IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 14 February 2025, accepted 12 March 2025, date of publication 26 March 2025, date of current version 16 April 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3555155

RESEARCH ARTICLE

On the Generation of Synthetic Invoices for Training Machine Learning Models

ROLANDAS GRICIUS AND IGORIS BELOVAS
Institute of Data Science and Digital Technologies, Vilnius University, 08412 Vilnius, Lithuania
Corresponding author: Rolandas Gricius (Rolandas.Gricius@mif.stud.vu.lt)

This work was supported by the Vilnius University.

ABSTRACT Currently, the problem of generating synthetic financial documents is particularly acute. Extending recent research on the topic, we present an enhanced tool for invoice generation. The primary motivation is the need for invoice corpora for machine learning in accounting automation. The generation produces synthetic invoices with randomized layouts and contents. As content fields are generated, annotations for supervised machine learning are saved along with the generated invoice, thus solving the problem of labor-intensive annotation tasks. The content and layout diversity is evaluated and compared to empirical and synthetic invoice corpora using SELF-BLEU, Alignment, and Overlap metrics. We have validated the stability of the modeling statistically. The modeling is consistent and reproducible. The final assessment is that the diversity of the generated invoices is on par with the real-world ones and, by most metrics, exhibits superiority over the foregoing ones.

INDEX TERMS Dataset generation, entity recognition, financial documents, machine learning.

I. INTRODUCTION

Nowadays, the rapid growth of the flow of financial documents no longer allows them to be effectively managed by traditional methods. The need for new approaches has given rise to automatic processing based on machine learning technologies (see [1], [2] and the references therein). At the same time, the application of machine learning algorithms requires a large amount of training data to be efficient [3]. However, gaining access to a large number of certain classes of financial documents may encounter difficulties.

Indeed, it is currently very complicated to get a sufficiently big corpus of varied enough invoices to train machine learning models. The main challenges are:

- *privacy* - invoices contain personal data and may be subject to General Data Protection Regulation (GDPR) rules, which require explicit consent from every data subject (person).
- *trade secret* - invoices inherently contain sensitive financial and commercial information (relations between customers and suppliers, bank transactions, products supplied, etc.);
- *variety* - invoices issued and received by a company are not varied enough in their contents and format. It means that these invoices may over-fit machine learning models, which complicates the creation of a corpus;
- *annotation* - even if a sufficiently large amount of varied enough invoices is available, most machine learning algorithms in natural language processing (NLP) are supervised learning ones, so the labor-intensive task of annotating (labeling) invoice fields is required.

These challenges are not unique to a specific language: even the *lingua franca* of international business communication - the English language - lacks an annotated invoice corpus for information extraction.

A. INVOICE CORPORA EXAMPLES

The principal features of four prominent corpora (SROIE, RVL-CDIP, ZUGFeRD, IDSEM) are as follows:

- SROIE dataset was created for ICDAR 2019 (*International Conference on Document Analysis and Recognition*) Competition on Scanned Receipt OCR and Information Extraction [4]. The dataset consists of 1000 scanned receipt images and annotations. It has ground truth (annotations) for the main fields of the receipt but no full text contained in the image. Also, it is comprised not of invoices but of cash receipts. They have most

The associate editor coordinating the review of this manuscript and approving it for publication was Jad Nasreddine.

invoice data fields (note that buyer information typically is not present) and a simpler layout.
- RVL-CDIP dataset consists of 400,000 grayscale document images belonging to 16 classes, with 25,000 images per class [5]. One of the classes is the invoice, so there are 25,000 invoices. The only metadata provided together with the image is the document class. The images have low resolution.
- ZUGFeRD invoice corpus - a corpus of pdf invoices in ZUGFeRD/Factur-X format which embeds invoice data in xml format into pdf, allowing to extract the data easily [6]. So, in this case, we have together invoice text (as pdf) and source of truth (as xml). The corpus contains invoices in English, French, and German. Unfortunately, the size of the dataset is small ($\sim$ 100 entries).
- IDSEM invoice corpus consists of bills for energy consumption in Spanish households [7]. It contains an impressive amount of 75,000 entries with annotations of 86 different data labels. Unfortunately, it includes just nine different invoice templates from eight companies.

The main features and drawbacks of the described corpora are summarised in Table 1.

B. LITERATURE SURVEY

The problems of synthetic invoice generation have been intensively covered in the scientific literature. Next, we survey the most significant recent studies.

Blanchard et al. [8] created a bill-type document generator to produce several types of invoices (in French or English). Two ways for the layout generation are presented in the study. The first option is to create templates from real invoices, manually describing positions and sizes of information blocks contained in the invoice. The second option for the layout automatically generates random positions for information blocks. Data for the invoice fields are generated using regular expressions (e.g., for fields such as date or invoice number) or taken from a database included in the project (e.g., for addresses).

Belhadj et al. [9] enhanced this generator, allowing more semi-structured document types: besides invoices, it generates receipts and payslips. They also introduced significantly improved layout randomization and applied metrics to evaluate the quality of synthesized datasets.

Bensch et al. [10] needed document dataset for information extraction models comparison. They created a template-based document generator and used ten different templates to generate 12000 documents. To achieve variety, they shifted each field by a random offset. It is not a realistic scenario (but it fits their purpose).

Schulze et al. [11] created an invoice (and two other document types) generator. The main novelty of their approach is that noise patterns (frequently encountered in real industrial data) have been added.

Sánchez et al. [7] created an electricity invoice generator for the Spanish electricity market. This software was used to produce the IDSEM dataset described above. They focused on generating large amounts of invoices as similar to real ones as possible. They took real invoice templates from eight electricity companies and used them as input for the generator. However, this led to a very limited amount of layout variety.

In the commercial application space, company *Provectus* built an eponymous generator [12], with a random filling of data and various template formats. All the generated invoice examples are in English. The source code is not available for public access.

Note that for layout generation, machine learning approaches have been attempted. There is a certain progress in this direction, cf. [13], [14], [15], [16].

C. AIMS

Our objective is to develop a method and tool for generating synthetic invoice corpus intended to train machine learning models for information extraction in the accounting field. Indeed, since there is currently no published research considering the specifics of less-represented languages (e.g., Lithuanian), we have chosen it as our model object. Note that accounting is an up-and-coming field for the implementation of NLP-based automation. We explore the possibility of adopting and adapting the most recent state-of-the-art research on automated invoice generation (cf. literature survey). The novelty of our approach is expanded in the Method section.

The paper is organized as follows. The first part is the introduction. Section II addresses the benchmarks used to evaluate generation results. Section III details the method used for invoice generation. Section IV presents the results and describes their validation. The last part is devoted to the discussion and concluding remarks.

II. BENCHMARKS

An important part of the validation of the results is a selection of relevant benchmarks. As our generation process is comprised of layout generation and content generation, we need to cover both aspects of generation by selecting applicable benchmarks. It would be beneficial to compare our results to earlier results in the literature, so it is worth surveying the metrics used earlier.

Blanchard et al. [8] manually selected more than 3000 images generated by their generator. The selection was subjective "*by selecting those which seemed to us quite representative of the sought variations*". As this is a subjective evaluation, no useful metric could be derived nor results directly compared.

Belhadj et al. [9] used two metrics Alignment [17] and Overlap [17] to estimate the variation in the generated layout. Additionally, they introduced a new metric SCR (Semi-structured document Compositions Ratio [9]). Unfortunately, they have not calculated this metric for reference SROIE dataset citing it being very costly in time and not having enough information to calculate it. We will not use SCR metric in our evaluation. Authors also used SELF-BLEU [18] to estimate the content diversity.

TABLE 1. Characteristics of existing invoice corpora.

Dataset	SROIE	RVL-CDIP	ZUGFeRD	IDSEM
Size	1000	25000	100	75000
Language	Eng	Eng	Eng, Ger, Fr	Span
With image	Yes	Yes	Yes	Yes
With text	No	No	Yes	Yes
With entity annotations	Yes	No	Yes	Yes
Drawbacks	No extracted text, moderate size	No extracted text, no annotations	Very small	Only 9 invoice templates

Bensch et al. [10] used generation results to train two machine learning models: Chargrid and SpacyGrid. Authors concluded that "*it can be used to train, compare and evaluate different models for information extraction of invoices in PDF format*".

Sánchez et al. [7] created graphs with the distribution of the main numerical fields and visually (sic!) confirmed that they adhere to a normal or some other expected distribution. Note that the authors (in contrast to the current research) in their work did not test corresponding statistical hypotheses (i.e., goodness of fit tests).

Finally, Schulze et al. [11] and Provectus [12] do not provide any data about layout or content diversity.

Summarizing the above information, we can argue that the most comparable approach is to reuse metrics used in [9] and apply statistical hypotheses testing for the normality of the results. We strive to evaluate two main aspects of the generated documents: the layout diversity and the text content diversity. To compare our generation quality with results reported in Belhadj et al. [9], which we are building upon, we selected to use three metrics which the authors used: Alignment [17], Overlap [17] to estimate the variation in the generated layout and SELF-BLEU [18] to estimate the content diversity. As BLEU is more tailored to fluent texts than text fragments used in form-like documents such as invoices, we will also calculate newer token-level metric, based on contextual embeddings, BERTScore [19]. We computed BERTScore for our dataset; however, since prior works did not report this metric, we focused our comparison on Self-BLEU.

A. LAYOUT DIVERSITY

Alignment and Overlap metrics focus on the diversity of the positioning of data blocks in the same class. The Alignment score is measured by summing Manhattan distances between the top-left and bottom-right coordinates of two compared data blocks. Let us define block B_i as a pair of top-left and bottom-right points $B_i = (p_i^{tl}, p_i^{br})$, where point p is defined by a pair of coordinates $p = (x, y)$. Then, the Alignment score is,

$$\text{Alignment}(B_i, B_j) = \text{Mht}(p_i^{tl}, p_j^{tl}) + \text{Mht}(p_i^{br}, p_j^{br}), \quad (1)$$

where

$$\text{Mht}(p_a, p_b) = |x_a - x_b| + |y_a - y_b|. \quad (2)$$

The Alignment score of two identically positioned blocks equals 0, while the maximum score is achieved for small blocks in the diametrically opposite corners of the page. Note that the maximum score is twice the sum of page width and height. Thus, the normalized measure is

$$\text{Alignment\_Normalized}(B_i, B_j) = \frac{\text{Alignment}(B_i, B_j)}{2 \times (\text{width} + \text{height})}. \quad (3)$$

The Overlap score is computed as the ratio of overlapping areas of two data blocks,

$$\text{Overlap}(B_i, B_j) = 1 - \frac{2 \times \text{overlap\_area}(B_i, B_j)}{\text{area}(B_i) + \text{area}(B_j)}, \quad (4)$$

where B_i and B_j are the data blocks scored. The overlap_area of two data blocks B_i and B_j is defined as the intersection area of these two blocks. For non-intersecting blocks, overlap_area will be equal to 0. The Overlap score then will be equal to 1. In the opposite extreme case, where one block fully covers another block, the overlap_area will be equal to the area of the smaller one. If both overlapping blocks are identical, the Overlap score will reach its minimum value of 0.

Alignment and Overlap normalized scores are higher for positional diversity, equalling 0 for total alignment or overlap in the same positions on the page and 1 for maximal positional diversity.

B. CONTENT DIVERSITY

SELF-BLEU measures the differences between generated phrases of the same class. It is based on a sentence BLEU score, which assesses how similar two phrases are. To get the SELF-BLEU score, the BLEU score is calculated for every generated phrase, taking all other same-class phrases as a "reference translation". Next, the SELF-BLEU is calculated as an average of BLEU scores. More specifically, for the document set having n documents with m classes of phrases,

we have

$$\text{SELF-BLEU}_{class} = \frac{1}{n}\sum_{i=1}^{n}\text{BLEU}(g_i, G \setminus g_i),$$
$$\text{SELF-BLEU}_{doc\_set} = \frac{1}{m}\sum_{class=1}^{m}\text{SELF-BLEU}_{class}, \quad (5)$$

where n stands for the number of elements in the class of texts G and g_i stands for i-th generated text, $1 \leqslant i \leqslant n$; m is the number of classes. Note that a lower score means higher diversity, with 0 indicating no even partial matches (thus maximum diversity) and 1 indicating full match. SELF-BLEU takes into account only full words and ignores punctuation.

BERTScore computes a similarity score for each token between generated phrases of the same class. Token can be a complete word (for short commonly found words) or part of it. Tokens include punctuation marks as well. Unlike BLEU, instead of exact matches similarity is computed using cosine similarity between contextual embeddings. BERTScore is comprised of precision (P_{BERT}), recall (R_{BERT}) and F_1 (F_{BERT}) scores:

$$P_{BERT} = \frac{1}{|\hat{x}|}\sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} x_i^{\top}\hat{x}_j,$$
$$R_{BERT} = \frac{1}{|x|}\sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} x_i^{\top}\hat{x}_j,$$
$$F_{BERT} = 2\frac{P_{BERT}R_{BERT}}{P_{BERT} + R_{BERT}} \quad (6)$$

III. METHOD

A. BACKGROUND

We extend the work of [9], who presented a method and tool for invoice generation with randomized layouts and random synthetic data. We introduce easily localizable, user-supplied data fed (with the fallback to synthetic data) invoice and ground truth annotations generation tool. We improved the randomization and introduced a two-stage process, adding the possibility of the inclusion of user-supplied data via gazetteers.

The internal format for document generation is selected to be HTML, for its flexibility, openness, and popularity. Ground truth annotations are stored as HTML element attributes and can be easily extracted together with elements' coordinates for the rendered document. After the HTML rendering document is printed in pdf format, annotations are saved in xml format. This produces a set of documents and their annotations suitable for machine learning. The process is illustrated in Figure 1.

The final results are digitally produced invoices without any noise, distortions, or rotations. This perfectly accounts for the most common scenario when digitally produced documents are exchanged directly, e.g. via e-mail, downloaded from the website, or exchanged using other file transfer means. Another common scenario is when invoices

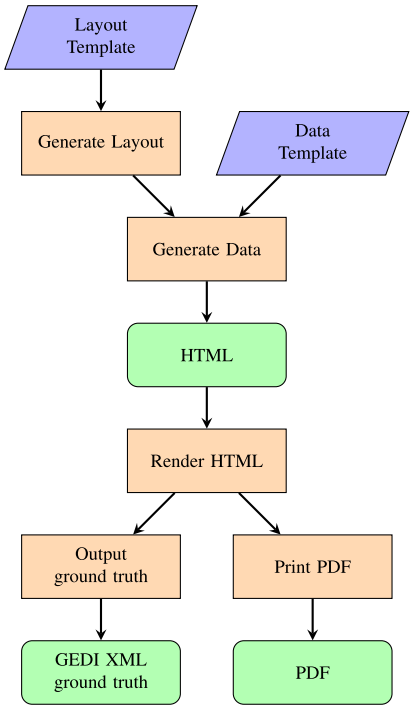


FIGURE 1. Document and annotations generation workflow.

are printed and then scanned or photographed, introducing potential skew, blur, and other imperfections. This can be easily accommodated by applying additional post-processing using one of the several image augmenting libraries such as Augraphy [20], Albumentations [21] or AuGly [22].

B. NOVELTY

The literature survey (see section I-B) has shown a rather limited presence of works dedicated to the construction of instruments for the generation of synthetic invoices. Even the latest studies were built on using rigid hardcoded templates and insufficient randomization.

We propose to split the generation of a document into two separate stages: the layout generation and the data generation. This will allow us to address the layout and data aspects of the document independently. As a consequence, any of these parts can be updated with improved ones at any time later.

In the first stage, the randomized layout of the document is generated. In the second stage, this layout may be used once or several times to be populated with generated data. The rationale for this separation lies in emulating real-world scenarios when the company issuing the document reuses the same document template for a while.

The document template repetition is present in existing invoice corpora. For example, [23] found that in the SROIE dataset, only 301 out of 1000 receipts have unique templates and other receipt templates are encountered up to 76 times (i.e., the biggest group has 76 receipts based on the same template). IDSEM dataset has driven repetition to the extreme, as its 75000 invoices are based only on nine different invoice templates.

Unfortunately, template repetitions are not always accounted for when splitting datasets into training and testing sets or when comparing information extraction results on already seen and unseen templates. [23] compared the state-of-the-art information extraction models, such as BERT, LayoutLM, LiLT, and their newer derivatives, and found that the average F1 score drops significantly from 96.38 on official splits to 88.78 on the split where test set has no document with a template from the training set.

C. LAYOUT GENERATION

We use the basic document layout model similar to the one presented in [9]. A document (page) is modeled as a vertical sequence of sections by default. There is also an option to place two or more sections horizontally. We expect to find one or more (usually related) fields in every section.

The fields can be of several types: pictures (company logos, other illustrations), tables (typically product table in the invoice), and most importantly, key-value fields comprising optional keywords and an optional value.

The selected format for describing the document layout and contents (HTML) has a very rich layout description language, CSS, which features will be used to define the layout of the blocks and shifts in their vertical and horizontal positions. For small random shifts of the sections and their order variation in the document layout template, the popular templating engine *Jinja*[1] is used. The same engine is employed during the data generation step to fill the template with data (see the next section). Shuffling the block order we get The layout randomization is achieved by shuffling (horizontally or vertically) block order according to the instructions in the layout template. The example of a basic template for an invoice is given below:

```
[% extends "A4_document.html" %]

[% block contents %]

    <!-- include section "top" -->
    [% include "top.html" %]

    <!-- sections "common" and "buyer" can
    be side by side or one after another -->
    <random-direction random-order>
        [% include "common.html" %]
        [% include "buyer.html" %]
    </random-direction random-order>

    [% include "products.html" %]

    <vertically random-order>
        [% include "bottom.html" %]
        [% include "seller.html" %]
    </vertically>

[% endblock %]
```

On a smaller scale, more randomization is added by providing configurable top and left offsets for content blocks, as proposed in [8]. These offsets are randomly calculated during the layout template generation process. The process is controlled by the configuration file parameters. The example of the top offset inclusion into the Jinja HTML template is presented further,

```
<div style="height: {{ _common_top }}px">
</div>
```

The offset datapoint `_common_top` is randomly generated. This process is described in the next section.

For the testing of our invoice generation tool, we produced five separate templates and configured their randomization.

TABLE 2. Benchmarks comparison of real and synthetic datasets (1000 documents each).

Dataset		SROIE	G_Invoices	NewDocGen
Alignment	↑	0.48	0.14	0.19386
Overlap	↑	0.998	0.997	0.90676
SELF-BLEU	↓	0.44	0.29	0.22717

D. DATA GENERATION

Although the invoices are synthetic, the application of some real-world data would make them more realistic. We use several methods for the data generation. Some fields are generated using random data from a pre-set interval or randomly chosen from the list. A gazetteer option is available for the company names, company codes, addresses, and contact data. Personal names are generated using a mixed approach - first and last names from the list are randomly sampled and combined. Next, similarly to [24], we chose to use the well-known *Faker* library[2] to generate standard data fields.

The rules of the data field generation are described in the data template (in YAML format, a superset of the well-known JSON format). According to the rules, the code generates data and submits it to the Jinja templating engine to finish the document generation in HTML format. The data and invoice generation processes are repeated as often as requested to produce several documents using the same layout template. Next, we will provide three examples of data field generation.

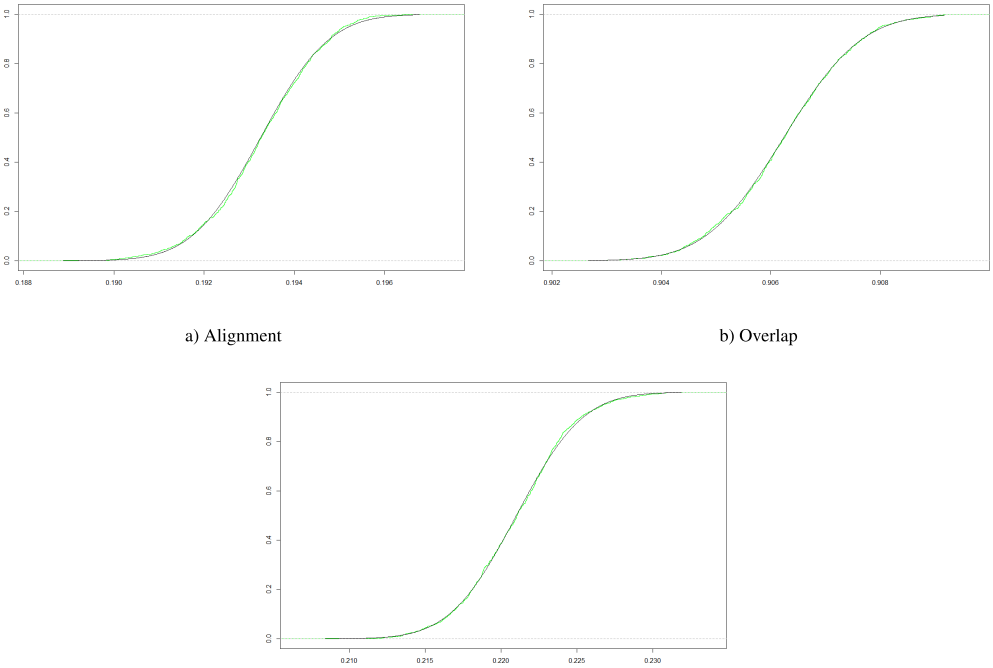
1) The top offset - calculated using the expression
   `_common_top: random.randint(0, 50)`
2) *Faker* library example for generating bank account number:
   `seller_account: fake.iban()`
3) Using gazetteer to include company data:
   ```
   ? seller_name, seller_code,
     seller_address, seller_email,
     seller_phone_no, seller_contact
   : <lithuanian_companies.csv
   ```
   Here `lithuanian_companies.csv` stands for the gazetteer file name.

[1]https://pypi.org/project/Jinja2/

[2]https://pypi.org/project/Faker/

TABLE 3. General and Gaussian-specific Lemeshko's tests of composite normality hypotheses for diversity metrics with the significance level $P = 0.05$ and sample size $n = 10^3$.

	$\hat{\Theta}$		General test		Specific test	
Metric	$\hat{\mu}$	$\hat{\sigma}$	$D_n(\hat{\Theta})$	$D_P(n, \hat{\Theta})$	$S_K(n, \hat{\Theta})$	C_P
Alignment	0.19325	0.00119	0.02344	0.02979	0.74657	0.90835
Overlap	0.90624	0.00112	0.01716	0.02792	0.54805	0.90835
SELF-BLEU	0.22102	0.00345	0.02408	0.02851	0.76663	0.90835



a) Alignment

b) Overlap

c) SELF-BLEU

FIGURE 2. Empirical distribution functions (green curves) and hypothetical (gaussian) distribution functions (black curves) for Alignment, Overlap and SELF-BLEU metrics.

The generated data are substituted into the template, while the data annotations are stored in HTML attributes. Next, HTML is rendered, producing the final layout and contents of the document. As a final result, two files per document are generated: a ground truth file in XML format with all fields, their labels and positions, and a PDF file with the graphical representation of the document.

State-of-the-art AI-based approaches have been covered in Xu Guo and Yiqiang Chen 2024 survey [25]. Approaches they review have comparable diversity measured by Self-BLEU score. However, these methods bring new challenges specific to Generative AI: correctness (generated data is not from the class requested) and hallucination (generated data is not only inaccurate but completely disconnected from reality). These unsolved issues compel us to refrain from the application of these approaches in our current work.

IV. RESULTS

We compare empirical and synthetic data in order to evaluate the quality of the generated invoices. For the SROIE English language receipt empirical dataset, benchmarks were calculated by [9]. Next, [9] generated a dataset of synthetic invoices (G_Invoices), which we also employ in the comparison. Both datasets (consisting of 1000 entries each) we use as a baseline.

In order to test our invoice generation tool, we have produced the same amount of synthetic documents as in the SROIE and G_Invoices datasets. The script for the generation is provided in the published code repository together with the tool source.[3] The new dataset is designated as *NewDocGen*. The results obtained are presented in Table 2. Note that the arrows point in the direction where the results are better.

The results indicate that we have succeeded in improving the document layout, compared to G_Invoices the Alignment score. The Overlap score is slightly worse but comparable to the baseline. This decrease in Overlap diversity is an expected consequence of template reuse; by design, our generator

[3]Code of the tool is hosted on GitHub: https://github.com/NewDocGen/NewDocGen/

TABLE 4. Statistics of three diversity metrics samples, sample size $n = 10^3$ (data aggregated from 10^6 randomly generated documents).

Metric	$\tilde{x}$	$x_{(1)}$	$x_{(n)}$	W_n	MSE	MAD
Alignment	0.19327	0.18888	0.19677	0.00789	1.425×10^{-6}	0.00080
Overlap	0.90625	0.90267	0.90917	0.00650	1.255×10^{-6}	0.00075
SELF-BLEU	0.22106	0.20844	0.23188	0.02344	1.191×10^{-5}	0.00229

TABLE 5. Confidence intervals for the mean value of the metrics (confidence level - 95%).

Metric	Confidence interval
Alignment	(0.1931794, 0.1933276)
Overlap	(0.9061674, 0.9063065)
SELF-BLEU	(0.2208066, 0.2212352)

balances variety with realism, which sometimes means reusing layouts and hence slightly less spatial variation. As for the content diversity score, we are getting significant improvements for the 1000 documents baseline.

Also, we explored the stability of the generated data, repeating the experiment 1000 times. Collected data have been analyzed statistically. First, sets of all three metrics[4] (Alignment, Overlap, and SELF-BLEU) have been testing for the normality, using Lemeshko's methodology for composite hypotheses (see [26], [27]). The results are presented in Table 3. Note that in this table $\hat{\Theta} = (\hat{\mu}, \hat{\sigma})$ stands for the *ML*-estimated parameters of the Gaussian distribution, $S_K = \sqrt{n}D_n + (6\sqrt{n})^{-1}$. D_P and C_P stand for the critical values of the general and Gaussian-specific Lemeshko's tests respectively.

As we can see from the table's data, the hypothesis of normality can not be rejected. This fact is visualized in Figure 2. Indeed, we can barely discern the empirical distribution functions (green curves) from the hypothetical (normal) distribution functions (black curves).

Assuming the normality of the data, we proceed with further analysis, calculating the median $\tilde{x}$, *min* and *max* order statistics ($x_{(1)}$ and $x_{(n)}$ respectively), the range W_n, the mean squared error (MSE) and the median absolute deviation (MAD). The results, presented in Table 4, clearly exhibit the robustness of our scoring. For the Alignment score we can see that even repeating the modeling 10^3 times (with 10^6 simulated documents) we are consistently getting better results, i.e., our worst value is better than G_Invoices one, $x_{(1)} = 0.18888 > 0.14$. The same is true for the SELF-BLEU metrics, $x_{(n)} = 0.23188 < 0.29$. These statistical experiments testify that the proposed method consistently gives better results compared to older ones.

Next, we check confidence intervals for these metrics. Table 5 presents calculated confidence intervals for the Alignment, Overlap, and SELF-BLEU metrics. Once again, one can see that we have made definite improvements in the Alignment and SELF-BLEU scores. The lengths of the confidence intervals are small (1.482×10^{-4}, 1.391×10^{-4} and 4.286×10^{-4} respectively), demonstrating once more the robustness of the modeling.

[4]Samples are hosted on GitHub: https://github.com/NewDocGen/NewDocGen/

V. CONCLUSION AND DISCUSSION

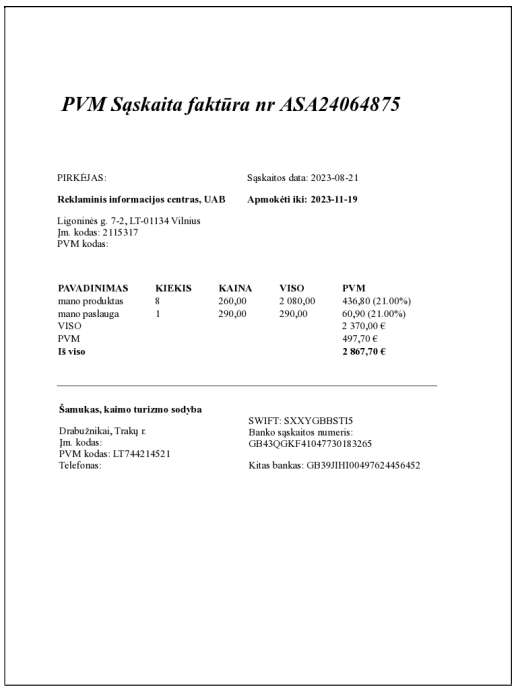
This paper presents the invoice generation tool, implementing a two-stage (layout generation - data generation) approach. Testing the tool experimentally, we have generated a new synthetic dataset, *NewDocGen*, and compared it, using three metrics, to the real-world SROIE English language corpus and G_Invoices synthetic dataset generated by an older tool. The experiment results show that our approach allows us to achieve noticeable improvement in the layout diversity, measured by the normalized Alignment score, and content diversity, measured by the SELF-BLEU score.

Statistically validating the stability of the generation, we have assured consistently better results over the foregoing ones. We have tested the normality of the metric scores using Lemeshko's methodology, getting that the hypothesis of normality can not be rejected. Further statistical analysis showed that even our worst results for the improved metrics are better than the G_Invoices one. Confidence interval lengths for the mean value of the metrics are small. It leads to the conclusion that we have statistically significant stability of dataset generation and that stability allows us to assert that our results are remarkably better.

We were unable to achieve baseline layout diversity measured by Overlap score. In future work, increasing the number of templates for layout generation may help to improve this score. Further synthetic invoice corpus evaluation using metrics that are more relevant to machine learning would be very beneficial. Next, existing metrics are not adequate enough to evaluate dataset suitability in machine learning applications. More baseline corpora would help to have a better understanding of the variety of diversity in different document datasets.

A significant improvement would be the ability to learn layout templates from existing documents and add variability to them. Note that there is a certain progress in this direction, cf. [13], [14], [15], [16].

A related problem is the augmentation of synthetic document images by applying distortions and noise using methods similar to those proposed in [28] and [29]. Obtained documents could be used for machine learning models working directly with images. Models under research could



PVM Sąskaita faktūra nr ASA24064875

PIRKĖJAS:

Reklaminis informacijos centras, UAB

Ligoninės g. 7-2, LT-01134 Vilnius
Jm. kodas: 2115317
PVM kodas:

Sąskaitos data: 2023-08-21

Apmokėti iki: 2023-11-19

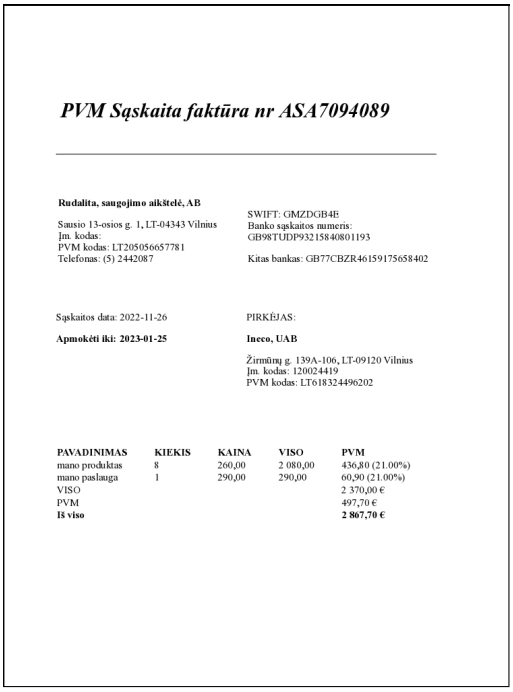
PAVADINIMAS	KIEKIS	KAINA	VISO	PVM
mano produktas	8	260,00	2 080,00	436,80 (21.00%)
mano paslauga	1	290,00	290,00	60,90 (21.00%)
VISO				2 370,00 €
PVM				497,70 €
Iš viso				**2 867,70 €**

Šamukas, kaimo turizmo sodyba

Drabužnikai, Trakų r.
Jm. kodas:
PVM kodas: LT744214521
Telefonas:

SWIFT: SXXYGBBSTI5
Banko sąskaitos numeris:
GB43QGKF41047730183265

Kitas bankas: GB39IIHI00497624456452



PVM Sąskaita faktūra nr ASA7094089

Rudalita, saugojimo aikštelė, AB

Sausio 13-osios g. 1, LT-04343 Vilnius
Jm. kodas:
PVM kodas: LT205056657781
Telefonas: (5) 2442087

SWIFT: GMZDGB4E
Banko sąskaitos numeris:
GB98TUDP93215840801193

Kitas bankas: GB77CBZR46159175658402

Sąskaitos data: 2022-11-26

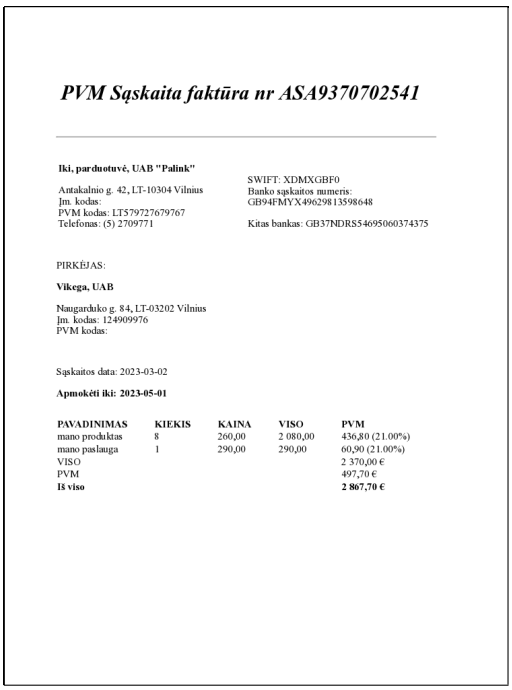
Apmokėti iki: 2023-01-25

PIRKĖJAS:

Ineco, UAB

Žirmūnų g. 139A-106, LT-09120 Vilnius
Jm. kodas: 120024419
PVM kodas: LT618324496202

PAVADINIMAS	KIEKIS	KAINA	VISO	PVM
mano produktas	8	260,00	2 080,00	436,80 (21.00%)
mano paslauga	1	290,00	290,00	60,90 (21.00%)
VISO				2 370,00 €
PVM				497,70 €
Iš viso				**2 867,70 €**



PVM Sąskaita faktūra nr ASA9370702541

Iki, parduotuvė, UAB "Palink"

Antakalnio g. 42, LT-10304 Vilnius
Jm. kodas:
PVM kodas: LT579727679767
Telefonas: (5) 2709771

SWIFT: XDMXGBF0
Banko sąskaitos numeris:
GB94FMYX49629813598648

Kitas bankas: GB37NDRS54695060374375

PIRKĖJAS:

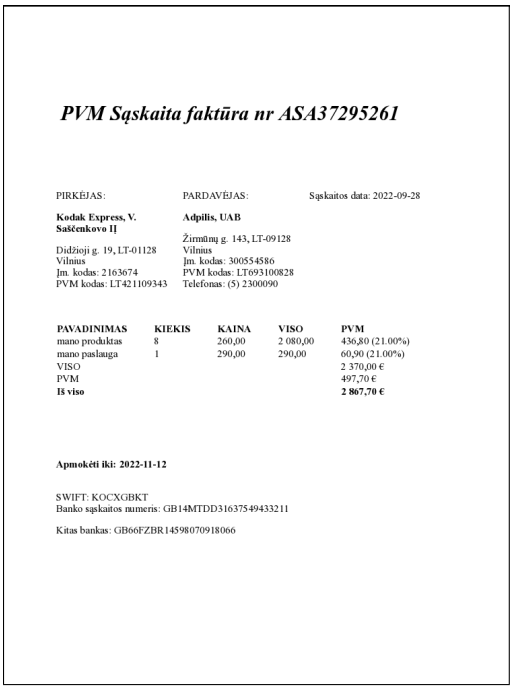
Vikega, UAB

Naugarduko g. 84, LT-03202 Vilnius
Jm. kodas: 124909976
PVM kodas:

Sąskaitos data: 2023-03-02

Apmokėti iki: 2023-05-01

PAVADINIMAS	KIEKIS	KAINA	VISO	PVM
mano produktas	8	260,00	2 080,00	436,80 (21.00%)
mano paslauga	1	290,00	290,00	60,90 (21.00%)
VISO				2 370,00 €
PVM				497,70 €
Iš viso				**2 867,70 €**



PVM Sąskaita faktūra nr ASA37295261

PIRKĖJAS:

Kodak Express, V. Saščenkovo IĮ

Didžioji g. 19, LT-01128
Vilnius
Jm. kodas: 2163674
PVM kodas: LT421109343

PARDAVĖJAS:

Adpilis, UAB

Žirmūnų g. 143, LT-09128
Vilnius
Jm. kodas: 300554586
PVM kodas: LT693100828
Telefonas: (5) 2300090

Sąskaitos data: 2022-09-28

PAVADINIMAS	KIEKIS	KAINA	VISO	PVM
mano produktas	8	260,00	2 080,00	436,80 (21.00%)
mano paslauga	1	290,00	290,00	60,90 (21.00%)
VISO				2 370,00 €
PVM				497,70 €
Iš viso				**2 867,70 €**

Apmokėti iki: 2022-11-12

SWIFT: KOCXGBKT
Banko sąskaitos numeris: GB14MTDD31637549433211

Kitas bankas: GB66FZBR14598070918066

range from basic OCR to end-to-end document processing. Note that these approaches differ from the approaches to document text and layout processing we are developing in the current study.

In this research, we have focused on synthetic invoices. However uncomplicated template changes could be used to synthesize other classes of financial and non-financial documents, e.g. payslips, orders, or receipts.

The introduced synthetic document generation software *NewDocGen* will be very useful as a tool to generate datasets for less-represented languages for the training of machine learning models for financial document processing automation. We repeated the random document generation experiment 1000 times and statistically validated the robustness of the results. *NewDocGen* software has the significant property of producing consistent and reproducible generation results (as has been shown by all metrics used).

The application of the *NewDocGen* allows us to generate large amounts of documents necessary for the successful application of machine learning techniques. The most promising approach lies in using the generated data for the pre-training of machine learning models with subsequent fine-tuning on small specific document sets [30], [31].

APPENDIX A
EXAMPLES OF SYNTHETIC NEWDOCGEN INVOICES

The images of synthetic newdocgen invoices are given at the top of the previous page.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for carefully reading the manuscript and providing constructive comments and suggestions, which have helped improve the article's quality.

REFERENCES

[1] H. Zhang, Q. Zheng, B. Dong, and B. Feng, "A financial ticket image intelligent recognition system based on deep learning," *Knowl.-Based Syst.*, vol. 222, Jun. 2021, Art. no. 106955.
[2] H. Zhang, B. Dong, Q. Zheng, and B. Feng, "Research on fast text recognition method for financial ticket image," *Appl. Intell.*, vol. 52, no. 15, pp. 18156–18166, Dec. 2022.
[3] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, and V. K. Asari, "A state-of-the-art survey on deep learning theory and architectures," *Electronics*, vol. 8, no. 3, p. 292, Mar. 2019.
[4] Z. Huang, K. Chen, J. He, X. Bai, D. Karatzas, S. Lu, and C. V. Jawahar, "ICDAR2019 competition on scanned receipt OCR and information extraction," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 1516–1520.
[5] A. W. Harley, A. Ufkes, and K. G. Derpanis, "Evaluation of deep convolutional nets for document image classification and retrieval," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2015, pp. 991–995.
[6] ZUGFeRD Community. (Sep. 2022). *ZUGFeRD Test Invoice Corpus*. [Online]. Available: https://github.com/ZUGFeRD/corpus
[7] J. Sánchez, A. Salgado, A. J. García, and N. Monzón, "IDSEM, an invoices database of the Spanish electricity market," *Sci. Data*, vol. 9, no. 1, p. 786, Dec. 2022.
[8] J. Blanchard, Y. Belaïd, and A. Belaïd, "Automatic generation of a custom corpora for invoice analysis and recognition," in *Proc. Int. Conf. Document Anal. Recognit. Workshops (ICDARW)*, vol. 7, Sep. 2019, p. 1.
[9] D. Belhadj, Y. Belad, and A. Belad, "Automatic generation of semi-structured documents," in *Document Analysis and Recognition—ICDAR* (Lecture Notes in Computer Science), E. H. Barney Smith and U. Pal, Eds., Cham, Switzerland: Springer, 2021, pp. 191–205.
[10] O. Bensch, M. C. Popa, and C. Spille, "Key information extraction from documents: Evaluation and generator," in *Proc. DeepOntoNLP/X-SENTIMENT@ESWC*, 2021.
[11] M. Schulze, M. Schröder, C. Jilek, and A. Dengel, "ptpDG: A purchase-to-pay dataset generator for evaluating knowledge-graph-based services," in *Proc. SEMWEB*, 2021. [Online]. Available: https://dblp.org/rec/conf/semweb/SchulzeSJ021.html?view=bibtex
[12] Provectus. (2021). *Synthetic Invoice Dataset Generator*. [Online]. Available: https://provectus.com/wp-content/uploads/2021/11/synthetic_compressed.pdf
[13] D. M. Arroyo, J. Postels, and F. Tombari, "Variational transformer networks for layout generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13637–13647.
[14] K. Gupta, J. Lazarow, A. Achille, L. Davis, V. Mahadevan, and A. Shrivastava, "LayoutTransformer: Layout generation and completion with self-attention," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 984–994.
[15] L. He, Y. Lu, J. Corring, D. Florêncio, and C. Zhang, "Diffusion-based document layout generation," in *Document Analysis and Recognition—ICDAR*, G. A. Fink, R. Jain, K. Kise, and R. Zanibbi, Eds., Cham, Switzerland: Springer, pp. 361–378.
[16] A. G. Patil, O. Ben-Eliezer, O. Perel, and H. Averbuch-Elor, "READ: Recursive autoencoders for document layout generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 2316–2325.
[17] J. Van Beusekom, D. Keysers, F. Shafait, and T. M. Breuel, "Distance measures for layout-based document image retrieval," in *Proc. 2nd Int. Conf. Document Image Anal. Libraries (DIAL)*, Apr. 2006, pp. 232–242.
[18] Y. Zhu, S. Lu, L. Zheng, J. Guo, W. Zhang, J. Wang, and Y. Yu, "Texygen: A benchmarking platform for text generation models," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.* New York, NY, USA: Association for Computing Machinery, Jun. 2018, pp. 1097–1100.
[19] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating text generation with BERT," in *Proc. Int. Conf. Learn. Represent.*, 2020. [Online]. Available: https://dblp.org/rec/conf/iclr/ZhangKWWA20.html?view=bibtex
[20] A. Groleau, K. W. Chee, S. Larson, S. Maini, and J. Boarman, "Augraphy: A data augmentation library for document images," in *Document Analysis and Recognition—ICDAR*, G. A. Fink, R. Jain, K. Kise, and R. Zanibbi, Eds., Cham, Switzerland: Springer, pp. 384–401.
[21] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, p. 125, Feb. 2020.
[22] Z. Papakipos and J. Bitton, "AugLy: Data augmentations for adversarial robustness," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 155–162.
[23] S. Laatiri, P. Ratnamogan, J. Tang, L. Lam, W. Vanhuffel, and F. Caspani, "Information redundancy and biases in public document information extraction benchmarks," in *Document Analysis and Recognition—ICDAR*, G. A. Fink, R. Jain, K. Kise, and R. Zanibbi, Eds., Cham, Switzerland: Springer, 2023, pp. 280–294.
[24] A. Kothare, S. Chaube, Y. Moharir, G. Bajodia, and S. Dongre, "SynGen: Synthetic data generation," in *Proc. Int. Conf. Comput. Intell. Comput. Appl. (ICCICA)*, Nov. 2021, pp. 1–4.
[25] X. Guo and Y. Chen, "Generative AI for synthetic data generation: Methods, challenges and the future," 2024, *arXiv:2403.04190*.
[26] B. Y. Lemeshko and S. B. Lemeshko, "Construction of statistic distribution models for nonparametric goodness-of-fit tests in testing composite hypotheses: The computer approach," *Qual. Technol. Quant. Manage.*, vol. 8, no. 4, pp. 359–373, Jan. 2011.
[27] B. Lemeshko, S. Lemeshko, and A. Rogozhnikov, "Real-time studying of statistic distributions of non-parametric goodness-of-fit tests when testing complex hypotheses," in *Proc. Int. Workshop 'Appl. Methods Stat. Anal. Simulations Stat. Inference'*, vol. 1, 2011, pp. 19–27.
[28] Q. A. Bui, D. Mollard, and S. Tabbone, "Automatic synthetic document image generation using generative adversarial networks: Application in mobile-captured document analysis," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 393–400.
[29] R. Karpinski and A. Belaïd, "Semi-synthetic data augmentation of scanned historical documents," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 268–273.
[30] R. Entezari, M. Wortsman, O. Saukh, M. M. Shariatnia, H. Sedghi, and L. Schmidt, "The role of pre-training data in transfer learning," in *Proc. ICLR Workshop Multimodal Represent. Learning, Perks Pitfalls*, 2023. [Online]. Available: https://dblp.org/rec/journals/corr/abs-2302-13602.html?view=bibtex
[31] A. Ramé, J. Zhang, L. Bottou, and D. Lopez-Paz, "Pre-train, fine-tune, interpolate: A three-stage strategy for domain generalization," in *Proc. 1st Workshop Interpolation Regularizers Beyond NeurIPS*, 2022. [Online]. Available: https://openreview.net/forum?id=47ypqHrYYZ

ROLANDAS GRICIUS received the degree in applied mathematics from Vilnius University, in 1994, where he is currently pursuing the Ph.D. degree in informatics.

He has been a Lecturer with the International School of Law and Business, Klaipeda University, and recently with Vilnius University. His research interests include information extraction and similar NLP topics.

IGORIS BELOVAS received the B.S. degree in applied mathematics and the M.S. degree in statistics from Vilnius University, in 1997 and 1999, respectively, and the joint Ph.D. degree in mathematics from Vilnius Gediminas Technical University and the Institute of Mathematics and Informatics, Vilnius, in 2004.

Since 1997, he has held positions with the Institute of Mathematics and Informatics (now the Institute of Data Science and Digital Technologies, subdivision Vilnius University Faculty of Mathematics and Informatics), now being a Professor and a Senior Researcher. Since 2003, he taught 25 courses with Vilnius University, Vilnius Gediminas Technical University, Mykolas Romeris University, Šiauliai University, and the International School of Law and Business. He participated in eight research projects (five times as a Principal Investigator). He is the author and the co-author of two books and over 50 articles. His research interests include the problems of mathematical modeling and number theory.

Dr. Belovas is a member of the Lithuanian Mathematical Society. He is an Editor of *Lietuvos Matematikos rinkinys* journal.