# Building an Efficient OCR System for Historical Documents with Little Training Data

**Jiří Martínek**[1] · **Ladislav Lenc**[2] · **Pavel Král**[1,2]

Accepted for Neural Computing and Applications journal

**Abstract** As the number of digitized historical documents has increased rapidly during the last a few decades, it is necessary to provide efficient methods of information retrieval and knowledge extraction to make the data accessible. Such methods are dependent on optical character recognition (OCR) which converts the document images into textual representations. Nowadays OCR methods are often not adapted to the historical domain, moreover, they usually need a significant amount of annotated documents.

Therefore, this paper introduces a set of methods that allows performing an OCR on historical document images using only a small amount of real, manually annotated training data. The presented complete OCR system includes two main tasks: page layout analysis including text block and line segmentation and OCR. Our segmentation methods are based on fully convolutional networks and the OCR approach utilizes recurrent neural networks. Both approaches are state of the art in the relevant fields.

We have created a novel real dataset for OCR from Porta fontium portal. This corpus is freely available for research and all proposed methods are evaluated on these data. We show that both the segmentation and OCR tasks are feasible with only a few annotated real data samples. The experiments aim at determining the best way how to achieve good performance with the given small set of data. We also demonstrate that obtained scores are comparable or even better than the scores of several state-of-the-art systems.

To sum up, this paper shows a way how to create an efficient OCR system for historical documents with a need for only a little annotated training data.

**Keywords** CNN · FCN · Historical Documents · LSTM · Neural Network · OCR · Porta fontium · Synthetic Data

[1] Dept. of Computer Science & Engineering
Faculty of Applied Sciences
University of West Bohemia
Plzeň, Czech Republic

[2] NTIS - New Technologies for the Information Society
Faculty of Applied Sciences
University of West Bohemia
Plzeň, Czech Republic
`{jimar,llenc,pkral}@kiv.zcu.cz`

## 1 Introduction

Digitization of historical documents is an important task for preserving our cultural heritage. During the last a few decades, the amount of digitized archival material has increased rapidly and therefore, an efficient method to convert these document images into a text form has become essential to allow information retrieval and knowledge extraction on such data. Novadays state-of-the-art methods are usually not adapted to the historical domain, moreover, they usually need a significant amount of annotated documents which is very expensive and time consuming to acquire.

Therefore, this paper introduces a set of methods to convert historical scans into their textual representation for efficient information retrieval based on a minimal number of manually annotated documents. This problem includes two main tasks: page layout analysis (including text block and line segmentation) and optical character recognition (OCR). We address all these issues and we propose several approaches to solve these tasks.

This research is realized in the frame of the *Modern Access to Historical Sources* project, presented through the Porta fontium portal[1]. One goal of this project is to enable an intelligent full-text access to the printed historical documents from the Czech-Bavarian border region. Accordingly, our original data sources are scanned texts from German historical newspapers printed with Fraktur from the second half of the $19^{th}$ century.

All proposed methods are evaluated and compared on the real data from the Porta fontium portal. We have also built a novel annotated corpus for historical OCR based on such data. This dataset is intended for evaluation of state-of-the-art OCR systems and is freely available for research purposes[2]. Based on the experiments, the reader can very quickly build his historical OCR system.

Traditional OCR approaches usually detect and segment words and single characters in the documents. The spaces between the isolated characters and words are sometimes not evident due to their particular form or by the noise in the images. This fact brings often OCR errors. With the development of deep learning (especially recurrent neural networks), the methods based on processing of whole text lines became dominant. Such methods benefit from the context that is available when processing the text line as a sequence of frames and are thus not afected by character segmentation errors. High computational power and solutions to the vanishing and exploding gradient issues [34] allowed the learning of deep architectures that are capable of accommodating such input data. A great benefit is also the connectionist temporal classification (CTC) loss [16] which can map the labels into specific image frames.

Traditional text segmentation methods used to be solved by simple computer vision algorithms. Nowadays, approaches based on deep neural networks outperform the traditional methods. The best results in the text segmentation field are obtained by fully convolutional networks (FCN) [17]. Therefore, our proposed segmentation is also based on this network type. The current trend in the OCR field is to utilize neural networks which process whole text lines using recurrent neural networks (RNN) [8] including convolutional neural networks (CNNs) for feature extraction [41]. A great benefit of these approaches is that the segmentation to characters is not necessary. Hence, our OCR method is also based on the combination of convolutional and recurrent neural networks. A natural ability of RNNs is to learn an implicit language model (LM) [38,25,44]. To be able to learn the LM, we must provide the network with a sufficient amount of meaningful text in the domain

[1] http://www.portafontium.eu/

[2] http://ocr-corpus.kiv.zcu.cz/

we work in. A frequently used way how to obtain the texts is generating synthetic data. Synthetic data should be created with respect to the LM. We will show the influence of different types of synthetic data in the experimental section.

The main contribution of this paper is thus as follows:

1. Proposing text region and text line segmentation approaches based on fully convolutional networks using transfer-learning for efficient training with only few real training examples;
2. Proposing an OCR method based on recurrent neural networks with a learning algorithm using synthetic data and a relatively small amount of real annotated data samples;
3. Building a novel dataset from the real historical data available on the Porta fontium portal dedicated to evaluation of OCR systems;
4. Evaluation of all methods on real data from Porta fontium portal;
5. **Giving an overview of how to create an efficient OCR system with minimal costs (i.e. minimal human effort during annotation process as well as the minimal time required to train models)**.

The paper structure is as follows. The following section gives an overview of related work. Section 3 describes our segmentation approaches used for page segmentation into individual blocks and lines, respectively. It also illustrates the pipeline of the whole process. Section 4 describes the OCR engine which is based on the combination of convolutional and recurrent neural networks. In Section 5, we present datasets used for our experiments. Then, we present experimental evaluation of the proposed methods on these data. This section also includes experimental set-up and hyperparameter settings. The last section concludes the paper and proposes some further research directions.

## 2 Related Work

This section is focused on four related research areas: image processing in general, segmentation, optical character recognition and finally, tools and systems for OCR and associated tasks (e.g. ground truth labelling).

### 2.1 Image processing in general

With the development of deep neural networks in the field of image processing, there are efforts to use an input image without any pre-processing (binarization or thresholding) and use an image-to-image architecture. Isola et al. [22] defined (as an analogy to automatic language translation) automatic image-to-image translation as the task of translating one possible representation of a scene into another, given sufficient training data. Such a translating operation, the goal of which is to predict pixels from pixels, has many applications that have an impact on the OCR task, i.e. binarization [1] or image segmentation [37]. Another example of image-to-image translating is the problem of detecting edges and object boundaries in natural images [47]. Huang et al. [21] go even further and present an unsupervised image-to-image translation the goal of which is to learn (based on a given image in the source domain) the conditional distribution of corresponding images in the target domain without seeing any examples of corresponding image pairs.

## 2.2 Segmentation

Before performing any document image processing task, layout analysis and page segmentation need to be implemented. Traditional approaches [9] usually use geometric algorithms or heuristics fined-tuned on a target domain. However, in the following text we focus on the use of neural networks which achieve state-of-the-art results in many research areas including segmentation and OCR.

Page segmentation is very similar to the semantic segmentation. Shelhamer et al. [40] show that a FCN trained end-to-end (pixels-to-pixels) can reach interesting results. The most important elements of a document are baselines of individual text lines and blocks of texts. Annotations in the dataset are segments of interest (baselines or text blocks) plotted in the image (e.g. image mask). An example of the FCN architecture – U-Net [37], was developed originally for medical image segmentation. With appropriate dataset, it can be simply trained on the text document images to perform a page segmentation. Another technique usable for semantic segmentation, object detection or page segmentation is a mask R-CNN [18]. For the feature extraction, the Mask R-CNN uses an FCN and it produces a class label and an appropriate bounding-box offset.

Breuel [7] describes the use of deep neural networks, in particular a combination of convolutional and multidimensional long short-term memory (LSTM) [20]. It is demonstrated that the relatively simple networks are capable of fast, reliable text line segmentation and document layout analysis even on complex and noisy inputs, without manual parameter tuning or heuristics. Furthermore, the method is easily adaptable to new datasets by retraining.

## 2.3 Optical character recognition

In recent years, the popular LSTM networks have been used for OCR tasks [8] because their training in the case of OCR is considerably simpler than previous training methods, since all training is carried out in terms of text line images and transcriptions. A combination of deep convolutional and recurrent neural networks (CRNN) is also used for line-based OCR [6] with even better performance. Convolutional neural networks [27] play a significant role in the feature extraction task. It is also a computationally efficient technique, which allows to process large images.

Graves et al. [16] introduced the connectionist temporal classification (CTC) alignment. According to them, the crucial step is to transform the network outputs into a conditional probability distribution over label sequences. The network can then be used as a classifier by selecting the most probable labelling for a given input sequence. More precisely, a CTC network has a softmax output layer with one extra unit. The activations are interpreted as the probabilities of observing the corresponding labels at particular times. The activation of the extra unit is the probability of observing no label (so called *blank symbol*). This algorithm (architecture and the corresponding loss function) allows us to train a classifier in a very efficient way, because we only need a ground-truth (GT) text sequence for the CTC loss function.

Furthermore, classical RNN requires pre-segmented training data to provide the correct target at each timestep, regardless its ability to model long-range dependencies. The CTC is a way to extend the RNN for this type of non-segmented data [12].

He et al. [19] used the deep-text recurrent network classifier with the CTC alignment for scene text reading, which is more difficult task than OCR. In their system, the CTC layer is directly connected to the outputs of LSTM layers, and works as the output layer of the whole RNN. Elagouni et al. [12] used the CTC for text recognition in videos.

Graves also introduced the sequence transduction as an alternative to CTC, which extends the CTC from discrete to continuous output sequences [15]. A combination of CTC with the attention mechanism [3] was utilized by Bluche et al. [4] in the handwritten text recognition task.

A line-based classifier which uses the CTC algorithm needs a huge number of training examples, which leads us to the synthetic data and data augmentation.

Jaderberg et al. presented a framework for recognition and synthetic data generation [23]. Margner et al. [30] presented synthetic data for Arabic OCR and another synthetic data generation was proposed by Gaur et al. [13]. Simply put, data augmentation is a way to increase the size of a dataset by perturbing existing data to create more examples. In the case of image processing, performing such operations as light rotations or skewing is a simple way to create new samples with low cost. If we consider the OCR task, we can perform a data augmentation operation on blocks of text, on text lines, on individual characters or even on a whole page. If we go to the extreme, we can even replace some characters with other similar looking ones ($0 \longrightarrow O$ or $i \longrightarrow l$ and so on). Perez et al. [35] tried to learn the best augmentations for a concrete dataset and classification task by a neural network – neural augmentation.

### 2.4 Existing tools and OCR systems

There is a number of tools (i.e. tool for Arabic OCR [30] or Aletheia [11]) which deal with the synthetic data generation and annotation. Aletheia is a full document image analysis system, which allows to annotate documents for layout evaluation (page element localization) and OCR in the form of XML file – e.g. PAGE XML [36]. There is also a web version of Aletheia[3].

OCRopus [5] is an efficient document analysis and OCR system. This system has a modular architecture and it is possible to use it freely for any purpose. The main components are dedicated to analysis of document layout, use of statistical language models and OCR. Last, but not least within the OCRopus, there is a tool ocropus-linegen that allows rendering text lines images, usable for training an OCR Engine.

Tesseract[4] is one of the best OCR engines in terms of integrated language support and recognition scores. It is available for Linux, Windows and Mac OS X, however, due to limited resources it is sufficiently tested only under Windows and Ubuntu [45]. The current version 4.0[5] uses a powerful LSTM based OCR engine and integrates models for 116 additional languages.

Transkribus [28,42] is another complex platform for analysis of historical documents which covers many research areas as for instance layout analysis, handwritten text recognition, etc. It includes also OCR using ABBYY Finereader Engine 11[6]. To the best of our knowledge, Tesseract and Transcribus are the best performing OCR systems. Therefore, they will be used for comparison with our approach.

[3] https://github.com/PRImA-Research-Lab/prima-aletheia-web

[4] https://github.com/tesseract-ocr/

[5] https://github.com/tesseract-ocr/tesseract/wiki/4.0-with-LSTM

[6] https://www.abbyy.com/

## 3 Page Layout Analysis and Segmentation

### 3.1 Whole process description

The OCR results depend on the layout analysis. Based on the layout analysis, we can perform image segmentation into smaller logical units as individual text blocks and lines which are the input of our OCR engine. The whole process is decomposed into three main tasks: block segmentation, line segmentation and optical character recognition itself as depicted in Figure 1.

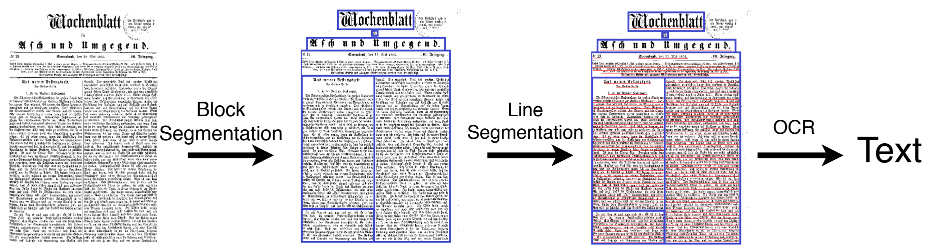


Fig. 1: Whole process pipeline: source image (left), text region segmentation (blue color; middle) and individual lines segmentation (red color; right).

The goal of the block segmentation is extracting text regions with respect to the reading order (an ordered set of image regions containing text is the output). The next task is segmenting the regions into individual text lines. Although there are some algorithms which can solve the text line segmentation from the whole page in one step, we prefer using the two-step approach, which allows to determine logical text units and simplifies determining the reading order. The subsequent segmentation into lines becomes then significantly easier.

Documents that we process have mostly a two-column layout. In the case of well separated columns, one-step approach would be sufficient. However, there are also more complicated pages with irregularities where the determination of reading order from coordinates of single lines is complicated. The one-step approach can also merge lines across column separators, which can jeopardize the reading order too. In most cases, the presented two-step approach is more appropriate because it takes into account the reading order and it is also able to filter out some types of noise, such as pictographic illustrations or various decorative elements. The last task is the OCR which converts the detected text lines into their textual representation.

Our segmentation method is as follows. The input page is first pre-processed which includes binarization and page rotation. The next step is the block segmentation. The page is first processed by an FCN which predicts a mask indicating the text region positions. Based on the predicted mask, we extract individual regions. The list of extracted text regions is the input to the line segmentation process. In this step, we apply a line segmentation method in order to obtain images of individual text lines. After a necessary post-processing which removes noise and parts of surrounding text lines we can feed the resulting text line images directly to the OCR engine. The above described layout analysis and segmentation processes are illustrated in Figure 2.

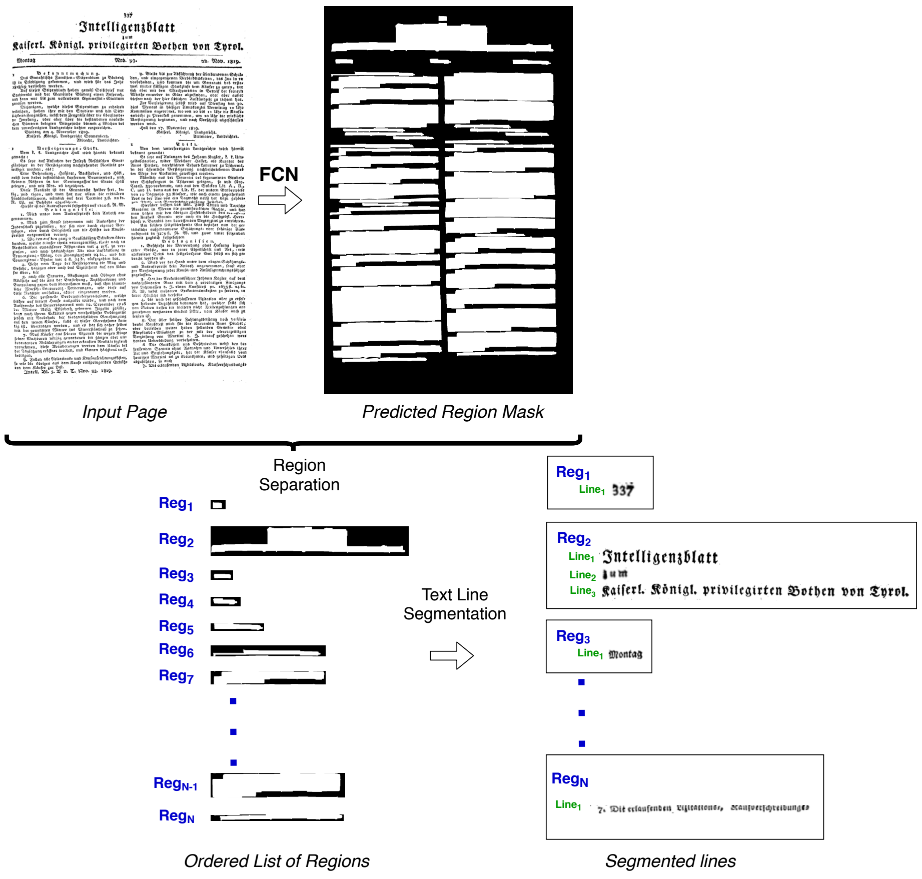


Fig. 2: Region and line segmentation tasks scheme

## 3.2 Text block segmentation

Recent methods for image segmentation are often based on fully convolutional networks. A well known example of such an architecture is U-Net [37] which was initially developed for semantic segmentation of medical images. The architecture of this network is depicted in Figure 3.

It consists of a contracting path (left side) and an expansive path (right side). The contracting path follows the typical architecture of a convolutional network. It consists of the repeated application of two $3 \times 3$ convolutions (unpadded convolutions), each followed by a rectified linear unit (ReLU) and a $2 \times 2$ max pooling operation with the stride 2 for downsampling. At each downsampling step we double the number of feature channels. Every step in the expansive path consists of an upsampling of the feature map followed by the $2 \times 2$ convolution ("up-convolution") that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two $3 \times 3$ convolutions, each followed by a ReLU. The cropping is necessary due to the loss of border pixels in every convolution. At the final layer the $1 \times 1$ convolution is used to map each 64-component feature vector to the desired number of classes.

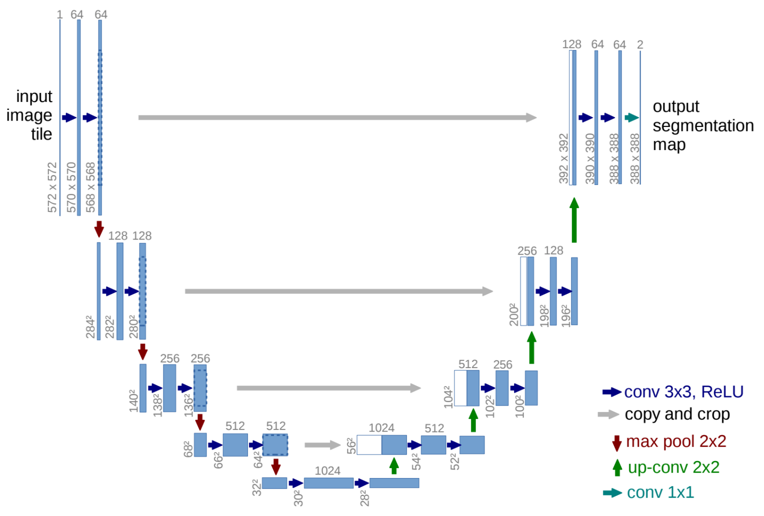


Fig. 3: U-Net architecture [37]

A modification of the U-Net model used for segmentation of historical document images was proposed by Wick and Puppe [46]. The main difference from U-Net is that it does not use skip connections. The whole architecture of this network is much simpler (see Figure 4) and the number of parameters is also much lower. The encoder part has 5 convolutional and two pooling layers. The size of convolution kernels is set to 5 and padding is used to keep the dimension. The decoder consists of 4 deconvolution layers. We will refer this modification as *U-Net (Wick)*.

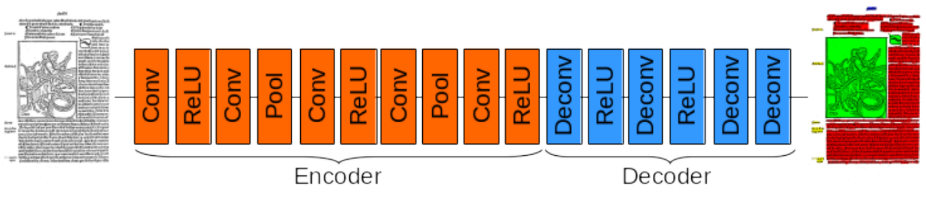


Fig. 4: Modified U-Net architecture proposed by Wick and Puppe [46]

We classify the image pixels into two classes: text region (regardless of whether the pixel is part of the text or background – white) and non-text region (black). The output of the FCNs is a prediction map with the same dimension as the input image. Each position in the map indicates the probability that a pixel belongs to the region. To obtain a binary image, we threshold the output. We use the widely used Otsu's [33] method for this task. The resulting binary image mask contains ones within the text regions and zeros elsewhere.

Based on the predicted segmentation mask, we divide the image into text regions and we also determine the reading order. We solve this task by recursive searching for horizontal and vertical separators in the segmentation map. Firstly, we search for horizontal separators and divide the image into several regions according to it. In the sequel, we attempt to divide the resulting regions. Again, we first search for horizontal separators and if none is found we search for the vertical ones. The recursion is applied, till we reach the desired level (granularity) of segmentation. We apply three levels of block segmentation on the data from Porta fontium. This approach respects the logical reading order of the processed documents. The outcome of this step is an ordered list of image regions containing text.

An example of the recursive search of region separators is shown in Figure 5. The red lines represent the horizontal separators (first level) while the green ones are the vertical separators (second level). The blue lines then mark the horizontal separators in the third level.

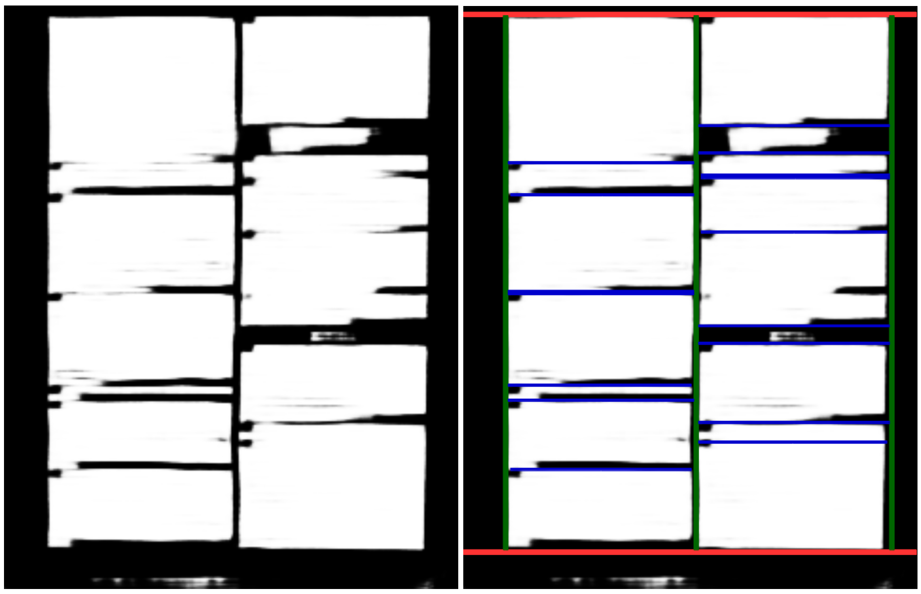


Fig. 5: Block segmentation example (recursive search of horizontal and vertical separators)

### 3.3 Text-line segmentation

In our previous work [31], we utilized a simple projection profile based text-line segmenter. The approach works very well for simple regions with regularly positioned lines. However, it might not be sufficient for more complex regions and hence, in this paper, we present two more sophisticated methods for text-line segmentation. We will refer to the original profile based method as *Profile*.

#### *3.3.1 ARU-Net*

Gruning et al. [17] proposed a novel deep neural network, called ARU-Net, which is able to detect text-lines in handwritten historical documents. This network extends U-Net by two more key concepts – spatial attention (A) and depth (R, residual structure). The attention mechanism allows the ARU-Net to focus on image content at different positions and scales [17]. As a consequence, it should lead to a better detection of texts with variable font sizes. Due to the deep structures of the architecture, residual blocks are used to address vanishing gradient issues [14].

Our method builds on the prediction result of the ARU-Net trained to recognize baselines. The output mask is first binarized and then we extract the positions of baselines. Next, we perform analysis of connected components on the binarized region image and we search the components that intersect with or are very close to the detected baseline. Based on the relevant component statistics, we can induce the boundaries of the text-line. We are then able to extract the line image according to the boundary. The process is depicted in Figure 6.

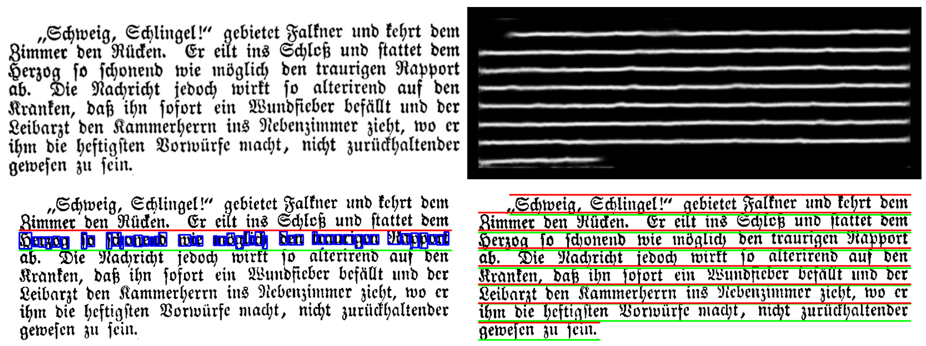

„Schweig, Schlingel!" gebietet Falkner und kehrt dem Zimmer den Rücken. Er eilt ins Schloß und ſtattet dem Herzog ſo ſchonend wie möglich den traurigen Rapport ab. Die Nachricht jedoch wirkt ſo alterirend auf den Kranken, daß ihn ſofort ein Wundfieber befällt und der Leibarzt den Kammerherrn ins Nebenzimmer zieht, wo er ihm die heftigſten Vorwürfe macht, nicht zurückhaltender geweſen zu ſein.

Fig. 6: ARU-Net text-line segmentation example

### 3.3.2 Kraken

Kraken[7] is an open-source OCR software forked from OCRopus. Although this tool offers a lot of features (e.g. script detection) we use only segmentation, more precisely text line bounding boxes prediction. When deploying on the extracted text block, it produces bounding boxes coordinates in a `json` file. The bounding boxes, based on `json` file, are depicted in Figure 7. Since Kraken provides only bounding boxes with a rectangular shape, it is beneficial to reduce segmentation errors by performing image pre-processing, especially deskewing and dewarping. Another option would be to use Kraken to locate word bounding boxes. In such a case, bounding boxes with rectangular shape would be sufficient.

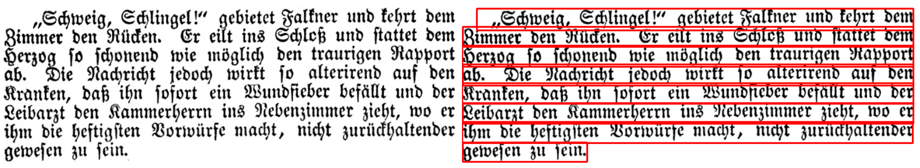

„Schweig, Schlingel!" gebietet Falkner und kehrt dem Zimmer den Rücken. Er eilt ins Schloß und ſtattet dem Herzog ſo ſchonend wie möglich den traurigen Rapport ab. Die Nachricht jedoch wirkt ſo alterirend auf den Kranken, daß ihn ſofort ein Wundfieber befällt und der Leibarzt den Kammerherrn ins Nebenzimmer zieht, wo er ihm die heftigſten Vorwürfe macht, nicht zurückhaltender geweſen zu ſein.

Fig. 7: Kraken text-line segmentation example

[7] https://github.com/mittagessen/kraken

Because the line bounding boxes sometimes overlap each other, it is necessary to remove the remainder of the text of the previous or next line (see Figure 8). We address this issue by removing connected components which are close to the bottom or top edge of the image.

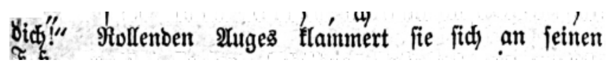

Dich!" Rollenden Auges klammert ſie ſich an ſeinen

Fig. 8: Overlapping text line example extracted by Kraken segmenter

## 4 OCR Engine

The classifier utilizes a combination of a convolutional neural network [27] and the LSTM neural network [20].

The inputs of our network are binarized line images with a dimension of 1250 × 40 pixels. Based on our preliminary experiments, the images are padded from the left by 50 pixels, so the input layer shape increases to 1300 × 40. On the input layer we apply two convolutional layers with 40 kernels with a shape of 3 × 3 and *MaxPooling* layers. After this first phase we obtain 40 feature maps and we also reduce dimensionality by scaling down the input to 325 × 10.

Through a reshaping mechanism we create a dense layer which is fed into two Bidirectional LSTM layers. Each Bi-LSTM layer comprises two LSTM layers which process the input from opposite sides. One LSTM layer contains 256 units. The outputs of the LSTMs from the first Bi-LSTM layer are merged by addition operation, while the outputs of the second pair of LSTMs are concatenated.

The concatenated output is given to a dense layer with softmax activation. It is a representation of probability distributions of individual symbols per each time frame. Let $\mathcal{A}$ ($|\mathcal{A}| = 90$) be a set of symbols. It includes out of vocabulary (*OOV*) and *blank symbol.* The most probable symbol $\hat{a}_t$ of each time frame $t$ is determined as:

$$\hat{a}_t = \underset{a_i \in \mathcal{A}}{\operatorname{argmax}}\, p_t^{a_i} \tag{1}$$

where $p_t^{a_i}$ is a probability of observing character $a_i$ at a given time $t$. At each time $t$ the sum of the probabilities of all symbols is equal to 1.

$$\sum_{i=1}^{|\mathcal{A}|} p_t^{a_i} = 1 \tag{2}$$

The final part of the classifier is a transcription layer, which decodes the predictions for each frame into an output sequence.

We use connectionist temporal classification (CTC) output layer. CTC is an output layer designed for sequence labeling with RNNs [16] and therefore it is capable of classifying unsegmented line images and retrieving the character sequences. The architecture of the classifier is depicted in Figure 9. Figure 10 shows another input image processed by the model and the CTC alignment. Note, that this model was already used in our previous work [31].

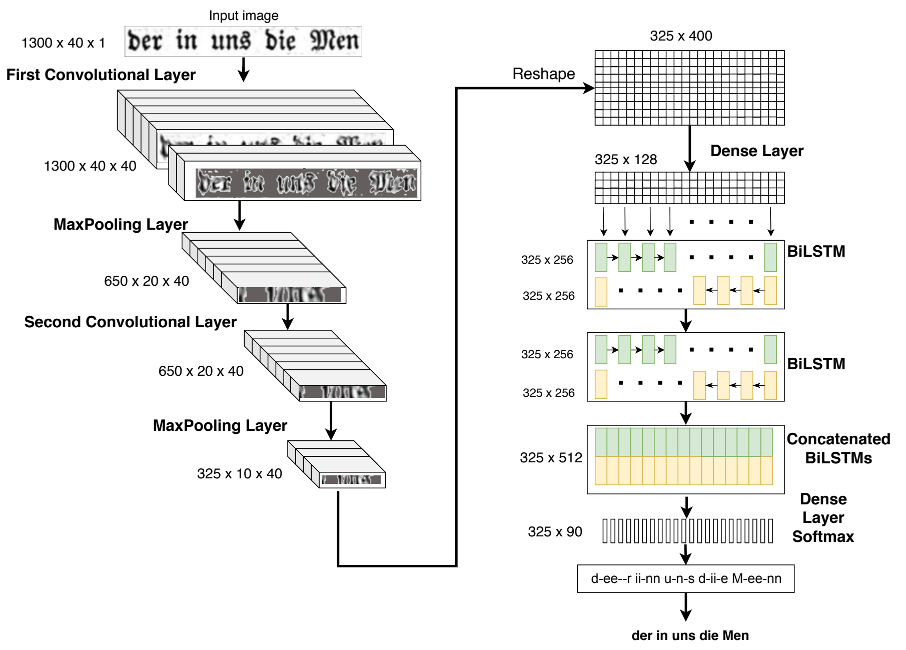


Fig. 9: OCR engine architecture

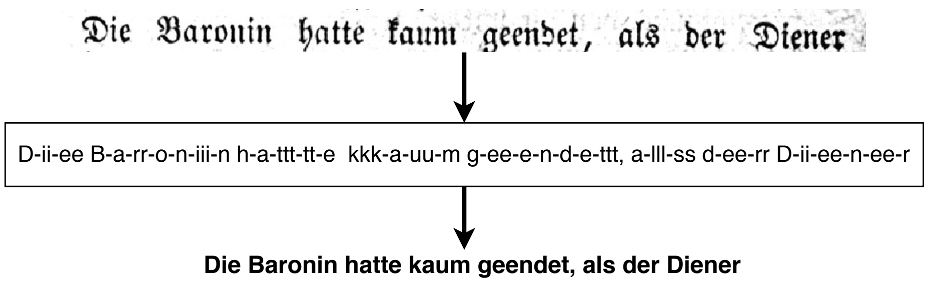

D-ii-ee B-a-rr-o-n-iii-n h-a-ttt-tt-e kkk-a-uu-m g-ee-e-n-d-e-ttt, a-lll-ss d-ee-rr D-ii-ee-n-ee-r
Die Baronin hatte kaum geendet, als der Diener

Fig. 10: Another example of CTC alignment

## 5 Datasets

In this section we describe the datasets we utilize for our experiments. The first one is the Europeana dataset. It was chosen because of its similarity to the data we are processing (newspaper pages). We use it to pre-train our page segmentation neural networks.

The second dataset is newly created from the Porta fontium portal[8]. It contains newspaper pages that we process in the frame of the *Modern Access to Historical Sources* project. It contains full transcriptions of the pages as well as layout information. It is utilized both for page segmentation and OCR engine training. It allows us to measure the quality of the developed methods on real data.

[8] http://www.portafontium.cz/

### 5.1 Europeana

The Europeana Newspapers Project Dataset [10] was created in the context of the Europeana Newspapers Project (ENP). The goal of the project is aggregating a representative set of historical newspapers. The dataset was created so that it takes into consideration all the challenges and issues related with the processing of historical document images. The dataset contains more than 500 newspaper pages and their ground truths containing full transcribed text, layout information and reading order.

From this set, we have selected a subset of 95 pages mostly written in German and with varying layouts so that it can provide enough flexibility for the text segmentation model. Table 1 illustrates the relevant statistical information and Figure 11 shows one page example from the Europeana dataset.

Table 1: Statistical information about the Europeana dataset

| | **Page #** | **Block #** | **Line #** | **Word #** | **Character #** |
|---|---|---|---|---|---|
| **Training** | 68 | 6,849 | 27,098 | 149,464 | 832,331 |
| **Validation** | 9 | 741 | 2,549 | 15,597 | 81,839 |
| **Testing** | 18 | 2,469 | 8,010 | 41,409 | 234,208 |

### 5.2 Porta fontium

Porta fontium is a project which aims at digitizing archival documents from Czech-Bavarian border area. The goal is to join again the Czech and German archival materials that come from the border area which were forcibly separated in the past.

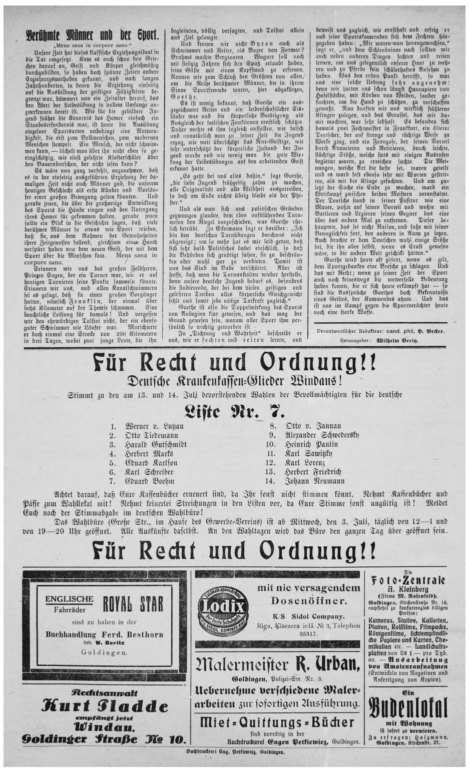
We have created an OCR and page layout analysis dataset from one selected newspaper, namely ”Ascher Zeitung”, printed in the second half of the nineteenth century. The main script used in this newspaper is ”Fraktur”. However, there are also some parts printed in Latin script and with different fonts. The dataset consists of 10 pages that are completely transcribed. All of them are accompanied with ground truths containing layout information and reading order. The ground truth is stored in the PAGE format [36].

We have selected the pages that are printed completely in Fraktur so that they can be used to train an OCR model for this script. We have divided the 10 pages into training, validation and testing parts (see Table 2 for the details). An example of one page is shown in Figure 12.

Table 2: Statistical information about the Porta fontium dataset

| | **Page #** | **Line #** | **Word #** | **Character #** |
|---|---|---|---|---|
| **Training** | 7 | 955 | 7,653 | 50,426 |
| **Validation** | 1 | 138 | 1,084 | 6,669 |
| **Testing** | 2 | 275 | 2,163 | 13,828 |

The second part of this dataset contains 15 additional pages with ground truths containing only the layout information. We have selected the pages with more complicated



## Berühmte Männer und der Sport.

„Mens sana in corpore sano.“

Unſere Zeit hat dieſes klaſſiſche Erziehungsideal in die Tat umgeſetzt. Kam es auch ſchon den Griechen darauf an, Geiſt und Körper gleichmäßig durchzubilden, ſo haben doch ſpätere Zeiten andere Erziehungsmethoden gekannt, und nach langen Jahrhunderten, in denen die Erziehung einſeitig auf die Ausbildung der geiſtigen Fähigkeiten begrenzt war, dämmert nun ein Zeitalter herauf, das den Wert der Leibesübung in vollem Umfange anzuerkennen bereit iſt. Wie für die gebildete Jugend früher die Kenntnis des Homer einfach ein Standeserfordernis war, iſt heute die Ausübung einzelner Sportsarten unbedingt eine Notwendigkeit, die erſt zum Vollmenſchen, zum modernen Menſchen ſtempelt. Ein Menſch, der nicht ſchwimmen kann, — lächelt man über ihn nicht eben ſo geringſchätzig, wie einſt gelehrte Kloſterſchüler über den Bauernburſchen, der nicht leſen kann?

Es wäre nun ganz verfehlt, anzunehmen, daß es in der einſeitig ausgeführten Erziehung der damaligen Zeit nicht auch Männer gab, die unſerem heutigen Geſchlecht als erſte Künder und Vorläufer einer großen Bewegung gelten könnten. Und gerade jenen, die über die großartige Entwicklung des Sports die Hände ringen und den Untergang ihres Homer für gekommen halten, gerade jenen ſollte ein Blick in die Geſchichte ſagen, daß viele berühmte Männer ſo etwas wie Sport trieben, daß ſie, ans dem Rahmen der Gewohnheiten ihrer Zeitgenoſſen fallend, gleichſam einen Hauch verſpürt haben von dem neuen Geiſt, der mit dem Sport über die Menſchen kam. Mens sana in corpore sano.

Erinnern wir uns des großen Feldherrn, Prinzen Eugen, der ein Turner war, wie er auf heutigen Turnieren ſeine Punkte ſammeln könnte. Erinnern wir uns, und allen Kanalſchwimmern ſei es geſagt, daß ſie einen großen Vorgänger hatten, nämlich Franklin, der einmal über ſechs Kilometer auf der Themſe ſchwamm. Eine beachtliche Leiſtung für damals! Und vergeſſen wir den ehrwürdigen Tolſtoi nicht, der ein ebenſo guter Schwimmer wie Läufer war. Marſchierte er doch einmal eine Strecke von 200 Kilometern in drei Tagen, wobei zwei junge Leute, die ihn begleiteten, völlig verſagten, und Tolſtoi allein ans Ziel gelangte.

Und kennen wir nicht Byron auch als Schwimmer und Reiter, als Boxer von Format? Brahms machte Bergtouren. Wagner ſoll noch mit ſiebzig Jahren ſich den Spaß erlaubt haben, ſeine Gäſte mit einem Kopfſtand zu erfreuen. Nennen wir zum Schluß den Größten von allen, um die Reihe berühmter Männer, die in ihrem Sinne Sportfreunde waren, hier abzukürzen, Goethe.

Es iſt wenig bekannt, daß Goethe ein ausgezeichneter Reiter und ein leidenſchaftlicher Eisläufer war und die körperliche Betätigung als Ausgleich der ſeeliſchen Funktionen ernſtlich ſchätzte. Daher mußte es ihm ſogleich auffallen, wie falſch und unnatürlich man zu ſeiner Zeit die Jugend erzog, wie weit überſchätzt das Nur-Geiſtige, wie ſehr unterſchätzt der körperliche Zuſtand der Jugend wurde und wie wenig man die gute Wirkung der Leibesübungen auf den arbeitenden Geiſt erkannt hatte.

„Es geht bei uns alles dahin,“ ſagt Goethe, „die liebe Jugend frühzeitig zahm zu machen, alle Originalität und alle Wildheit auszutreiben, ſo daß am Ende nichts übrig bleibt als der Philiſter.“

Und als man ſich aus politiſchen Gründen gezwungen glaubte, dem eben aufblühenden Turnweſen den Riegel vorzuſchieben, war Goethe ehrlich betrübt. Zu Eckermann ſagt er darüber: „Ich bin den deutſchen Turnübungen durchaus nicht abgeneigt; um ſo mehr hat es mir leid getan, daß ſich ſehr bald Politiſches dabei einſchlich, ſo daß die Behörden ſich genötigt ſahen, ſie zu beſchränken oder wohl gar zu verbieten. Damit iſt nun das Kind im Bade verſchüttet. Aber ich hoffe, daß man die Turnanſtalten wieder herſtelle, denn unſere deutſche Jugend bedarf es, beſonders die ſtudierende, der bei dem vielen geiſtigen und gelehrten Treiben alles körperliche Gleichgewicht fehlt und ſomit jede nötige Tatkraft zugleich.“

Goethe iſt alſo die Doppelwirkung des Sports von Anbeginn klar geweſen, und das mag der Grund geweſen ſein, warum aller Sport ihm perſönlich ſo wichtig geworden iſt.

In „Dichtung und Wahrheit“ beſchreibt er uns, wie er fechten und reiten lernte, und beweiſt uns zugleich, wie ernſthaft und eifrig er und ſeine Sportskameraden ſich dem Fechten hingegeben haben: „Wir waren nun herangewachſen,“ ſagt er, „und dem Schlendriane nach ſollten wir auch neben anderen Dingen fechten und reiten lernen, um uns gelegentlich unſerer Haut zu wehren und zu Pferde kein ſchülerhaftes Anſehen zu haben. Was den erſten Punkt betrifft, ſo war uns eine ſolche Uebung ſehr angenehm: denn wir hatten uns ſchon längſt Haurapiere von Haſelſtöcken, mit Körben von Weiden, ſauber geflochten, um die Hand zu ſchützen, zu verſchaffen gewußt. Nun durften wir uns wirklich ſtählerne Klingen zulegen, und das Geraſſel, das wir damit machten, war ſehr lebhaft. Es befanden ſich damals zwei Fechtmeiſter in Frankfurt, ein älterer Deutſcher, der auf ſtrenge und tüchtige Weiſe zu Werke ging, und ein Franzoſe, der ſeinen Vorteil durch Avancieren und Retirieren, durch leichte, flüchtige Stöße, welche ſtets mit einigen Ausrufen begleitet waren, zu erreichen ſuchte. Die Meinungen, welche Art die beſte ſei, waren geteilt und es ward faſt ebenſo ſehr mit Worten geſtritten, als mit der Klinge gefochten. Und um zuletzt der Sache ein Ende zu machen, ward ein Wettkampf zwiſchen beiden Meiſtern veranſtaltet. Der Deutſche ſtand in ſeiner Poſitur wie eine Mauer, paßte auf ſeinen Vorteil und wußte mit Battieren und Legieren ſeinen Gegner das eine über das andere Mal zu entfernen. Dieſer behauptete, das ſei nicht Raiſon, und fuhr mit ſeiner Beweglichkeit fort, den anderen in Atem zu ſetzen. Auch brachte er dem Deutſchen wohl einige Stöße bei, die ihn aber ſelbſt, wenn es Ernſt geweſen wäre, in die andere Welt geſchickt hätten.“

Goethe wird heute oft zitiert, wenn es gilt, dem Sportgedanken eine Breſche zu ſchlagen. Und das mit Recht; wenn zu ſeiner Zeit der Sport auch noch keineswegs die univerſelle Bedeutung haben konnte, die er ſich heute erkämpft hat — ſo ſind die Ausſprüche Goethes doch Bekenntniſſe eines Geiſtes, der Kommendes ahnte. Und das iſt uns im Kampf gegen die Sportverächter heute noch eine ſtarke Waffe.

Verantwortlicher Redakteur: cand. phil. H. Becker.
Herausgeber: Wilhelm Beritz.

**Für Recht und Ordnung!!**

**Deutſche Krankenkaſſen-Glieder Windaus!**

Stimmt zu den am 13. und 14. Juli bevorſtehenden Wahlen der Bevollmächtigten für die deutſche

**Liſte Nr. 7.**

| | | | |
|---|---|---|---|
| 1. | Werner v. Lutzau | 8. | Otto v. Jannau |
| 2. | Otto Tiedemann | 9. | Alexander Schwederſky |
| 3. | Harald Gutſchmidt | 10. | Heinrich Paulin |
| 4. | Herbert Marks | 11. | Karl Sawitzky |
| 5. | Eduard Karlſon | 12. | Karl Lorenz |
| 6. | Karl Schreiber | 13. | Herbert Friedrich |
| 7. | Eduard Boehm | 14. | Johann Neumann |

Achtet darauf, daß Eure Kaſſenbücher erneuert ſind, da Ihr ſonſt nicht ſtimmen könnt. Nehmt Kaſſenbücher und Päſſe zum Wahllokal mit! Nehmt keinerlei Streichungen in den Liſten vor, da Eure Stimme ſonſt ungültig iſt! Meldet Euch nach der Stimmabgabe im deutſchen Wahlbüro!

Das Wahlbüro (Große Str., im Hauſe des Gewerbe-Vereins) iſt ab Mittwoch, den 3. Juli, täglich von 12—1 und von 19—20 Uhr geöffnet. Alle Auskünfte daſelbſt. An den Wahltagen wird das Büro den ganzen Tag über geöffnet ſein.

**Für Recht und Ordnung!!**

ENGLISCHE Fahrräder ROYAL STAR
sind zu haben in der
Buchhandlung Ferd. Besthorn
Inh. W. Beritz
Goldingen.

Rechtsanwalt
**Kurt Fladde**
empfängt jetzt
Windau,
Goldinger Straße № 10.

Labākals apavu krēms
Lodix
Der beste Schuhputz
A/S Sidol Company, Riga
mit nie versagendem
Dosenöffner.
K/S Sidol Company.
Riga, Ķišezera ielā № 3, Telephon 35317.

**Malermeiſter R. Urban,**
Goldingen, Polizei-Str. Nr. 3.
Uebernehme verſchiedene Malerarbeiten zur ſofortigen Ausführung.

**Miet-Quittungs-Bücher**
ſind vorrätig in der
Buchdruckerei Eugen Petkiewicz, Goldingen.

Die
**Foto-Zentrale**
A. Kleinberg
(Firma M. Roſenfeld).
Goldingen, Kirchenſtraße Nr. 16,
empfiehlt zu konkurrenzlos billigen Preiſen:
Kameras, Stative, Kaſſetten, Platten, Rollfilme, Filmpacks, Röntgenfilme, lichtempfindliche Papiere und Karten, Chemikalien etc. — Landſchaftsplatten von Ls 1.— pro Dtzd. an. — Ausarbeitung von Amateuraufnahmen (Entwickeln von Negativen und Anfertigung von Kopien).

Ein
**Budenlokal**
mit Wohnung
iſt ſofort zu vermieten.
Zu erfragen: Holzmann,
Goldingen, Kirchenſtr. 27.

Buchdruckerei Eug. Petkiewicz, Goldingen.

Fig. 11: Page example from the Europeana dataset

layouts. The intended use of these additional pages is facilitating the training of page segmentation models.

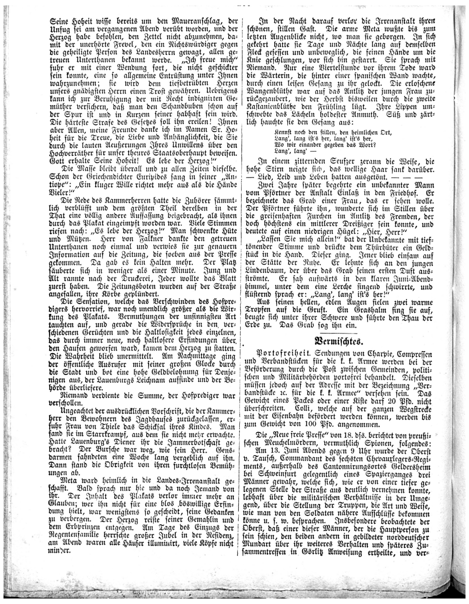
## 5.3 Synthetic data

Using synthetic data is very important in the proposed OCR training procedure, because we used such data for pre-training of the OCR models. This section analyzes different types of synthetic data creation. It is beneficial that these data are as similar as possible to the annotated images depicted in Figure 13. We have created two types of synthetic data. The first type is referred as *Hybrid* according to our previous work [31]. It is basically a composition of images containing single characters. The second one is referred as *Generated* and is produced simply by a generator with specified font.

### *5.3.1 Impact of the implicit language model*

Language models are often used in the OCR field to correct recognition errors [43]. Sabir et al. [38] showed that LSTM based models are able to learn language model implicitly.This LM is trained during the learning of the whole network.

Therefore, in the following experiments we will study the impact of this implicit language model for the final OCR accuracy. We assume that the best results will be obtained if



Seine Hoheit wiſſe bereits um den Maueranſchlag, der Unfug ſei am vergangenen Abend verübt worden, und der Herzog habe befohlen, den Zettel nicht abzunehmen, damit der unerhörte Frevel, den ein Nichtswürdiger gegen die geheiligte Perſon des Landesherrn gewagt, allen getreuen Unterthanen bekannt werde. „Ich freue mich“ fuhr er mit einer Wendung fort, die nicht geſchickter ſein konnte, eine ſo allgemeine Entrüſtung unter Ihnen wahrzunehmen; ſie wird dem tiefbetrübten Herzen unſers gnädigſten Herrn einen Troſt gewähren. Uebrigens kann ich zur Beruhigung der mit Recht indignirten Gemüther verſichern, daß man den Schandbuben ſchon auf der Spur iſt und in Kurzem ſeiner habhaft ſein wird. Die härteſte Strafe des Geſetzes ſoll ihn ereilen! Ihnen aber Allen, meine Freunde danke ich im Namen Sr. Hoheit für die Treue, die Liebe und Anhänglichkeit, die Sie durch die lauten Aeußerungen Ihres Unwillens über den Hochverräther für unſer theures Staatsoberhaupt beweiſen. Gott erhalte Seine Hoheit! Es lebe der Herzog!“

Die Maſſe bleibt überall und zu allen Zeiten dieſelbe. Schon der Griechendichter Euripides ſang in ſeiner „Antiope“: „Ein kluger Wille richtet mehr aus als die Hände Vieler!“

Die Rede des Kammerherren hatte die Zuhörer ſämmtlich verblüfft und dem größten Theil derelben in der That eine völlig andere Auffaſſung beigebracht, als ihnen durch das Plakat eingeimpft worden war. Viele Stimmen riefen nach: „Es lebe der Herzog!“ Man ſchwenkte Hüte und Mützen. Herr von Falkner dankte den getreuen Unterthanen noch einmal und verwies ſie zur genauern Information auf die Zeitung, die ſoeben aus der Preſſe gekommen. Da gab es kein Halten mehr. Der Platz ſäuberte ſich in weniger als einer Minute. Jung und Alt rannte nach der Druckerei, Jeder wollte das Blatt zuerſt haben. Die Zeitungsboten wurden auf der Straße angefallen, ihre Körbe geplündert.

Die Senſation, welche das Verſchwinden des Hofpredigers hervorrief, war noch unendlich größer als die Wirkung des Plakats. Vermuthungen der unſinnigſteu Art tauchten auf, und gerade die Widerſprüche in den verſchiedenen Gerüchten und die Haltloſigkeit jedes einzelnen, das durch immer neue, noch haltloſere Erfindungen über den Haufen geworfen ward, kamen dem Herzog zu ſtatten. Die Wahrheit blieb unermittelt. Am Nachmittage ging der öffentliche Ausrufer mit ſeiner großen Glocke durch die Stadt und bot eine hohe Geldbelohnung für Denjenigen aus, der Lauenburgs Leichnam auffinde und der Behörde überliefere.

Niemand verdiente die Summe, der Hofprediger war verſchollen.

Ungeachtet der ausdrücklichen Vorſchrift, die der Kammerherr den Bewohnern des Jagdhauſes zurückgelaſſen, erfuhr Frau von Thiele das Schickſal ihres Kindes. Man fand ſie im Starrkrampf, aus dem ſie nicht mehr erwachte. Hatte Lauenburg’s Diener ihr die Jammerbotſchaft gebracht? Der Burſche war weg, wie ſein Herr. Gensdarmen fahndeten eine Woche lang vergeblich auf ihn. Dann ſtand die Obrigkeit von ihren furchtloſen Bemühungen ab.

Meta ward heimlich in die Landes-Irrenanſtalt geſchafft. Bald ſprach nur hie und da noch Jemand von ihr. Der Inhalt des Plakats verlor immer mehr an Glauben; wer ihn nicht für eine blos böswillige Erfindung hielt, war wenigſtens ſo geſcheidt, ſeine Gedanken zu verbergen. Der Herzog reiſte ſeiner Gemahlin und dem Erbprinzen entgegen. Am Tage des Einzugs der Regentenfamilie herrſchte großer Jubel in der Reſidenz, am Abend waren alle Häuſer illuminirt, viele Köpfe nicht minder.

In der Nacht darauf verlor die Irrenanſtalt ihren ſchönen, ſtillen Gaſt. Die arme Meta wußte bis zum letzten Augenblicke nicht, wo man ſie geborgen. In ſich gekehrt hatte ſie Tage und Nächte lang auf demſelben Fleck geſeſſen und unbeweglich, die feinen Hände um die Knie geſchlungen, vor ſich hin geſtarrt. Sie ſprach mit Niemand. Nur eine Viertelſtunde vor ihrem Tode ward die Wärterin, die hinter einer ſpaniſchen Wand wachte, durch einen leiſen Geſang zu ihr gelockt. Die erloſchene Wangenblüthe war auf das Antlitz der jungen Frau zurückgezaubert, wie der Herbſt bisweilen durch die zweite Kaſtanienblüthe den Frühling lügt. Ihre Lippen umſchwebte das Lächeln holdeſter Anmuth. Süß und zärtlich hauchte ſie den Geſang aus:

Kennſt noch den ſtillen, den heimlichen Ort,
Lang’, lang iſt’s her, lang’ iſt’s her,
Wo wir einander gegeben das Wort?
Lang’, lang’ —

In einem zitternden Seufzer zerann die Weiſe, die hohe Stirn neigte ſich, das wellige Haar ſank darüber. — Lied, Leid und Leben hatten ausgetönt. — — —

Zwei Jahre ſpäter begehrte ein unbekannter Mann vom Pförtner der Anſtalt Einlaß in den Friedhof. Er bezeichnete das Grab einer Frau, das er ſehen wolle. Der Pförtner führte ihn, wunderte ſich im Stillen über die greiſenhaften Furchen im Antlitz des Fremden, der doch höchſtens ein mittlerer Dreißiger ſein konnte, und deutete auf einen niedrigen Hügel: „Hier, Herr!“

„Laſſen Sie mich allein!“ bat der Unbekannte mit tieftönender Stimme und drückte dem Thürhüter ein Geldſtück in die Hand. Dieſer ging. Jener blieb einſam auf der Stätte der Ruhe. Er lehnte ſich an den jungen Lindenbaum, der über das Grab ſeinen erſten Duft ausſtrömte. Er ſah aufwärts in den klaren Juni-Abendhimmel, unter dem eine Lerche ſingend ſchwirrte, und flüſternd ſprach er: „Lang’, lang’ iſt’s her!“

Aus ſeinen hellen, edlen Augen fielen zwei warme Tropfen auf die Gruft. Ein Grashalm fing ſie auf, beugte ſich unter ihrer Schwere und führte den Thau der Erde zu. Das Grab ſog ihn ein.

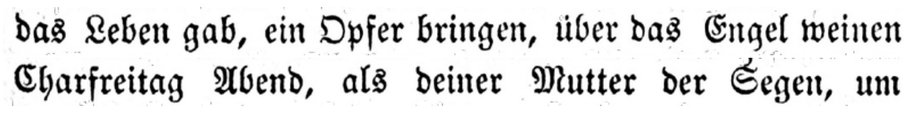
**Vermiſchtes.**

Portofreiheit. Sendungen von Charpie, Compreſſen und Verbandſtücken für die k. k. Armee werden bei der Beförderung durch die Poſt zwiſchen Gemeinden, politiſchen und Militärbehörden portofrei behandelt. Dieſelben müſſen jedoch auf der Adreſſe mit der Bezeichnung „Verbandſtücke ꝛc. für die k. k. Armee“ verſehen ſein. Das Gewicht eines Packes oder einer Kiſte darf 20 Pfd. nicht überſchreiten. Colli, welche auf der ganzen Wegſtrecke mit der Eiſenbahn befördert werden können, werden bis zum Gewicht von 100 Pfd. angenommen.

Die „Neue freie Preſſe“ von 18. dſs. berichtet von preußiſchen Meuchelmördern, vermuthlich Spionen, folgendes:

Am 13. Juni Abends gegen 9 Uhr wurde der Oberſt v. Tauſch, Commandant des ſechsten Chevauxlegers-Regiments, außerhalb des Cantonnirungsortes Geldersheim bei Schweinfurt gelegentlich eines Spazierganges drei Männer gewahr, welche ſich, wie er von einer tiefer gelegenen Stelle der Straße aus deutlich vernehmen konnte, lebhaft über die militäriſchen Verhältniſſe in der Umgegend, über die Stellung der Truppen, die Art und Weiſe, wie man von den Soldaten nähere Aufſchlüſſe bekommen könne u. ſ. w. beſprachen. Insbeſondere beobachtete der Oberſt, daß einer dieſer Männer, der die Hauptperſon zu ſein ſchien, den beiden andern in gebildeter norddeutſcher Mundart über ihr weiteres Verhalten und ſpäteres Zuſammentreffen in Görlitz Anweiſung ertheilte, und ver-

Fig. 12: Page example from the Porta fontium dataset



das Leben gab, ein Opfer bringen, über das Engel weinen
Charfreitag Abend, als deiner Mutter der Segen, um

Fig. 13: Examples of Fraktur script from Porta fontium dataset

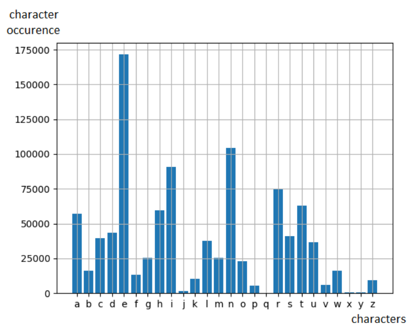
we use the training data from the target domain which are, in our case, historical German texts from the second half of the nineteenth century. To show the influence of the implicit model, we used three text sources for network training:

1. Completely random text – characters have a uniform probability distribution
2. Historical German text
3. Modern German text from the Reuters dataset [29]

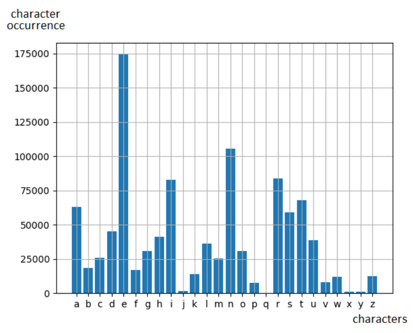
Although the second and the third type of data have similar distribution of characters (see Figure 14), the quality of the language model will slightly differ (types of expression, vocabulary, and so on).

### *5.3.2 Hybrid data*

The first approach to generate text lines for OCR training consists in concatenation of the images of individual characters. It must be done with respect to the target font though. Based on our previous studies [31], we utilize so called *Random space* approach for generating the hybrid data. This method adds a gap of a random size (within some reasonable bounds) between each pair of adjacent characters.



(a) Historical dataset



(b) Modern Reuters dataset

Fig. 14: Histograms of the character frequencies of the historical (left) and modern (right) text data sources

To introduce variances in our generated text lines, we have several different examples of each symbol and we use a randomly selected image for a given symbol. This is a form of data augmentation technique (e.g. Perez et al. [35]) because from one source text line it is possible to create several different corresponding images. Based on our previous experiments [31], we use random gaps between characters in the interval [1; 5] pixels. Figure 15 shows an example of generated data using this approach.

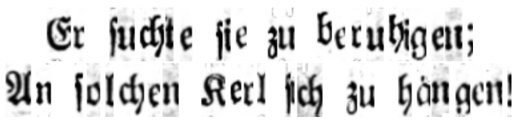

Er ſuchte ſie zu beruhigen;
An ſolchen Kerl ſich zu hängen!

Fig. 15: Examples of hybrid data generated by *Random space* method

### 5.3.3 Generated data

The second approach uses the data generated by the `TextRecognitionDataGenerator`[9]. This tool, written in Python, has many parameters that influence generated images, which includes for example background settings (white background or Gaussian noise). Moreover, it allows to choose a font, source texts and several types of image transformations (skewing or warping) to make the desired image of the text line. Figure 16 illustrates two line examples generated by this tool. These examples clearly show that the data are significantly different from the previous ones generated by *Random space* method, because the rendered character is always the same.

[9] https://github.com/Belval/TextRecognitionDataGenerator

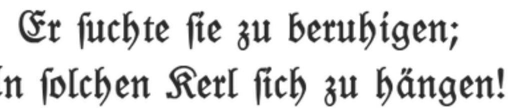

Er ſuchte ſie zu beruhigen;
An ſolchen Kerl ſich zu hängen!

Fig. 16: Two line examples of generated data by `TextRecognitionDataGenerator`

## 6 Experiments

All presented experiments are conducted on a PC with Intel Core i7 processor and 64GB RAM. All neural network based computing is performed on GPU GeForce RTX 2080 Ti with 11 GB RAM.

### 6.1 Layout analysis and segmentation

This experiment is carried out in order to identify the best block segmentation approach which will be then used in all following experiments.

Two text block segmentation approaches described in Section 3, namely *U-Net* and *U-Net (Wick)*, are evaluated on both the Porta fontium dataset.

Because the amount of training data in the Porta Fontium corpus ed, we utilize transfer learning [32] in this case. e train the models first on a subset of the Europeana newspaper dataset and then the models are fine-tuned on the training set of the Porta fontium dataset.

In both networks, we use ReLU [39] activation function for all layers and Adam [26] optimizer with 0.001 learning rate. We chose the binary cross-entropy as a loss function, since there are only two classes (a pixel is labelled either as a part of the text region or not).

Tables 3 and 4 show results of this experiment. The first table shows the ability to predict text blocks. The measures compare directly the segmentation result with the ground truth map. Table 4 presents the results computed only on foreground pixels within the text blocks.

The segmentation results are evaluated and visualized using DIVA layout evaluator [2] which calculates usual evaluation metrics for this task: Exact match, F1 score, Jaccard index and Hamming score.

Table 3: Text block segmentation: comparison of the performance of *U-Net* and *U-Net (Wick)* trained on Europeana and Porta fontium datasets, evaluated on Porta fontium dataset

| **Model** | **Exact match** | **F1 score** | **Jaccard index** | **Hamming score** |
|---|---|---|---|---|
| U-Net (Europeana) | 94.53 | 94.62 | 89.96 | 94.53 |
| U-Net (Porta fontium) | 97.67 | 97.67 | 95.50 | 97.67 |
| U-Net (Wick) (Europeana) | 95.36 | 95.42 | 91.36 | 95.36 |
| U-Net (Wick) (Porta fontium) | 97.60 | 97.60 | 95.36 | 97.60 |

We further visualize the segmentation results for qualitative analysis (see Figure 17). The green colour represents correctly assigned pixels (each green pixel is predicted as a part of a text region). The red colour indicates that the model predicted this pixel to be part

Table 4: Text block segmentation: comparison of the performance of *U-Net* and *U-Net (Wick)* trained on Europeana and Porta fontium datasets, evaluated on Porta fontium dataset, measured only on foreground (text) pixels

| **Model** | **Exact match** | **F1 score** | **Jaccard index** | **Hamming score** |
|---|---|---|---|---|
| U-Net (Europeana) | 99.65 | 99.65 | 99.30 | 99.65 |
| U-Net (Porta fontium) | 99.91 | 99.91 | 99.82 | 99.91 |
| U-Net (Wick) (Europeana) | 99.73 | 99.73 | 99.47 | 99.73 |
| U-Net (Wick) (Porta fontium) | 99.92 | 99.92 | 99.84 | 99.92 |

of a text region, but it is not. The blue (turquoise) coloured pixels are pixels that should be considered as part of a text region but the model omitted them.

Tables 3 and 4 show that the performance of both models is comparable and both of them are suitable for our task. The advantage of the Wick modification is its lower number of parameters and therefore the faster training. The results in Table 4 are very similar and comparable and one cannot make a unbiased decision. The numbers indicate that both networks are able to recognize the text content with the accuracy close to 100%. On the other hand, the results in Table 3 show slightly better performance of U-Net trained on Porta fontium dataset. The results of this table are more important for us as the main goal is to recognize the text regions. Figure 17 shows that *U-Net* model tends to better differentiate text paragraphs and could thus be more suitable for our task (significantly less red and turquoise colour).

The time needed for training of the U-Net model was 36 minutes on the Europeana dataset and 11 minutes on the Porta fontium dataset, respectively.

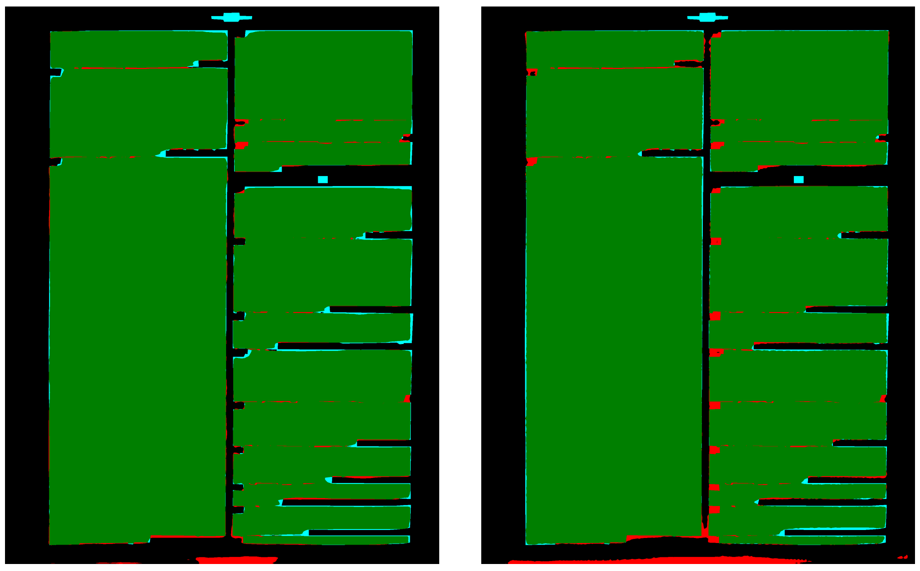

(a) U-Net (b) U-Net (Wick)

Fig. 17: Visualisation of text block segmentation using *U-Net* and *U-Net (Wick)* on Porta fontium dataset

## 6.2 OCR results when training on a small amount of real annotated data

We assume that using only a few real annotated samples for OCR model training will not be sufficient for reaching a good performance.

To support this hypothesis, we performed the following series of experiments to show the OCR results when our engine is trained from scratch using only a little real annotated data. To show the impact of the number of model parameters, we used two different models. The first one [31] has the input width of 650 pixels, while the second, more complex one, uses input width of 1250 pixels. We also evaluate and compare three previously described line segmentation approaches, namely *Profile*, *ARU-Net* and *Kraken*.

Our OCR engine utilizes ReLU activation function for all hidden layers. We use the CTC loss function and stochastic gradient descent (SGD) as an optimizer. For training the model from scratch, we apply 0.002 learning rate, while for fine-tuning, we set a learning rate 0.001. The output layer uses the softmax activation function.

All models are trained with early stopping. We train it until the validation lost begins to stagnate or decrease for some subsequent iterations. We ran all experiments 5 times and present the average values. This experimental settings is used in all following experiments.

For evaluation we use the standard word error rate (WER) and character error rate (CER) metrics. Additionally, we employ the edit distance, also known as the Levenshtein distance. All metrics are averaged across the text lines. For WER/CER, we first add up all deletions, insertions and substitutions in a text line and divide this value by the number of words/characters in the line. We average the number across all lines in the test corpus. The same procedure is applied to the edit distance metric.

A desirable value of CER, which is mostly reported to evaluate the quality of OCR systems, is below 1% for good quality printed texts. In our case, taking into consideration a lower quality of historical scans and the old language, an acceptable value lies around 2%.

Table 5 shows the OCR results of the smaller model having an input width of 650 pixels while Table 6 illustrates the results of the OCR model with the input width of 1250 pixels. The comparison of the two tables shows that the larger model needs the smaller number of epochs to train. On the other side, the larger model converges worse in the case of the Kraken line segmentation than the smaller one (significantly higher values of the all analyzed metrics).

Table 5: Comparison of the OCR results using different line segmentation methods: input width of 650 pixels and a training on a little real annotated data from Porta fontium dataset

| **Line segmentation method** | **Epoch #** | **Validation loss** | **WER** | **CER** | **Edit distance** |
|---|---|---|---|---|---|
| ARU-Net | 670 | 4.557 | 0.143 | 0.034 | 1.370 |
| Kraken | 568 | 4.142 | 0.142 | 0.030 | 1.216 |
| Profile | 549 | 3.538 | 0.160 | 0.036 | 1.615 |

The bottom line is that the training from scratch with a small amount of annotated data is possible, however, it is difficult to set the network hyper-parameteres so that the network converges. The obtained character error rates are also higher than desired and it is not possible to use such a model in a production system.

Table 6: Comparison of the OCR results using different line segmentation methods: input width of 1250 pixels and a training on a little real annotated data from Porta fontium dataset

| Line segmentation method | Epoch # | Validation loss | WER | CER | Edit distance |
|---|---|---|---|---|---|
| ARU-Net | 276 | 3.790 | 0.114 | 0.028 | 1.073 |
| Kraken | 288 | 12.555 | 0.234 | 0.070 | 2.813 |
| Profile | 306 | 2.909 | 0.163 | 0.036 | 1.584 |

## 6.3 OCR results when training on synthetic data

As has been proven in many previous studies [23, 24], synthetic data are of great importance when training OCR models. Therefore in this experiment, we evaluate and compare the performance of our OCR model trained only on different types of synthetic data. Moreover, this experiment evaluates the impact of the implicit language model (LM) on the OCR results. We use the U-Net based block segmentation and Kraken based line segmentation in this experiment, as it achieved the best results in the previous experiments.

Table 7 shows the results of this experiment. We use the model with the input size of $1250 \times 40$ pixels.

Table 7: Comparison of the OCR results of models trained only on different synthetic data

| Training data | Epoch # | Val. loss | WER | CER | Edit distance |
|---|---|---|---|---|---|
| **Hybrid data** | | | | | |
| Random text | 12 | 108.624 | 0.833 | 0.434 | 20.778 |
| Historical German | 10 | 32.830 | **0.664** | **0.205** | **9.613** |
| Modern German from Reuters | 9 | 83.857 | 0.805 | 0.358 | 16.927 |
| **Generated data** | | | | | |
| Random text | 9 | 157.321 | 0.977 | 0.725 | 39.166 |
| Historical German | 8 | 145.461 | 0.966 | 0.668 | 32.353 |
| Modern German from Reuters | 8 | 159.603 | 1.000 | 0.938 | 45.688 |

This table shows insufficient results for the model trained only on synthetic datasets. The best CER we obtained was around 20% for the hybrid training data based on historical German text which is far to be usable in real application. Moreover, the obtained average edit distance value is also very high (we need almost 10 operations - deletions, insertions or substitutions on average to transform the predicted text in the correct form). We can also conclude that the model based on hybrid data significantly outperforms the model trained on generated data.

Moreover, these results show that the implicit language model has an important impact on the final results. Hence, the historical German language model is the best option.

To sum up, it is crucial to use a text source from the target domain for implicit LM training. The training of the model took one hour and 20 minutes.

## 6.4 OCR results when training on synthetic data with fine-tuning on few real samples

The following experiments confirm the assumption that the models learned on synthetic data with a subsequent fine-tuning using small amount of real annotated samples bring a significant improvement in the OCR task. We took all three hybrid data models and performed additional training with all three types of extracted text line images (ARU-Net, Kraken, Profile).

Table 8: OCR results of the model pre-trained on synthetic data and fine-tuned using a small annotated dataset

| Training data | Epochs | Val. loss | WER | CER | Edit distance | Accuracy |
|---|---|---|---|---|---|---|
| **ARU-Net** | | | | | | |
| Hybrid random | 139 | 3.557 | 0.111 | 0.028 | 1.078 | 0.513 |
| Hybrid historical | 106 | 3.592 | 0.106 | 0.025 | 1.022 | **0.531** |
| Hybrid Reuters | 115 | 3.537 | 0.123 | 0.029 | 1.095 | 0.495 |
| **Kraken** | | | | | | |
| Hybrid random | 117 | 2.843 | 0.108 | 0.024 | 0.926 | 0.494 |
| Hybrid historical | 106 | 2.877 | **0.104** | **0.022** | **0.838** | 0.520 |
| Hybrid Reuters | 112 | 2.693 | 0.112 | 0.024 | 0.889 | 0.495 |
| **Profile** | | | | | | |
| Hybrid random | 90 | 3.291 | 0.130 | 0.030 | 1.308 | 0.455 |
| Hybrid historical | 94 | 3.265 | 0.120 | 0.028 | 1.265 | 0.451 |
| Hybrid Reuters | 100 | 3.459 | 0.140 | 0.031 | 1.382 | 0.428 |

The results of this experiment are depicted in Table 8. This table shows that the fine-tuning of the model significantly improves the final OCR performance. This experiment further shows that all results are more or less comparable, however, the model retrained on data provided by Kraken achieved the best results. Therefore, we chose this setting for the final experiment.

It is also worth noting that the accuracy is around 50%, which means that, after retraining and fine-tuning, we are able to perfectly recognize a half of our testing dataset. The training time of the fine-tuning was 15 minutes.

## 6.5 Comparison with selected OCR systems

This experiment compares the results of our OCR system on a whole page with Tesseract and Transkribus systems. Based on the previous experiment, we use U-Net based block segmentation, Kraken based line segmentation and Hybrid data with fine-tuning for training of our OCR engine.

In the case of Tesseract, we used two models which are available with the system, namely *deu_frak.traineddata* and *Fraktur.traineddata*. Both are trained on Fraktur script, which is the most common font in our dataset. We report the results of both models. We ran all OCR systems on ten annotated pages with two column layout. To improve the significance of this experiment, we carried out a five-folds cross-validation.

The results of this experiment are depicted in Table 9. This table shows that our OCR system outperformed both Tesseract and Transkribus systems, however the results obtained by Transkribus are almost comparable. Even though we made the cross-validation experiment, we obtained the best average accuracy value and we confirmed our accuracy result obtained previously (see Table 8). This table also shows that Transkribus using ABBYY Finereader Engine 11 outperformed Tesseract significantly.

Table 9: Comparison of the results of our OCR system with Tesseract and Transcribus on Porta fontium dataset

| | **Our approach** | **Tesseract (deu_frak)** | **Tesseract (Fraktur)** | **Transkribus** |
|---|---|---|---|---|
| Avg ACC | **0.488** | 0.221 | 0.217 | 0.398 |
| Avg ED | **1.137** | 2.518 | 2.152 | 1.230 |
| Avg WER | **0.118** | 0.191 | 0.187 | 0.120 |
| Avg CER | **0.024** | 0.053 | 0.045 | 0.027 |

## 7 Conclusions and Discussion

In this paper, we introduced a complex system for text segmentation and OCR of historical German documents. As a main result, we show that it is possible to use a small amount of real annotated data for training and achieve good results. We guided through the whole process of building an efficient OCR system for historical documents from the pre-processing steps through seeking an optimal strategy for training the OCR system.

We divided the paper into two logical parts. The first part dealt with page layout analysis and segmentation. Within it, we discussed approaches for page segmentation and we provided a comprehensive analysis. The second part is dedicated to the OCR system itself.

A great benefit is certainly the analysis and comparison of several methods for both the segmentation and OCR training. In the case of historical documents, we struggle with a lack of OCR methods that are adapted to such a domain and they usually need a huge training dataset. We also created a set of synthetic data with respect to the language of the given era. We compared several methods for synthetic data preparation and their influence on the final results. We also evaluated and compared a set of line segmentation approaches, namely ARU-Net, Kraken and simple projection profile based algorithm.

We can conclude that we have developed an efficient domain-dependent OCR system that focuses on historical German documents by picking the best tools and approaches available. We also provided a comparison with several state-of-the-art systems and showed that our system outperforms all of them on our task. Furthermore, we have created a novel Porta fontium historical dataset that can be used for segmentation experiments as well as for the OCR evaluation. Novelty of this paper also lies in the focus on minimal costs needed for the system training. We analyzed mainly the costs related to the preparation of annotated data which is a cornerstone when preparing such a system. We also presented and evaluated several scenarios how to train the best possible models with the limited annotated dataset.

Although this paper has presented several contributions, we would like to highlight the most important one, that is, the possibility to create an efficient OCR system even with a small amount of real annotated training data.

For future work, we plan to enrich our Porta fontium dataset with more annotated pages and we want to finish a transcription of the rest of 25 pages. Last but not least, we would like to state that this paper can be also considered as an overview of the state-of-the-art methods in areas relevant to historical document processing.

## Acknowledgement

This work has been partly supported from ERDF "Research and Development of Intelligent Components of Advanced Technologies for the Pilsen Metropolitan Area (InteCom)" (no.: CZ.02.1.01/0.0/0.0/17_048/0007267) and by Cross-border Cooperation Program Czech Republic - Free State of Bavaria ETS Objective 2014-2020 (project no. 211). We also thank Stephen Eugene Taylor for his valuable comments and advice.

## References

1. Afzal, M.Z., Pastor-Pellicer, J., Shafait, F., Breuel, T.M., Dengel, A., Liwicki, M.: Document image binarization using lstm: A sequence learning approach. In: Proceedings of the 3rd International Workshop on Historical Document Imaging and Processing, pp. 79–84. ACM (2015)
2. Alberti, M., Bouillon, M., Ingold, R., Liwicki, M.: Open Evaluation Tool for Layout Analysis of Document Images. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), pp. 43–47. Kyoto, Japan (2017). DOI 10.1109/ICDAR.2017.311
3. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
4. Bluche, T., Louradour, J., Messina, R.: Scan, attend and read: End-to-end handwritten paragraph recognition with mdlstm attention. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, pp. 1050–1055. IEEE (2017)
5. Breuel, T.M.: The ocropus open source ocr system. In: Document Recognition and Retrieval XV, vol. 6815, p. 68150F. International Society for Optics and Photonics (2008)
6. Breuel, T.M.: High performance text recognition using a hybrid convolutional-lstm implementation. In: Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, vol. 1, pp. 11–16. IEEE (2017)
7. Breuel, T.M.: Robust, simple page segmentation using hybrid convolutional mdlstm networks. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, pp. 733–740. IEEE (2017)
8. Breuel, T.M., Ul-Hasan, A., Azawi, M.I.A.A., Shafait, F.: High-performance ocr for printed english and fraktur using lstm networks. 2013 12th International Conference on Document Analysis and Recognition pp. 683–687 (2013)
9. Bukhari, S.S., Shafait, F., Breuel, T.M.: Improved document image segmentation algorithm using multiresolution morphology. In: Document recognition and retrieval XVIII, vol. 7874, p. 78740D. International Society for Optics and Photonics (2011)
10. Clausner, C., Papadopoulos, C., Pletschacher, S., Antonacopoulos, A.: The enp image and ground truth dataset of historical newspapers. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 931–935. IEEE (2015)
11. Clausner, C., Pletschacher, S., Antonacopoulos, A.: Efficient ocr training data generation with aletheia. Proceedings of the International Association for Pattern Recognition (IAPR), Tours, France pp. 7–10 (2014)
12. Elagouni, K., Garcia, C., Mamalet, F., Sébillot, P.: Text recognition in videos using a recurrent connectionist approach. In: International Conference on Artificial Neural Networks, pp. 172–179. Springer (2012)
13. Gaur, S., Sonkar, S., Roy, P.P.: Generation of synthetic training data for handwritten indic script recognition. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, pp. 491–495. IEEE (2015)
14. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics, pp. 249–256 (2010)
15. Graves, A.: Sequence transduction with recurrent neural networks. arXiv preprint arXiv:1211.3711 (2012)
16. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd international conference on Machine learning, pp. 369–376. ACM (2006)
17. Grüning, T., Leifert, G., Strauß, T., Michael, J., Labahn, R.: A two-stage method for text line detection in historical documents. International Journal on Document Analysis and Recognition (IJDAR) **22**(3), 285–302 (2019)

18. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp. 2961–2969 (2017)
19. He, P., Huang, W., Qiao, Y., Loy, C.C., Tang, X.: Reading scene text in deep convolutional sequences. In: Thirtieth AAAI conference on artificial intelligence (2016)
20. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
21. Huang, X., Liu, M.Y., Belongie, S., Kautz, J.: Multimodal unsupervised image-to-image translation. In: The European Conference on Computer Vision (ECCV) (2018)
22. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1125–1134 (2017)
23. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. arXiv preprint arXiv:1406.2227 (2014)
24. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Reading text in the wild with convolutional neural networks. International Journal of Computer Vision **116**(1), 1–20 (2016)
25. Karpathy, A., Johnson, J., Fei-Fei, L.: Visualizing and understanding recurrent networks. arXiv preprint arXiv:1506.02078 (2015)
26. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
27. LeCun, Y., Bengio, Y., et al.: Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks **3361**(10), 1995 (1995)
28. Leifert, G., Strauss, T., GrÃijning, T., Labahn, R.: Citlab argus for historical handwritten documents (2016)
29. Lewis, D.D., Yang, Y., Rose, T.G., Li, F.: Rcv1: A new benchmark collection for text categorization research. Journal of machine learning research **5**(Apr), 361–397 (2004)
30. Margner, V., Pechwitz, M.: Synthetic data for arabic ocr system development. In: Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on, pp. 1159–1163. IEEE (2001)
31. Martínek, J., Lenc, L., Král, P., Nicolaou, A., Christlein, V.: Hybrid training data for historical text OCR. In: 15th International Conference on Document Analysis and Recognition (ICDAR 2019), pp. 565–570. Sydney, Australia (2019). DOI 10.1109/ICDAR.2019.00096
32. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and transferring mid-level image representations using convolutional neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
33. Otsu, N.: A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics **9**(1), 62–66 (1979)
34. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: International conference on machine learning, pp. 1310–1318 (2013)
35. Perez, L., Wang, J.: The effectiveness of data augmentation in image classification using deep learning. arXiv preprint arXiv:1712.04621 (2017)
36. Pletschacher, S., Antonacopoulos, A.: The page (page analysis and ground-truth elements) format framework. In: 2010 20th International Conference on Pattern Recognition, pp. 257–260. IEEE (2010)
37. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention, pp. 234–241. Springer (2015)
38. Sabir, E., Rawls, S., Natarajan, P.: Implicit language model in lstm for ocr. In: Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, vol. 7, pp. 27–31. IEEE (2017)
39. Shang, W., Sohn, K., Almeida, D., Lee, H.: Understanding and improving convolutional neural networks via concatenated rectified linear units. In: international conference on machine learning, pp. 2217–2225 (2016)
40. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(4), 640–651 (2017). DOI 10.1109/TPAMI.2016.2572683. URL `https://doi.org/10.1109/TPAMI.2016.2572683`
41. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE transactions on pattern analysis and machine intelligence **39**(11), 2298–2304 (2017)
42. Strauss, T., Weidemann, M., Michael, J., Leifert, G., Grüning, T., Labahn, R.: System description of citlab's recognition & retrieval engine for icdar2017 competition on information extraction in historical handwritten records (2018)
43. Tong, X., Evans, D.A.: A statistical approach to automatic ocr error correction in context. In: Fourth Workshop on Very Large Corpora (1996)
44. Ul-Hasan, A., Breuel, T.M.: Can we build language-independent ocr using lstm networks? In: Proceedings of the 4th International Workshop on Multilingual OCR, pp. 1–5 (2013)

45. Vincent, L., Lead, U.T.: Announcing tesseract OCR. Google Code, http://googlecode. blogspot. com. au/2006/08/announ cing-tesseract-ocr. html Last accessed **1**(9), 2015 (2006)
46. Wick, C., Puppe, F.: Fully convolutional neural networks for page segmentation of historical document images. In: 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), pp. 287–292. IEEE (2018)
47. Xie, S., Tu, Z.: Holistically-nested edge detection. In: Proceedings of the IEEE international conference on computer vision, pp. 1395–1403 (2015)