DOI: 10.15514/ISPRAS–2020–32(4)–14

Использование синтетических данных для тонкой настройки моделей сегментации документов

[1] *О.В. Беляева, ORCID: 0000-0002-6008-9671 <belyaeva@ispras.ru>*
[2] *А.И. Перминов, ORCID: 0000-0001-8047-0114 <perminov@ispras.ru>*
[1] *И.С. Козлов, ORCID: 0000-0002-0145-1159 <kozlov-ilya@ispras.ru>*

[1] *Институт системного программирования им. В.П. Иванникова РАН, 109004, Россия, г. Москва, ул. А. Солженицына, д. 25*
[2] *Московский государственный университет имени М.В. Ломоносова, 119991, Россия, Москва, Ленинские горы, д. 1*

Аннотация. В рамках задачи автоматического анализа документов мы решаем задачу сегментации изображений документов DLA (Document Layout Analysis). Целью работы является сегментация изображений документов в условиях ограниченного набора реальных данных и использование для обучения искусственно созданных данных. В качестве данных рассматривается PDF-документы сканированных договоров, коммерческих предложений и технических заданий без текстового слоя. В работе мы обучаем известную высокоуровневую модель FasterRCNN сегментировать текстовые блоки, таблицы, печати и подписи на изображениях рассматриваемых данных. Работа направлена на генерацию синтетических данных схожих с реальными. Это обусловлено потребностью модели в большом наборе данных для обучения и высокой трудозатратностью их подготовки. В работе приведено описание этапа постобработки для устранения артефактов, полученных в результате сегментации. В работе приводится тестирование и сравнение качества модели, обученной на разных наборах данных (с/без синтетических данных, малом/большом наборе реальных данных, с/без этапа постобработки). В итоге мы показываем, что генерация синтетических данных и использование постобработки увеличивает качество модели при малом обучающем наборе реальных данных.

Ключевые слова: анализ физической структуры документа; сегментация документа; анализ макета документа; обнаружение объектов на изображении; тонкая настройка модели; активное обучение

Для цитирования: Беляева О.В., Перминов А.И., Козлов И.С. Использование синтетических данных для тонкой настройки моделей сегментации документов. Труды ИСП РАН, том 32, вып. 4, 2020 г., стр. 189–202. DOI: 10.15514/ISPRAS–2020–32(4)–14

Synthetic data usage for document segmentation models fine-tuning

[1] *Belyaeva O.V., ORCID: 0000-0002-6008-9671 <belyaeva@ispras.ru>*
[2] *Perminov A.I., ORCID: 0000-0001-8047-0114 <perminov@ispras.ru>*
[1] *Kozlov I.S., ORCID: 0000-0002-0145-1159 <kozlov-ilya@ispras.ru>*

[1] *Ivannikov Institute for System Programming of the RAS, 25, Alexander Solzhenitsyn Str., Moscow, 109004, Russia*
[2] *Lomonosov Moscow State University, GSP-1, Leninskie Gory, Moscow, 119991, Russian Federation*

Abstract. In this paper, we propose an approach to the document images segmentation in a case of limited set of real data for training. The main idea of our approach is to use artificially created data for training and post-processing. The domain of the paper is PDF documents, such as scanned contracts, commercial proposals and technical specifications without a text layer is considered as data. As part of the task of automatic document analysis, we solve the problem of segmentation of DLA documents (Document Layout Analysis). In the paper we train the known high-level FasterRCNN \cite{ren2015faster} model to segment text blocks, tables, stamps and captions on images of the domain. The aim of the paper is to generate synthetic data similar to real data of the domain. It is necessary because the model needs a large dataset for training and the high labor intensity of their preparation. In the paper, we describe the post-processing stage to eliminate artifacts that are obtained as a result of the segmentation. We tested and compared the quality of a model trained on different datasets (with / without synthetic data, small / large set of real data, with / without post-processing stage). As a result, we show that the generation of synthetic data and the use of post-processing increase the quality of the model with a small real training data.

Keywords: Document Layout Analysis; Document Segmentation; Physical Document Structure; Image Object Detection; Model fine-tuning; Active Learning

For citation: Belyaeva O.V., Perminov A.I., Kozlov I.S. Synthetic data usage for document segmentation models fine-tuning. Trudy ISP RAN/Proc. ISP RAS, vol. 32, issue 4, 2020. pp. 189–202 (in Russian). DOI: 10.15514/ISPRAS–2020–32(4)–14

1. Введение

В настоящее время большое число документов представляют из себя файлы в формате PDF (Portable Document Format). Как правило, они удобны и просты в использовании. Но в частных случаях, встречаются PDF-документы, содержащие текстовый слой низкого качества, который был автоматически распознан. В худшем случае, документ представляет из себя скан бумажного носителя и не содержит вовсе текстового слоя. К таким документам, могут быть применимы только методы анализа изображения для извлечения его содержимого и структуры.

Выделяют *физическую* и *логическую* структуры документа. Первая задается такими классами объектов, как изображение, текст, таблицы, подписи и так далее. Во второй объекты разделяют на заголовки, параграфы и другие логические элементы.

Например, в статье [2] рассматривается первый вид структуры, который характеризуется геометрическим расположением объектов на странице.

Рис. 1. Пример сегментации документа [3]
Fig. 1. An example of document layout analysis [3]

Геометрическая организация документа: шрифт, размер текста, отступы содержат важную информацию, помогающую человеку понять смысл документа. Отсюда, важно сохранить такую информацию при автоматическом его анализе. Таким образом, одним из важных этапов автоматического анализа документа является сегментация страницы (Document

Layout Analysis, DLA, рис. 1). Задача сегментации изображения документа является частным случаем задачи сегментации изображения.

Сегментация страниц отсканированных документов проводится на первых этапах анализа и задает дальнейший характер распознавания документа в целом. В ходе DLA документ разбивается на области, каждая из которых должна содержать однородную информацию (например, только текст, заголовки или только одну таблицу).

Для сегментации изображений активно применяются методы на основе глубоких нейронных сетей, таких как [1]. Методы глубокого машинного обучения в рамках решения задачи DLA появились недавно [3-5] и уже нашли широкое применение для анализа сложных макетов документов. Методы сегментации на основе глубоких нейронных сетей демонстрируют высокое качество работы, но требуют огромного объёма данных для обучения, разметка которых требует высокой внимательности и много человеко-часов. Решением данной проблемы является автоматическая генерация данных. Например, в статье [3] используются данные с сайта [6] для автоматического получения обучающей выборки сегментации научных статей, в результате авторам удалось создать огромный обучающий датасет.

В нашей работе мы исследуем возможность применения глубокого обучения к задаче DLA на данных другой области: технических заданиях и юридических документах. Мы решаем задачу многоклассовой сегментации, которая комбинирует в себе задачи Object Detection и классификации.

В условиях ограниченного набора реальных данных, нами решается задача активного обучения [7]. Существует несколько подходов активного обучения с использованием отбора существующих данных [8] или синтеза новых данных [9]. Мы достигаем адаптации модели к новому множеству за счёт её тонкой настройки на новом наборе данных. Набор создается автоматически путем генерации искусственных документов близких к нашему домену. В рамках данной работы тонкая настройка представляет собой обучение весовых коэффициентов модели на новом наборе данных. Таким образом, тонкая настройка производится на искусственно сгенерированных документах, похожих на реальные. Далее предобученная модель дообучается на ограниченном наборе имеющихся реальных данных. Таким образом, нам удалось достичь высокого качества сегментации документов в ограниченном наборе реальных данных.

Мы использовали сложную сеть Faster RCNN [1] для сегментации документа в рамках задачи Object Detection с выделением нескольких классов на изображении: текстовый блок (Text), таблица (Table), картинка (Picture). Первоначальное обучение производится на датасете PubLayNet [3], мощностью 360 тысяч размеченных изображений медицинских статей.

Для улучшения точности сегментации мы используем постобработку для устранения артефактов, образованных в результате предсказания модели. Постпроцессинг представляет собой расширение или сужение границ обнаруженных объектов с целью избежать обрезки текста, таблиц; устранить наложения объектов друг на друга и т.д. Всё вышеперечисленное влияет на качество распознавания сегментируемых объектов в дальнейшем.

2. DLA решения

Развитие DLA берет начало с ранних известных эвристических методов Smearing [10], Recursive XY-cut [11], Docstrum [12], а также сегментация методом наибольших белых прямоугольников [13]. И в настоящее время активно проводятся исследования в области DLA с использованием машинного обучения. Выделяются два типа работ в этой области: бинаризации изображений и многоклассовая сегментация. Работы [4-5] сегментируют документы путем бинаризации изображений на None-Text и Text классы (текстовые/нетекстовые области). Как правило, это нужно для анализа исторических документов, на которых важно отфильтровать только текстовую информацию. В [4-5] предлагают использование свёрточной сети для сегментации страниц исторических

документов. Авторы [4] используют простую сеть только из одного свёрточного слоя для распознавания исторических рукописных документов и сравнивают результаты с более глубокими сложными сетями. В [5] для сегментирования исторических документов предлагают Fully Convolution Network (FCN) сеть, использующую метрику, которая учитывает только пиксели переднего плана на бинаризованной странице и игнорирует фоновые пиксели.

Работы [1, 3, 14] занимаются многоклассовой сегментацией документов. В работе [15] авторы обнаруживают и распознают структуру таблиц на изображениях. В [3] создали огромный датасет на основе данных с сайта PubMed и обучили FasterRCNN на новых данных. В [14] решают задачу детекции таблиц и даграмм с помощью глубокой свёрточной сети с CRF (conditional random field) слоем.

Как показывает практика, нельзя выбрать одну конкретную сеть и сказать, что она лучше остальных. В зависимости от задачи и требований к производительности (скорости обработки изображений в секунду) и качеству сегментации, выбор может быть сделан в пользу любой из них. Ниже приведены основные характеристики данных архитектур, влияющие на их качество:

- детектор признаков (VGG16, ResNet, Inception, MobileNet, рис. 2);
- размер выхода детектора признаков;
- разрешения входного изображения;
- стратегия соответствия и порог IoU (Intersection over Union);
- количество предложений или прогнозов;
- увеличение данных путём аугментации;
- набор данных для обучения;
- какой слой или слои карты признаков используются для обнаружения объектов;
- функция потерь;
- конфигурации обучения, включая размер пакета, изменение размера входного изображения, скорость обучения и снижение скорости обучения (табл. 1).

Табл. 1. Скорость работы различных архитектур [16]
Table 1. Speed of different architectures [16]

Архитектура	**Минимальный FPS**	**Максимальный FPS**
Fast R-CNN	3	10
Faster R-CNN	5	17
SSD	22	59
YOLO	40	91

Чтобы сравнивать модели между собой, необходимо выбрать единый набор данных, на котором будет происходить сопоставление. Обычно для этого используется набор соревнования COCO [17], в котором выполняется сегментация по 80 различным классам. Ниже приводится сравнительная таблица качества сегментации архитектур (табл. 2).

Табл. 2. Точность сегментации различных архитектур (на наборе данных COCO)
Table 2. Accuracy of layout analysis n of various architectures (on the COCO dataset)

Архитектура	**Точность**
Fast R-CNN	21.9
Faster R-CNN	34.9
SSD300	23.2
SSD512	26.8
YOLO	33.0
RetinaNet	40.8

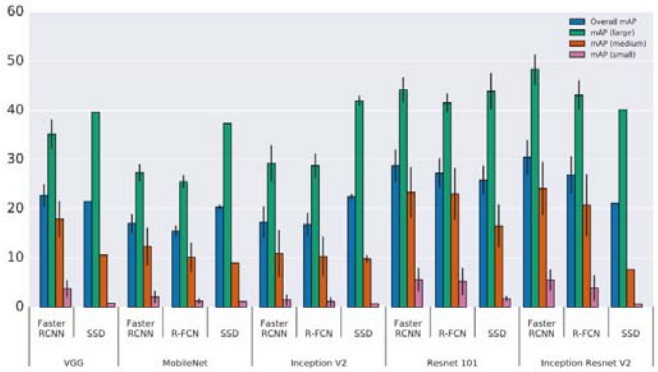


Рис. 2. Сравнение точности архитектур по детекторам признаков [16]
Fig. 2. Comparison of the accuracy of architectures by feature detectors [16]

На основе работ [1-3] в качестве модели был взят FasterRCNN с опорной сетью ResNet101. Согласно рис. 2, архитектура ResNet101 имеет высокую точность детекции больших объектов mAP^{large}, которых в наших данных большинство, FasterRCNN опережает большинство других моделей по точности сегментации на COCO данных (табл. 2).

Согласно работе [3], FasterRCNN достигает state-of-the-art качества обнаружения таблиц на изображениях.

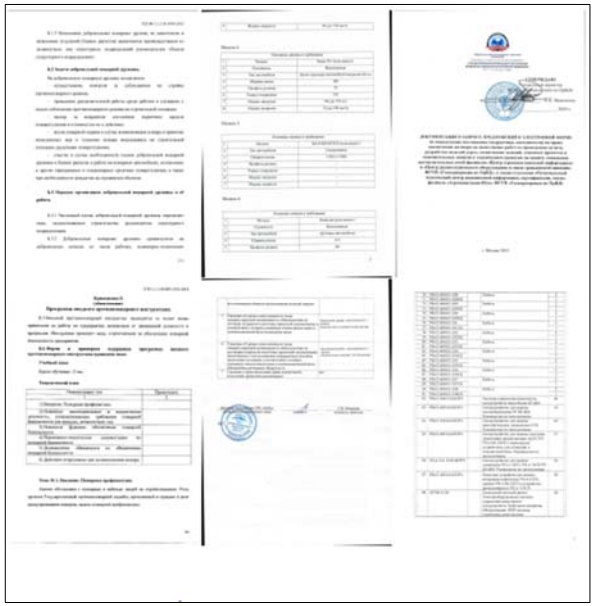


Рис. 3. Примеры изображений реальных документов
Fig. 3. Examples of images of real documents

3. Входные данные

Входные данные состоят из документов технических заданий и нормативно-правовых актов. Рассматриваемые документы находятся в открытом доступе и доступны по ресурсу [18]. Размеченный датасет изображений сканированных документов из [18] выложен и доступен по ссылке [19]. Сканированные документы, как правило, характеризуются высоким качеством символов, белым фоном, низким уровнем шума, манхэттенским стилем оформления, одноколоночностью. На рис. 3 представлены примеры входных данных.

3.1 Описание классов сегментирования

В рассматриваемых выше документах мы выделяем 3 класса объектов на изображениях:

- Text – текстовые блоки, содержащие однородный текст с единым форматированием (размером, жирностью, шрифтом, отступами между строк);
- Table – класс таблиц с границами, которые могут содержать объединенные ячейки по вертикали или по горизонтали. Заголовок таблиц может отличаться от тела другим форматированием;
- Picture – класс, содержащий печати, подписи, изображения в документах.

Выделение в отдельные классы элементов *Список*, *Заголовок* лучше производить на более низком уровне с использованием регулярных выражений и семантики.

4. Генерация документов

Для обучения модели сегментации необходимо иметь большой набор реальных данных, разметка которых, как было сказано ранее – процесс дорогостоящий. Поэтому было принято решение создать набор из искусственных данных путём самостоятельной генерации. Для этого реализован модуль генерации, позволяющий создавать случайные документы схожими визуально с реальными по текстовым шрифтам, межстрочным интервалам, жирности и стилям текста (рис. 4). Модуль предоставляет возможность извлечения ограничивающих прямоугольников для всех объектов извлекаемой логической структуры. Для создания документов выбрана стратегия случайного создания различных блоков и добавления их на страницу с последующим изменением форматирования. Сгенерированные блоки формируют единый docx документ, используемый впоследствии для создания PDF версии для упрощения извлечения изображений страниц.

Рис. 4. Алгоритм работы модуля генерации
Fig. 4. Algorithm of the generation module

4.1 Особенности создаваемых документов

Для максимальной близости генерируемых документов к предметной области модуль генерации использует следующие методы:

1. создаваемые документы одноколоночны;
2. используются различные семейства шрифтов;
3. для заголовков используются жирные шрифты и/или шрифты большего размера;
4. таблицы имеют все границы, а также объединённые ячейки;
5. создаются многостраничные таблицы с одинаковыми заголовками;

6. колонтитулы и нумерация страниц выравнивается случайным образом;
7. межстрочные интервалы и отступы имеют высокую вариативность.

4.2 Генерация текстовых блоков

Для генерации текстовой составляющей программный модуль содержит в себе текстовые фрагменты в виде текстовых файлов, разбитых по количеству слов в предложении, а также по типу блока – в обычном тексте, в списке или таблице. Для получения очередного предложения модуль случайным образом выбирает файл и строку в нём, а затем добавляет её содержимое в создаваемый документ.

4.3 Генерация списков

Алгоритм генерации списков очень схож с созданием обычных текстовых блоков, однако для списков ещё выбираются тип списка (маркированный или нумерованный) и количество пунктов. В качестве содержимого элемента списка выбираются строки файла, содержащего элементы различных списков.

4.4 Генерация таблиц

Модуль генерации выбирает случайным образом количество строк и столбцов и заполняет ячейки. Текстовое содержимое ячеек извлекается из файлов с текстовой информацией ячеек таблиц реальных данных. После заполнения ячеек содержимым, алгоритм выбирает случайное количество ячеек и тип объединения (горизонтальное или вертикальное) и объединяет их. Для некоторых ячеек случайным образом изменяется направление текста и его форматирование.

4.5 Генерация других блоков

Помимо обычного текста, списка и таблиц, модуль генерации добавляет в документы верхний и/или нижний колонтитул, а также нумерацию страниц в случайном месте (слева/по центру/справа, снизу/сверху).

4.6 Получение координат и классов блоков сегментации

Из-за отсутствия в docx информации о местоположении объектов на странице документов прямое получение разметки невозможно. Поэтому для каждого типа блока выбираются уникальные контрастные цвета, которыми они заливаются во время генерации.

По завершении генерации документ экспортируется в PDF формат. Затем заливка блоков удаляется и документ также экспортируется в PDF. Очищенные документы конвертируются в набор изображений страниц и используются в качестве входных обучающих данных для моделей сегментации.

Для получения разметки необходимо выделить контуры созданных блоков и получить их координаты. Для этого модуль извлечения разметки выполняет следующие действия:

- получает изображение с залитыми блоками;
- применяет маски, выполняющие фильтрацию изображения для определённого диапазона цветов;
- выделяет контуры с полученных изображений;
- находит пустое место средствами OpenCV [20] и уменьшает контуры с последующей нормализацией координат;
- добавляет полученные границы в список разметки.

После обработки всех изображений разметка сохраняется в формате COCO [17].

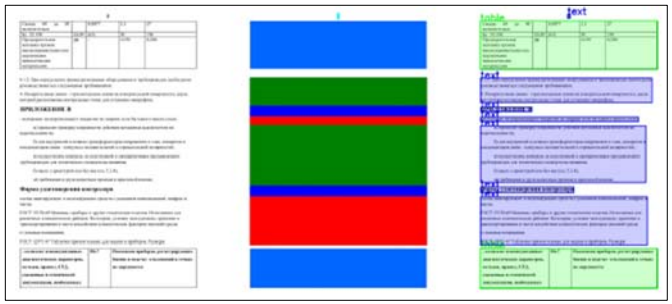


Рис. 5. Примеры сгенерированной страницы, её залитой версии и извлечённые контуры разметки
Fig. 5. Examples of the generated page, its flooded version and the extracted markup contours

5. Обучение модели

Мы обучали модель FasterRCNN [1] с помощью фреймворка TensorFlow Object Detection [21]. Веса и конфигурационные файлы выложены по ресурсу [19]. Тренировочный процесс состоял из трех этапов.

- Модель с претренированными весами на COCO датасете [17] тренировалась на 125 тысячах данных PubLayNet в течение 20 эпох [22];
- Модель доучивалась 4 эпохи на данных мощностью в 18 тысяч изображений, полученных с помощью модуля генерации;
- Далее модель обучалась на множестве реальных данных. В рамках эксперимента мы обучали на малом и большем наборах реальных данных (мощностью в 100 и 500 экземплярах соответственно).

Обучение модели на каждом этапе производилось со следующими параметрами:

- входные изображения формата A4 приводились к размеру 800 на 600 пикселей;
- значения learning rate устанавливались 0.001;0.0001;0.00001 начиная с 0, 1, 900 000-го шага соответственно;
- использовался SGD (Stochastic Gradient Descent) алгоритм с коэффициентом момента 0.9 и градиентным отсечением (gradient clipping) 10.0.

6. Постобработка

В результате предсказания обученной модели и невнимательной разметки данных могут образовываться артефакты, негативно влияющие на качество дальнейшего распознавания каждого сегментируемого блока. В качестве артефактов можно выделить следующее:

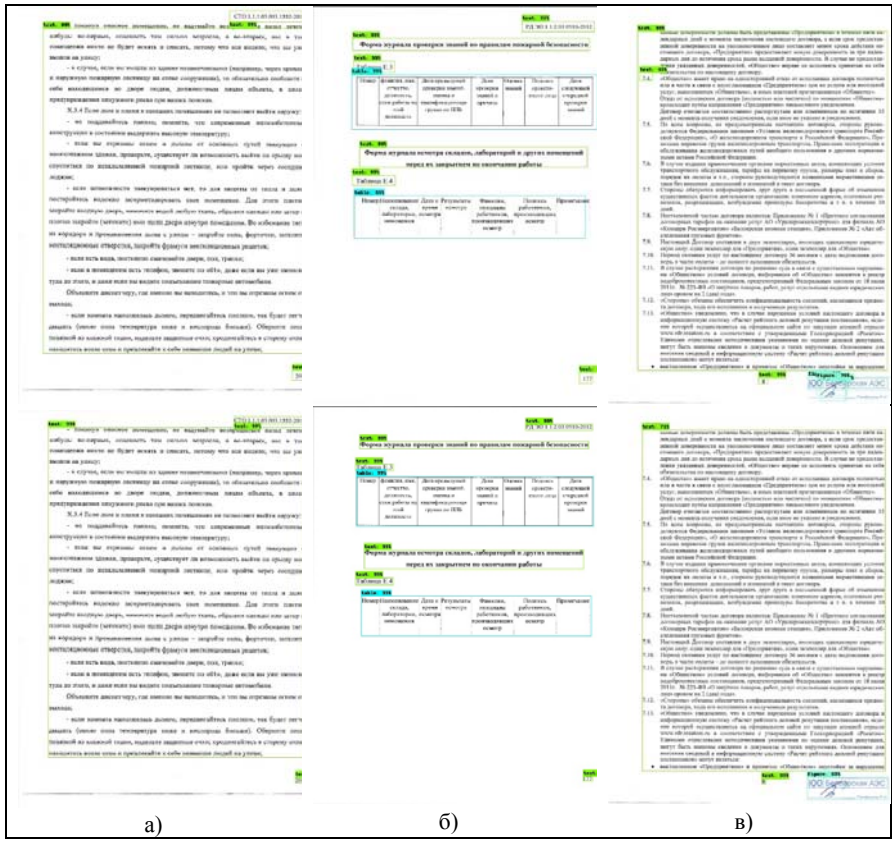
- обрезка текстового блока;
- обрезка таблиц;
- наложение блоков друг на друга (приводит к дублированию информации);
- избыточный захват фоновых пикселей по границам выделенного блока.

Для устранения артефактов сегментирования в пайплайн был включен модуль постобработки, включающий следующие этапы.

- На первом этапе постобработки, мы используем изменённый алгоритм None-Maximum-Supression (NMS) [23]. Мы хотим максимально сохранять текстовую информацию на изображениях, поскольку ее потеря критична для распознавания структуры документа в целом. С этой целью мы изменили классический подход NMS. Мы объединяем дубли с меньшей уверенностью для каждого объекта в один большой, вместо их удаления. В качестве дублей выступают объекты одного класса, пересечение которых Intersection over Union (IoU) превышает задаваемого порога $IoU > \delta$. Мы объединяем дубли,

пересечение которых IoU превосходит $\delta = 0.4$, в один объект и вычисляем score для нового объекта как средневзвешенное между старыми дублями.

- На втором этапе, изображение бинаризируется и инвертируется.
- На третьем этапе мы расширяем границы сегментируемых объектов на предварительно бинаризованном изображении, анализируя объекты от большей уверенности к меньшей. Объекты расширяются до тех пор, пока не пересекут соседние объекты или не превысят порога расширения. Порог расширения по горизонтали/вертикали задается как (ширина изображения / 30) и (высота изображения / 30) соответственно. Данный этап является подготовительным для следующего.
- На четвертом этапе границы сегментируемых объектов сужаются (уменьшается размер объектов). Сужение каждой границы к центру объекта проводится до пересечении с текстовым содержимым, со значениями пикселей > 0.
- На последнем этапе, устраняются пересекающиеся текстовые объекты, которые не были объединены на первом этапе.



а) б) в)

Рис. 6. Результаты работы постобработки. В первой строке расположены результаты предсказания модели до постобработки, во второй с постобработкой

Fig. 6. Results of post-processing work. The first line contains the results of model prediction before post-processing, the second with post-processing

Результаты работы постобработки представлены на рис. 6. На изображении (а) видно, без применения постобработки теряется верхняя строка основного текстового блока, на точности сегментации это бы не сильно отразилось, в отличии от точности распознавания текстовой информации на странице. На изображении (б) сегментатор обрезает часть второй таблицы без использования постобработки. На изображении (в) сегментатор дополнительно выделяет блок с текстом «7.4», и 2 раза подпись снизу. При применении 1-го шага постобработки, мы устранили бы дублирующую информацию. Вдобавок постобработка устраняет избыточный захват фоновых пикселей на краях выделенных объектов.

7. Результаты

Был подготовлен тестовый набор реальных данных мощностью 278 изображений, на котором проводились замеры качества моделей: **PLN** – набор данных научных статей PupLayNet, 125 тысяч изображений [22], **GEN** – набор сгенерированных данных, 18 тысяч изображений, **NPA-small** – набор реальных данных, 100 изображений, **NPA-big** – набор реальных данных 500 изображений [19].

Табл. 3. Количество объектов каждого класса в разных наборах данных
Table 3. The number of objects of each class in different datasets

	Текст	**Таблица**	**Картинка**	**Количество изображений**
GEN	104514	12413	0	18 000
NPA-small	474	53	31	100
NPA-big	2146	196	96	500
NPA test	1379	74	6	278

В табл. 4, 6 представлены результаты замеров качества модели FasterRCNN, обученной на разных наборах данных. В качестве метрик оценки качества использовались average precision (AP) $IoU = 0.5$ (PASCAL VOC метрика [24]) и mean average recall (MAR) со значениями порогов IoU [0.5; 0.95] с учетом large объектов, вычисленные с помощью coco api [17].

Табл. 4. Результаты сегментации (F_1-мера)
Table 4, Results of layout analysis

	Текст	**Таблица**	**Картинка**	**Итого**
PLN+GEN+NPA-big	**0.810**	**0.937**	**0.696**	**0.820**
PLN+GEN+NPA-small	0.502	0.824	0.336	0.559
PLN+NPA-small	0.489	0.846	0.346	0.565
PLN	0.045	0.065	0.004	0.039

В табл. 5, 7 отражена точность сегментирования с этапом постобработки. Измерения качества проводились на тестовом наборе с постобработкой.

Табл. 5. Результаты сегментации (F_1-мера) с постобработкой
Table 5, Results of layout analysis with post-processing

	Текст	**Таблица**	**Картинка**	**Итого**
PLN+GEN+NPA-big	0.810	0.937	0.696	0.820
PLN+GEN+NPA-big + Post	**0.840**	**0.968**	**0.755**	**0.855**
PLN+GEN+NPA-small	0.502	0.824	0.336	0.559
PLN+GEN+NPA-small + Post	0.652	0.888	0.448	0.663

Из табл. 4 видно, что генерация данных позволила повысить качество распознавания только класса «Текст» (строка 2) на 0.04, с применением постобработки результат заметно лучше – изменение на 0.15 (табл. 5). Качество сегментации класса «Картинка» не улучшилось по причине его отсутствия в генерированных данных GEN. Качество сегментации класса «Таблица» ухудшилось по причине переобучения на генерированных данных (в наборе данных GEN 12413 таблиц, в NPA-small – 53 таблицы) согласно табл. 3. Дообучение на большем наборе реальных данных из 500 изображений позволило достигнуть существенного увеличения качества по всем классам (строка 1) табл. 4. Модель, дообученная на малом наборе данных, показывает высокие результаты только с постобработкой согласно строкам 3-4 табл. 5. Удовлетворительное качество класса «Картинка» во 2-3 строках табл. 4 обусловлено его отсутствием в сгенерированном наборе данных GEN и малым количеством в наборе NPA-small.

В результате выбор синтетических данных оправдался для класса «Текст». Сегментатор переобучился на синтетических данных класса «Таблица», по этой причине точность сегментации этого класса возрасла только на наборе NPA-big.

Таким образом, можно достигнуть высоких результатов сегментирования на новых наборах данных NPA-small, NPA-big в условиях их ограниченности с использованием генерации дополнительных данных и применения методов постобработки.

Табл. 6. Результаты сегментации (точность / полнота)
Table 6. Results of layout analysis (precision / recall)

	Текст	**Таблица**	**Картинка**	**Итого**
PLN+GEN+NPA-big	0.941 0.711	0.968 0.907	0.899 0.568	0.936 0.729
PLN+GEN+NPA-small	0.609 0.427	0.915 0.749	0.509 0.251	0.678 0.476
PLN+NPA-small	0.570 0.428	0.926 0.778	0.502 0.264	0.666 0.490
PLN	0.029 0.099	0.053 0.085	0.002 0.017	0.028 0.067

Табл. 7. Результаты сегментации (точность / полнота) с постобработкой
Table 7. Results of layout analysis (precision / recall) with post-processing

	Текст	**Таблица**	**Картинка**	**Итого**
PLN+GEN+NPA-big	0.941 0.711	0.968 0.907	0.899 0.568	0.936 0.729
PLN+GEN+NPA-big + Post	0.886 0.798	0.968 0.969	0.811 0.706	0.888 0.824
PLN+GEN+NPA-small	0.609 0.427	0.915 0.749	0.509 0.251	0.678 0.476
PLN+GEN+NPA-small + Post	0.708 0.604	0.912 0.866	0.490 0.412	0.678 0.627

8. Заключение

В работе была выбрана модель сегментации и проведена ее тонкая настройка в условиях ограниченного объёма реальных данных. Несмотря на ограниченность набора данных, нам удалось достичь высоких результатов сегментации с постобработкой в 0.663 и 0.855 f-меры для 100 и 500 тренировочных экземпляров соответственно. Это достигается за счет генерации

искусственных данных с распределением схожим с реальными и постобработкой для устранения артефактов после сегментации изображения. Обученные веса модели и наборы данных выложены и публично доступны [19].

Исходя из полученных результатов, можно утверждать, что использование синтетических данных для дообучения моделей сегментации на малом количестве реальных данных и использование методов постобработки позволяет достичь высокой точности и не требует ручной разметки большого количества данных.

Список литературы / References

[1] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, issue 6, 2017, pp. 1137-1149.

[2] G.M. Binmakhashen and S.A. Mahmoud. Document layout analysis: a comprehensive survey. ACM Computing Surveys (CSUR), vol. 52, issue 6, 2019, pp. 1-36.

[3] X. Zhong, J. Tang, and A.J. Yepes. Publaynet: largest dataset ever for document layout analysis. arXiv:1908.07836, 2019.

[4] K. Chen, M. Seuret, J. Hennebert, and R. Ingold. Convolutional neural networks for page segmentation of historical document images. In Proc. of the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, 2017, pp. 965-970

[5] C. Wick and F. Puppe. Fully convolutional neural networks for page segmentation of historical document images. In Proc. of the 13th IAPR International Workshop on Document Analysis Systems (DAS), 2018, pp. 287-292.

[6] Pubmed. national library of medicine. URL: https://pubmed.ncbi.nlm.nih.gov. Accessed: 2020-21-07.

[7] G. Csurka. Domain adaptation for visual applications: a comprehensive survey. arXiv:1702.05374, 2017.

[8] М.А. Рындин, Д.Ю Турдаков. Проактивная разметка примеров для адаптации к домену. Труды ИСП РАН, том 31, вып. 5, 2019 г., стр. 145-152. DOI: 10.15514/ISPRAS-2019-31(5)-11 / M.A. Ryndin, D.Y. Turdakov. Domain adaptation by proactive labeling. Trudy ISP RAN/Proc. ISP RAS, vol.31, issue 5, 2019, pp. 145-152 (in Russian).

[9] C. R. De Souza, A. Gaidon, Y. Cabon, and A. M. López. Procedural Generation of Videos to Train Deep Action Recognition Networks. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2594-2604.

[10] L. Angeline, K. Teo, and F. Wong. Smearing algorithm for vehicle parking management system. In Proc. of the 2nd Seminar on Engineering and Information Technology, 2009, pp. 331-337.

[11] J. Ha, R. M. Haralick, and I. T. Phillips. Recursive xy cut using bounding boxes of connected components. In Proc. of the 3rd International Conference on Document Analysis and Recognition, vol. 2, 1995, pp. 952—955.

[12] L. O'Gorman. The document spectrum for page layout analysis. IEEE Transactions on pattern analysis and machine intelligence, vol. 15, issue 11, 1993, pp. 1162-1173.

[13] T.M. Breuel. An algorithm for finding maximal whitespace rectangles at arbitrary orientations for document layout analysis. In Proc. of the Seventh International Conference on Document Analysis and Recognition, 2003, pp. 66—70.

[14] I. Kavasidis, C. Pino, S. Palazzo, F. Rundo, D. Giordano, P. Messina, and C. Spampinato. A saliency-based convolutional neural network for table and chart detection in digitized documents. Lecture Notes in Computer Science, vol. 11752, 2019, pp. 292—302.

[15] S. Schreiber, S. Agne, I. Wolf, A. Dengel, and S. Ahmed. Deepdesrt: deep learning for detection and structure recognition of tables in document images. In Proc. of the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, 2017, pp. 1162—1167.

[16] Object detection: speed and accuracy comparison (faster r-cnn, r-fcn, ssd, fpn, retinanet and yolov3). URL: https://medium.com/@jonathan_hui/object-detectionspeed-and-accuracy-comparison-faster-r-cnn-r-fcnssd-and-yolo-5425656ae359. Accessed: 2020-18-07.

[17] Coco, common objects in context. URL: https://cocodataset.org/#home. Accessed: 2020-18-07.

[18] Единая информационная система в сфере закупок, ЕИС. URL: https://zakupki.gov.ru/ (дата обращения 27.05.2020) / Unified information system in the field of procurement, EIS. URL: https://zakupki.gov.ru/ (in Russian).

[19] Dla-dataset. EIS. URL: https://disk.yandex.ru/d/XVjQf20BVsElKA (accessed: 2020-18-07).

[20] Open source computer vision library. URL: https://opencv.org (accessed: 2020-18-07).
[21] Tensorflow object detection api. URL: https://github.com/tensorflow/models/tree/master/research/object_detection. Accessed: 2020-18-07.
[22] Publaynet dataset. URL: https://github.com/ibm-aurnlp/PubLayNet/tree/master/pre-trained-models. Accessed: 2020-18-07.
[23] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis. Soft-nms−improving object detection with one line of code. In Proc. of the IEEE International Conference on Computer Vision, 2017, pp. 5561-5569.
[24] Pascal voc evaluation. URL: http://host.robots.ox.ac.uk/pascal/VOC/voc2012/htmldoc/devkit_doc.html#SECTION00064000000000000000 (accesed: 08.09.2020).

Информация об авторах / Information about authors

Оксана Владимировна БЕЛЯЕВА – аспирантка. Научные интересы: распознавание структуры документов, цифровая обработка изображений, нейросетевая обработка данных, распознавание образов.

Oksana Vladimirovna BELYAEVA – a PhD Student. Research interests: document layout analysis, digital image processing, neural network data processing, image pattern recognition.

Андрей Игоревич ПЕРМИНОВ является студентом магистратуры кафедры системного программирования. Научные интересы: цифровая обработка сигналов, нейросетевая обработка данных, создание искусственных данных, цифровая обработка изображений.

Andrey Igorevich PERMINOV – master student of the Department of System Programming. Research interests: digital signal processing, neural network data processing, generation of artificial data, digital image processing.

Илья Сергеевич КОЗЛОВ является стажером-исследователем. Научные интересы: распознавание структуры документов, цифровая обработка изображений, нейросетевая обработка данных, распознавание образов.

Ilya Sergeevich KOZLOV – researcher at ISP RAN. Research interests: document layout analysis, digital image processing, neural network data processing, image pattern recognition.