1)} \right\rceil$. Then, we assigned each character value in the chosen base. At the first position in the set of characters, we added an *empty character* that represents the document background. Finally, we normalize the encoded values by the base value so that the encoded values fall within the range of $\langle 0, 1\rangle$.

An example of a *chargrid* is in Figure 2. In order to use this representation, it was necessary to increase the number of kernels of the input convolutional layer of the YOLOv8 model.



WHTH WNKO Proposal
1000 N. 40th Street
P.O. Box 1057
Newark, Oh 43058-1057

Phone : 7405228171

Date : 3/4/2020

Page # 1

Client:
MEDIA FANANCIAL SERVICES
ATTN: TIFFANY PEROTTE 9TH FLOOR
1655 PALM BEACH LAKES BLVD, STE 903
WEST PALM BEACH, FL 33401

Order Information

Start Date : 03/05/20
End Date : 03/17/20
Date Sold : 03/04/20

Order : 12256

Comments :
WNKO-WHTH radio advertising contracts do not discriminate On the basis of race or ethnicity.

Order Desc : ANNOUNCEMENTS
Est Num : 3171526
Campaign : HOUSE REPUBLICAN CAMPAIGN CMTE
Billing Instruction : END OF FLIGHT/AGENCY TM

Quan : 47 Total Investment : $503.70

Schedule

Station	Week of	M	T	W	T	F	S	S	Type	Len	Rate	Schedule
WHTH	03/02/20				2	2			CM	60	$11.50	06:00:00A-10:00:00A
WHTH	03/02/20				2	2			CM	60	$9.20	10:00:00A-03:00:00P
WHTH	03/02/20				2	2			CM	60	$11.50	03:00:00P-07:00:00P
WHTH	03/09/20	2	2	2	2	2			CM	60	$11.50	06:00:00A-10:00:00A
WHTH	03/09/20	2	2	2	2	2			CM	60	$9.20	10:00:00A-03:00:00P
WHTH	03/09/20	2	2	2	2	2			CM	60	$11.50	03:00:00P-07:00:00P
WHTH	03/16/20	1	1						CM	60	$11.50	06:00:00A-10:00:00A
WHTH	03/16/20	2							CM	60	$9.20	10:00:00A-03:00:00P
WHTH	03/16/20	1							CM	60	$11.50	03:00:00P-07:00:00P

Accepted by ______________________________ (For Client)

Accepted by ______________________________ (For Station)

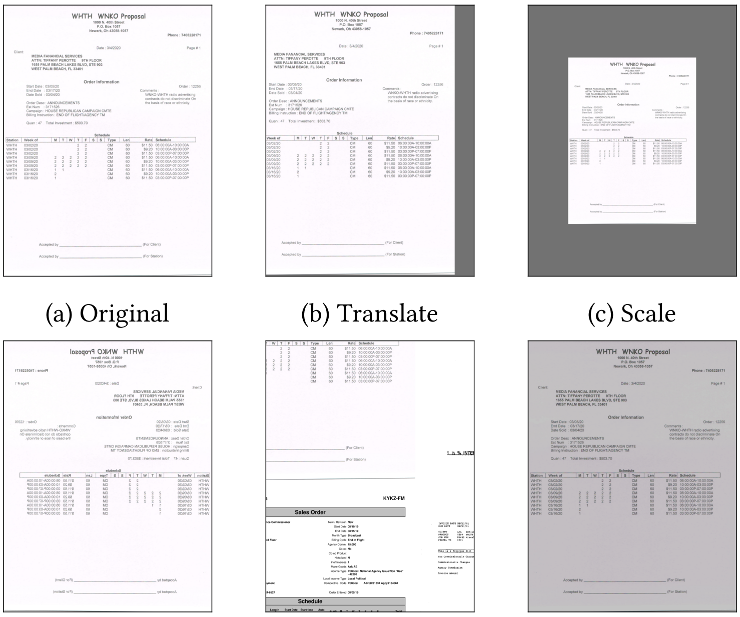
(a) Document

(b) Chargrid

Figure 2: Example of *chargrid* representation of the document.

3.3. Augmentations

Data augmentation is one of the most common practices used during the training of the model to prevent overfitting and improve its ability to generalize by increasing the variability of the input data while simulating real data examples that are missing from the training set. The original YOLOv8 implementation uses several commonly used augmentations. In our experiments, we explored the usage of some of these augmentations, in Figure 3 you can see examples of individual augmentations that were used in experiments.



(a) Original (b) Translate (c) Scale

(d) Left-right flip (e) Mosaic (f) HSV

Figure 3: Examples of the explored augmentations.

The first two augmentations are *translate* and *scale*, both augmentations alter the position of the image in the frame. Translate moves the image in the frame and scale shrinks the image or zooms in on it. If there is no image in the part of the frame after the translation or change of scale, this part is filled with a predefined neutral gray color. Both augmentations simulate the fact that the position of individual fields can largely vary within each frame. *Left-right flip* augmentation reflects columns of the image along the vertical axis. As with the two previous augmentations, this augmentation changes the position of the fields in the frame but also changes the appearance of the text. It is noteworthy that this augmentation does not represent any real cases, but we included it to better understand what is and what is not important for the model. The last geometric augmentation used in experiments is *mosaic* augmentation. *Mosaic* augmentation combines four images into one by cropping one part from each image and stitching them together. This augmentation also alters the position of the fields in the image but also allows the model to learn that fields can be in the document in any context. The last used augmentation was *HSV* augmentation which alters the brightness of the image.

Additional augmentations provided by the YOLOv8 framework are *rotation*, *up-down flip*, *shear*, *perspective*, and *mixup*. We did not use these augmentations in our experiments. *Rotation*, *shear*, *up-down flip*, and *perspective* are similar to the other geometric augmentations that we used. Finally, we believe that the *mixup* is not relevant to our task.

4. Experiments

In this section, we will describe experiments that were conducted in order to determine the best parameters of the model and other factors that can influence the training procedure. Our aim is to get a better understanding of this task and what is beneficial for it.

Experimental Setup. If not stated otherwise, we trained models for 100 epochs with an initial learning rate 1×10^{-3} using optimizer AdamW [7]. An input image size was set to 640×640 pixels and batch size to 16. In experiments, where the input image size is equal to 1280×1280, there was the batch size decreased to 8 due to memory requirements. All models were trained from randomly initialized weights. If not specified otherwise, models were trained on annotated train set and evaluated on annotated validation set. Synthetic and unlabeled datasets were not used. All the provided results are averages over three runs initialized from different seeds and the last checkpoint is always used for evaluation.

The model expects the input image to be a square. Instead of a standard resizing of the image, which would distort the aspect ratio, the shorter side of the image is padded from both sides with a neutral background color.

Augmentations. We decided to explore possible data augmentations because the model trained without them showed clear signs of overfitting. Data augmentation is a common practice, but the specific augmentations used can vary depending on the task. In this experiment, we aimed to determine the most beneficial augmentations for KILE and LIR tasks. We conducted experiments with augmentations illustrated in Figure 3. The two most beneficial augmentations were *mosaic* and *translate* augmentations. Their comparison is in Table 2. The *scale* augmentation also showed some improvements, although not as much as the previous two. Unfortunately, a combination of *translate* and *scale* did not improve the results compared to *translate* alone. On the other hand, the two remaining augmentations did not help with overfitting and in some cases even worsened the results.

It should be noted that all augmentations that have been beneficial are geometric augmentations that alter the position of objects in images. We argue that the model trained without these augmentations strongly relies on the position of the individual classes in the image. However, when the augmentations are applied, the model is forced to obtain information about the object from other visual clues, such as the size, length, and formatting of the text. The inferior performance observed during training with the *HSV* augmentation can be attributed to the loss of details after applying this augmentation. Despite the *left-right flip* being a geometric augmentation that alters the position of objects in the image, there was no improvement in the results. This is expected as this augmentation does not represent any real-world scenario. Additionally, since the augmentation is applied with a probability of 0.5, the model receives mixed information about the structure and appearance of the text. This indicates that model can distinguish the appearance and context of the fields. This is further supported by the fact that the model trained with *left-right flip* augmentation with the probability of 1.0 achieved comparable results with the model trained without it.

In our final setup, the chosen augmentations were applied with the default setting defined in the YOLOv8 config file. For *translate* probability was 1.0 and the image could be translated by a maximum of 10% of its size in each axis. *Mosaic* augmentation was also applied with probability 1.0. Although the *mosaic* augmentation itself achieved the best results, we used *translate* in all subsequent experiments, because training with this augmentation was approximately four times faster than with the *mosaic* augmentation.

Table 2
Comparison of the influence of the two most beneficial augmentations. All models were evaluated on the validation set.

			KILE		LIR	
Model	Translate	Mosaic	AP	F1	AP	F1
YOLOv8n	✗	✗	0.303	0.457	0.252	0.450
YOLOv8n	✓	✗	0.474	**0.601**	0.378	0.561
YOLOv8n	✗	✓	**0.479**	**0.601**	0.380	0.562
YOLOv8n	✓	✓	0.470	0.597	**0.385**	**0.564**

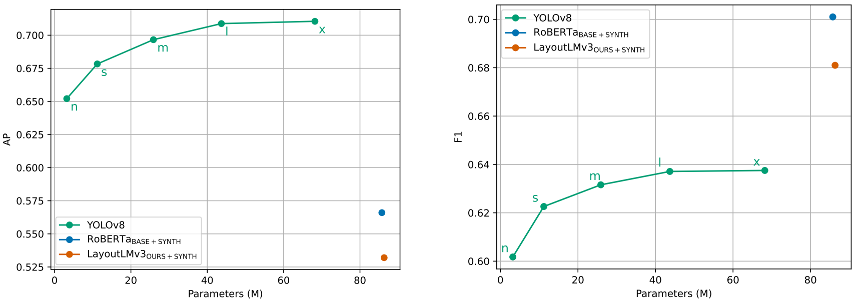
Chargrid In [5] was as an input for the model used only *chargrid* generated from the document. The advantage of *chargrid* is that it can capture even small characters that could be lost in a standard image at low resolution. On the other hand, *chargrid* does not convey information about specific formatting of the text and other visual features on the page e.g. borders of the table. However, these features are presented in the original image.

We decided to compare a model trained only on images, a model trained on *chargrids* only, and a model trained on both. The results of the experiment are in Table 3. The model trained only with *chargrids* achieved better results than the model trained only on images. This indicates that the semantic information contained within the text is more useful than the visual information. However, when both representations were used, the model achieved better results than when using the representations separately. This means that each representation provides unique information.

Table 3
Comparison of different training setups - using images, *chargrids*, or both. All models were evaluated on the validation set.

			KILE		LIR	
Model	Image	Chargrid	AP	F1	AP	F1
YOLOv8n	✓	✗	0.474	0.601	0.335	0.525
YOLOv8n	✗	✓	0.492	0.612	0.347	0.534
YOLOv8n	✓	✓	**0.511**	**0.628**	**0.349**	**0.539**

Model size. YOLOv8 has several variants that differ in the number of parameters. We observed that models with a higher number of parameters were more prone to overfitting, as expected. This tendency was more prominent in the KILE task compared to the LIR task. This behavior can be attributed to the larger number of objects in the LIR task, where tables consist of multiple lines that can differ in appearance. Better results with larger models could be probably achieved with more data and longer training. In Figure 4 are compared YOLOv8 model variants with different numbers of parameters with baseline methods proposed by competition organizers in [1]. Interestingly, even the smallest model with 3.2 M parameters achieved better results than transformer-based models with more than 80 M parameters on the KILE task. This indicates that model size is not the only important factor in successfully addressing this task.



(a) KILE taks

(b) LIR task

Figure 4: Comparison of performance of models with the different number of parameters on KILE and LIR task.

In Table 4 are summarized numbers of parameters and FLOPs of YOLOv8 variants, RoBERTa and LayotLMv3. All variants of YOLOv8 have a lower number of parameters compared to RoBERTa and LayotLMv3. However, only the smallest variants of YOLOv8 have a lower number of FLOPs than RoBERTa and LayotLMv3.

Table 4
Comparison of a number of parameters and FLOPs of the models. Values provided for YOLOv8 are calculated for a model with image input size 1280 × 1280 and values provided for RoBERTa and LayotLMv3 are with 512 tokens at the input. The number of parameters for RoBERTa and LayotLMv3 is in the format: *model parameters + embedding parameters*.

Model	Parameters (M)	FLOPs (B)
YOLOv8n	3.157	35.430
YOLOv8s	11.167	115.267
YOLOv8m	25.903	317.282
YOLOv8l	43.692	662.971
YOLOv8x	68.230	1034.189
RoBERTa	86.131 + 91.812	87.747
LayoutLMv3	87.402 + 91.812	123.407

Other. For completeness, we also verified the obvious parameters for which we expected an improvement in the results. First, we verified the impact of the size of the input image. From the results in Table 5, it can be seen that the increase in size greatly improved results. We believe it is caused by the fact that details that are lost in low resolution images are important for detection.

Table 5
Comparison of the effect of input image size on the performance. All models were evaluated on the validation set.

	Image	KILE		LIR	
Model	Size	AP	F1	AP	F1
YOLOv8n	640	0.476	0.600	0.334	0.557
YOLOv8n	1280	**0.594**	**0.672**	**0.345**	**0.587**

It is common that models trained from weights pre-trained on different dataset learn faster and have better results. YOLOv8 provides weights pre-trained on the COCO dataset [8]. Even though objects in the COCO dataset and images of documents are very different, starting from the pre-trained model was beneficial as indicated by results in Table 6.

Table 6
Comparison of the effect of weight initialization on the results. All models were evaluated on the validation set.

		KILE		LIR	
Model	Weights	AP	F1	AP	F1
YOLOv8n	Random	0.474	0.601	0.378	0.561
YOLOv8n	COCO	**0.513**	**0.632**	**0.395**	**0.576**

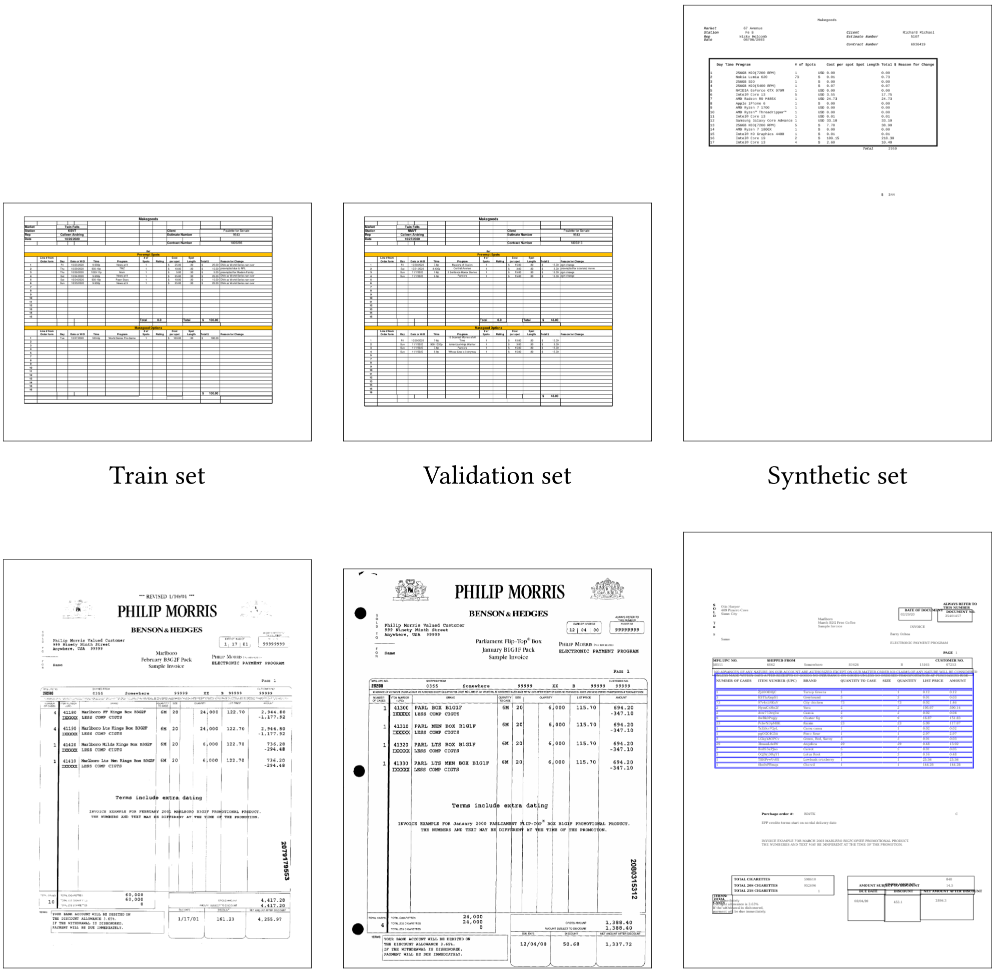
The dataset contains a large set of syntactic documents and unlabeled documents. It is often the case that the more data, the better the result. In Table 7, there are the results of a model trained on a train set only and of a model trained on the train set together with a synthetic set. Unexpectedly, the model trained on the train and synthetic set performed worse on the validation set than the model trained on the train set only. When trained only on synthetic data model showed signs of overfitting and the results for the validation set were close to zero.

Table 7
Results of a model trained on a synthetic dataset. All models were evaluated on the validation set.

		KILE		LIR	
Model	Training set	AP	F1	AP	F1
YOLOv8n	train	**0.476**	**0.600**	**0.371**	**0.557**
YOLOv8n	train+synthetic	0.417	0.552	0.354	0.527

In some cases, we observed that the documents in the synthetic set are visually different from the documents in the validation set, and train set, even when they belong to the same cluster. Examples can be viewed in Figure 5. This is not an issue for baseline methods RoBERTa and LayoutLMv3 presented in [1] that both use textual information, but it can be an issue for YOLOv8 which mainly relies on visual information. The results indicate that YOLOv8 is not able to sufficiently generalize information from documents that are visually different from the validation set documents, even when they contain semantically similar information at similar positions in the document.

In object detection, there are no commonly used techniques for pre-training of the model on unlabeled data, therefore we did not utilize unlabeled dataset.



Train set

Validation set

Synthetic set

Train set

Validation set

Synthetic set

Figure 5: Examples of documents from the same layout clusters from different datasets. The first row represents documents from the cluster: 293 and second row documents from the cluster: 554.

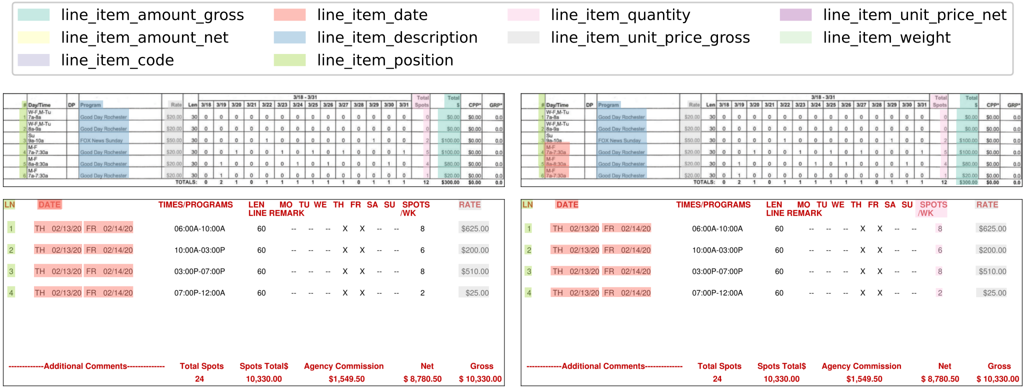
5. Results

The final training setup was based on the experiments conducted in the previous section. We used the largest variant of the model, YOLOv8x pre-trained on the COCO dataset. Input image had resolution 1280×1280 and was concatenated with *chargrid*. As augmentation was used only *translate* augmentation. The initial learning rate was set to 1×10^{-3}, the batch size was 8, and the AdamW optimizer was used. The model used for the KILE task was trained for 30 epochs and the model for LIR was trained for 50 epochs.

Table 8
Comparison of results of baseline models presented in [1] and YOLOv8x on KILE & LIR tasks on validation set.

	KILE				LIR			
Model	AP	F1	Prec.	Recall	AP	F1	Prec.	Recall
RoBERTa$_{\text{BASE}}$ [9]	0.552	0.688	0.681	0.694	0.552	0.688	0.709	0.668
RoBERTa$_{\text{BASE+SYNTH}}$ [9]	0.566	0.689	0.680	0.698	**0.567**	**0.701**	**0.721**	**0.683**
LayoutLMv3$_{\text{OURS}}$ [10]	0.513	0.657	0.651	0.662	0.546	0.666	0.688	0.645
LayoutLMv3$_{\text{OURS+SYNTH}}$ [10]	0.532	0.674	0.680	0.668	0.564	0.681	0.704	0.659
DETR$^{\text{table}}$ + RoBERTa$_{\text{BASE}}$ [11]	-	-	-	-	0.553	0.682	0.719	0.648
YOLOv8x	**0.716**	**0.772**	**0.747**	**0.799**	0.435	0.638	0.603	0.677

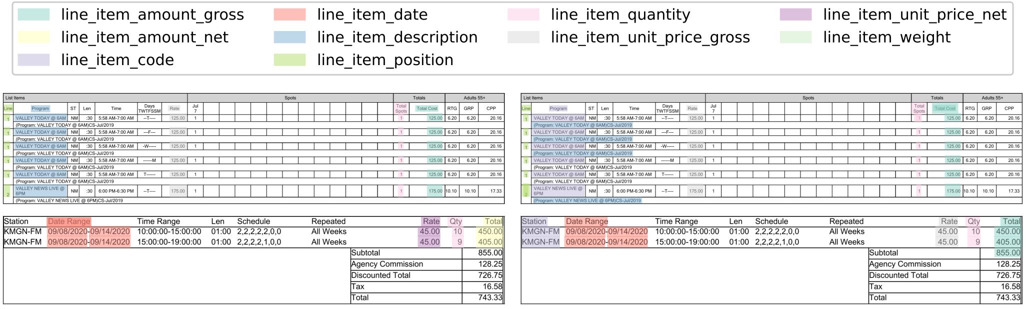
Results on the validation set are presented in Table 8. The model achieved superior results compared to the baseline methods on the KILE task. However, the model performed suboptimally on the LIR task. This could be attributed to various factors, one of which is the high number of false positive detections. In most tables not all columns are relevant, but the model was always not able to effectively distinguish which columns are important for the given table and which are not. As a result, there were excessive detections of fields in the tables, leading to a large number of false positives. Another common error that contributed to the high number of false positives, was the miss-classification of fields. An example of excessive detections is shown in Figure 6 and an example of miss-classifications is shown in Figure 7. In the case of miss-classified fields, the error often occurred with semantically similar classes e.g. *gross*, *net*, etc.


line_item_amount_gross
line_item_date
line_item_quantity
line_item_unit_price_net
line_item_amount_net
line_item_description
line_item_unit_price_gross
line_item_weight
line_item_code
line_item_position

(a) Annotations

(b) Predictions

Figure 6: Example of excessive predictions in tables.


line_item_amount_gross
line_item_date
line_item_quantity
line_item_unit_price_net
line_item_amount_net
line_item_description
line_item_unit_price_gross
line_item_weight
line_item_code
line_item_position

(a) Annotations

(b) Predictions

Figure 7: Example of miss-classified fields and excessive predictions in tables.

In Table 9, there are presented results on the test set. Results on the KILE task are consistent with the validation set. Nevertheless, the YOLOv8 results on the LIR task decreased more than the results of the baseline methods.

Table 9
Comparison of results of baseline models presented in [1] and YOLOv8x on KILE & LIR tasks on test set.

	KILE				LIR			
Model	AP	F1	Prec.	Recall	AP	F1	Prec.	Recall
RoBERTa$_{\text{BASE}}$ [9]	0.534	0.664	0.658	0.671	0.576	0.686	0.695	0.678
RoBERTa$_{\text{BASE+SYNTH}}$ [9]	0.539	0.664	0.659	0.699	**0.583**	**0.698**	**0.710**	**0.687**
LayoutLMv3$_{\text{OURS}}$ [10]	0.507	0.639	0.636	0.641	0.531	0.661	0.682	0.641
LayoutLMv3$_{\text{OURS+SYNTH}}$ [10]	0.512	0.655	0.662	0.648	0.582	0.691	0.709	0.673
DETR$^{\text{table}}$ + RoBERTa$_{\text{BASE}}$ [11]	-	-	-	-	0.560	0.682	0.706	0.660
YOLOv8x	**0.680**	**0.747**	**0.735**	**0.759**	0.383	0.597	0.599	0.595

This fact motivates us to analyze the results in more detail. Both validation and test sets can be divided into three subsets based on the number of documents from the same cluster appearing in the train set. The results for each subset can be found in Table 10. It is obvious that YOLOv8 compared to baseline methods performed worse on the zero-shot subset as the relative decrease in score is greater for YOLOv8. On many-shot and few-shot subsets, the relative change for all models is similar.

Table 10
Comparison of results on subsets of the validation set in the first half of the table and on subsets of the test set in the second half. Values in brackets indicate a relative change in the percentage of the corresponding metrics compared to the results on all documents. AP values are the results for the KILE task and F1 values are the results for the LIR task. The results with the largest relative increase or the smallest decrease are highlighted in bold.

	All		Many-shot		Few-shot		Zero-shot	
Model	AP	F1	AP	F1	AP	F1	AP	F1
YOLOv8x	0.716	0.646	0.800 (+12)	0.703 (+9)	**0.730 (+2)**	**0.613 (-5)**	0.488 (-32)	0.384 (-41)
RoBERTa$_{\text{BASE+SYNTH}}$	0.566	0.701	0.624 (+10)	**0.792 (+13)**	0.566 (+0)	0.608 (-13)	**0.406 (-28)**	**0.465 (-34)**
LayoutLMv3$_{\text{OURS+SYNTH}}$	0.532	0.681	**0.800 (+50)**	0.599 (-12)	0.524 (-2)	0.530 (-22)	0.365 (-31)	0.431 (-37)
YOLOv8x	0.680	0.597	**0.814 (+20)**	**0.683 (+15)**	**0.646 (-5)**	**0.489 (-18)**	0.393 (-42)	0.402 (-33)
RoBERTa$_{\text{BASE+SYNTH}}$	0.539	0.698	0.615 (+14)	0.760 (+9)	0.499 (-7)	0.568 (-19)	**0.384 (-29)**	**0.631 (-10)**
LayoutLMv3$_{\text{OURS+SYNTH}}$	0.512	0.691	0.601 (+17)	0.773 (+12)	0.465 (-9)	0.538 (-22)	0.338 (-34)	0.586 (-15)

5.1. Future Work

The YOLOv8 model proved to be useful for the extraction of information from documents, but we believe that there is still room for improvement. We have not sufficiently explored the possibilities of using synthetic data and unlabeled data for training, because our initial experiments did not show any significant differences while using them but the training was substantially longer. While pre-training models on unlabeled data is a crucial step in NLP tasks, it is not a common practice in object detection. However, there could be potential opportunities to explore and adapt this procedure for improving the performance of the object detection model as well.

Augmentations present another area with potential opportunities for improvement. We observed that geometric augmentations had a positive impact on model training. However, we used standard augmentations used in object detection. In the future, it may be beneficial to design augmentations specifically tailored for this task.

Last but not least the post-processing of the detected objects can be possibly improved given that we did not use any sophisticated methods. Especially in the LIR task, we observed a high number of false positive detections. The use of a filtration method based on a priori occurrences of different object classes in one table could potentially lead to better results.

6. Conclusion

Document information extraction using transformer-based models is a common approach, typically treating the task as an NLP problem. However, in our work, we demonstrated that the

KILE and LIR tasks can be effectively addressed as object detection tasks using the CNN model.

We mainly focused on exploring various factors that can impact the training process, such as image size and different augmentations. Additionally, we successfully demonstrated that *chargrid* representation concatenated with input image is beneficial for training.

Furthermore, we compared the results of YOLOv8 with baseline methods, analyzed the results, and provided suggestions for future work. To evaluate the performance of our proposed approach, we compared the results of YOLOv8, with baseline methods. On the KILE task, YOLOv8 surpassed the baseline methods, achieving an 0.716 AP. However, for the LIR task, YOLOv8 did not outperform the baseline methods, the best achieved F1 score was 0.638.

Acknowledgments

Computational resources were provided by the e-INFRA CZ project (ID:90254), supported by the Ministry of Education, Youth, and Sports of the Czech Republic. The work was supported by the University of West Bohemia, project No. SGS-2022-017.

References

[1] Š. Šimsa, M. Šulc, M. Uřičář, Y. Patel, A. Hamdi, M. Kocián, M. Skalický, J. Matas, A. Doucet, M. Coustaty, D. Karatzas, DocILE Benchmark for Document Information Localization and Extraction, in: 17th International Conference on Document Analysis and Recognition, ICDAR 2021, San José, California, USA, August 21–26, 2023, Lecture Notes in Computer Science, Springer, 2023.

[2] Š. Šimsa, M. Uřičář, M. Šulc, Y. Patel, A. Hamdi, M. Kocián, M. Skalický, J. Matas, A. Doucet, M. Coustaty, D. Karatzas, Overview of DocILE 2023: Document Information Localization and Extraction, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, N. Ferro (Eds.), Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023), LNCS Experimental IR Meets Multilinguality, Multimodality, and Interaction., 2023.

[3] Y. Baek, D. Nam, S. Park, J. Lee, S. Shin, J. Baek, C. Y. Lee, H. Lee, Cleval: Character-level evaluation for text detection and recognition tasks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 564–565.

[4] G. Jocher, A. Chaurasia, J. Qiu, YOLO by Ultralytics, 2023. URL: https://github.com/ultralytics/ultralytics.

[5] A. R. Katti, C. Reisswig, C. Guder, S. Brarda, S. Bickel, J. Höhne, J. B. Faddoul, Chargrid: Towards understanding 2D documents, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 4459–4469. URL: https://aclanthology.org/D18-1476. doi:10.18653/v1/D18-1476.

[6] Mindee, doctr: Document text recognition, https://github.com/mindee/doctr, 2021.

[7] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, arXiv preprint arXiv:1711.05101 (2017).

[8] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft coco: Common objects in context, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer, 2014, pp. 740–755.

[9] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

[10] Y. Huang, T. Lv, L. Cui, Y. Lu, F. Wei, Layoutlmv3: Pre-training for document ai with unified text and image masking, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 4083–4091.

[11] B. Smock, R. Pesala, R. Abraham, PubTables-1M: Towards comprehensive table extraction from unstructured documents, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 4634–4642.