CHRONICLING GERMANY: AN ANNOTATED HISTORICAL NEWSPAPER DATASET

Anonymous authors
Paper under double-blind review

ABSTRACT

The correct detection of dense article layout and the recognition of characters in historical newspaper pages remains a challenging requirement for Natural Language Processing (NLP) and machine learning applications on historical newspapers in the field of digital history. Digital newspaper portals for historic Germany typically provide Optical Character Recognition (OCR) text, albeit of varying quality. Unfortunately, layout information is often missing, limiting this rich source's scope. Our dataset is designed to enable the training of layout and OCR models for historic German-language newspapers. The Chronicling Germany dataset contains 693 annotated historical newspaper pages from the time period between 1852 and 1924. The paper presents a processing pipeline and establishes baseline results on in- and out-of-domain test data using this pipeline. Both our dataset and the corresponding baseline code are freely available online. This work creates a starting point for future research in the field of digital history and historic German language newspaper processing. Furthermore, it provides the opportunity to study a low-resource task in computer vision.

1 INTRODUCTION

Newspapers are essential sources of information, not just for modern readers, but particularly in the past when other communication channels like the internet or radio were not yet available. Even more importantly, historical newspapers allow historians and social scientists to study social groups' opinions and cultural values and to use information from newspapers in causal models (for more details, see A.4). This paper presents the *Chronicling Germany* -dataset, consisting of 693 annotated high-resolution scanned newspaper pages from the period between 1852 and 1924.

With the emergence of digital newspaper portals, using historical newspapers has become easier in recent years [1]. These portals provide text via Optical Character Recognition (OCR) but often lack reliable layout information for their German language content, which is essential for digital history applications, many of which would require newspaper articles to be treated as individual documents. Our dataset will help to reduce the character error rate and aims at considerably improving the detection of individual elements of a newspaper page, like articles or single advertisements. The former is important to prevent algorithms from connecting unrelated text regions and preserve the order in which text regions should be read. To this end, the text layout is systematically annotated using nine classes.

From a computer science view, a collection of successful approaches allows us to process modern documents (Blecher et al., 2023; Davis et al., 2022). For historical documents, large-scale data sets exist (Dell et al., 2024) but are mostly focused on English language material set in Antiqua-like typefaces. For continental European languages, existing datasets are much smaller (Abadie et al., 2022; Kodym & Hradis, 2021; Clausner et al., 2015; Nikolaidou et al., 2022) .

Until more annotated data becomes available, the processing of historical continental European newspaper pages is, therefore, a low-resource task, highlighting the need for more data. While low-resource tasks are well-established in natural language processing (Adams et al., 2017; Fadaee

[1] For Germany, e.g., the *Deutsche Zeitungsportal* (`https://www.deutsche-digitale-bibliothek.de/newspaper/`) and *zeit.punkt NRW* (`https://zeitpunkt.nrw/`)

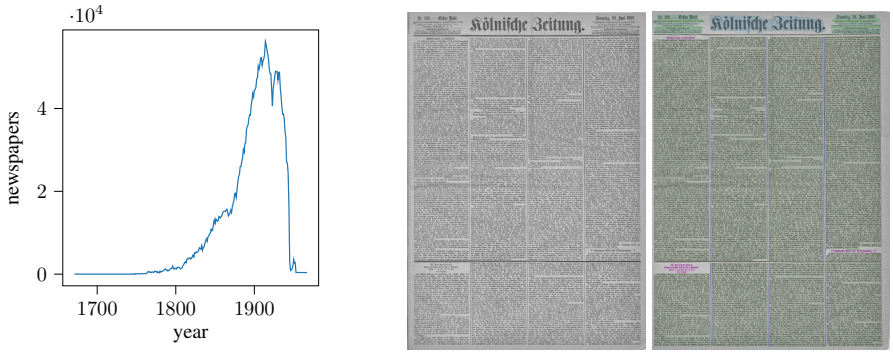


Figure 1: Left: Number of available digitized newspapers per year at `www.deutsche-digitale-bibliothek.de/newspaper` over time. Data from January 2024. Center: Front page of the *Kölnische Zeitung* from the 1^{st} of January 1924. Right: Corresponding annotation to the page at the center of this figure.

et al., 2017; Hedderich et al., 2021; Zoph et al., 2016) , low-resource settings remain under-explored in computer vision (Zhang et al., 2024) . Historical German newspapers are interesting in this context due to their dense layout (see also Supplementary Figure 5) and the Fraktur font. Fraktur differs significantly from the Antiqua typefaces that dominate modern Western texts. To the contemporary eye, Fraktur letters appear dense, which also impacts layout recognition. Furthermore, in addition to the font, our dataset features the archaic 'long s' or ' ſ ', which is no longer used today. The 'sz' or 'ß' is specific to the German language and also appears in the data. Historically, it emerged when the common combination 'ſz' merged into a single letter 'ß', unlike the 'long s' it still appears in contemporary texts. The abovementioned differences limit our ability to transfer existing solutions designed for modern documents or English-language historical newspapers. This motivates the collection of additional data.

The task of processing German newspapers is also highly relevant to history scholars. Especially in the 19th century, local communities, interest groups, and political parties created their own newspapers. The *Deutsche Zeitungsportal*[2] counts 698 German newspapers in 1780, this number rose to over 14,000 in 1860 and peaked at 50,848 papers in 1916 (see Figure 1).

Plenty of digitized pages allow researchers to systematically search for cultural values and historical change. Unfortunately, most of the pages available on such platforms contain either no or incorrect layout information. With text lines being in disarray, the pre-processed text data from these pages cannot be analysed computationally. Additionally, untrained modern human readers struggle with font differences, limiting the usefulness of unprocessed data to researchers lacking this specific skill. Thus, creating a pipeline capable of accurately processing this vast amount of data to a format readable to both a machine and a researcher without specific language and typeface skills is an important step in making these resources accessible. Furthermore, the availability of machine-readable newspaper archives is valuable to social scientists who recently started to use historical newspapers to track treatment variables or measure the impact of institutions or policies on social life (Beach & Hanlon, 2023).

Additionally, the layout of German historical newspapers is often complex, consisting of several columns, multiple horizontal sections and up to 500 elements to annotate per page. To create this dataset, eleven student assistants with a background in history have spent a total of 1,500 hours annotating the layout of 693 pages. These include approx. 1,900 individually annotated advertisements that consist of approx. 5,700 polygon regions. We also provide ground truth text annotations, which are not as costly since we start from network-generated OCR -output and correct errors. Overall, our dataset includes almost 30,000 layout polygon regions as well as approx. 350,000 text lines and almost 3 million words. See Table 1 for an overview of the dataset details.

This dataset is larger than the Europeana corpus Clausner et al. (2015) with its 528 pages from European newspapers, the 197-page "Deutscher Reichsanzeiger und Preußischer Staatsanzeiger"

[2]https://www.deutsche-digitale-bibliothek.de/newspaper/

Table 1: Datasheet listing newspaper names, page counts, lines, words and polygon-regions. As layouts change in a paper's history, we include issues from different years. In such cases a paper appears more than once in the table.

Year	Newspaper	Pages	Lines	Words	Regions
1785	Schwäbischer Merkur	11	1,023	7,071	77
1813	Donau Zeitung	4	302	1,704	47
1834	Fränkischer Kurier	4	422	3,095	62
1851	Ostpreussische Zeitung	4	1,176	9,428	123
1856	Der Bazar	11	1,331	10,253	114
1857	Berliner Börsen Zeitung	5	1,197	7,386	169
1866	Bonner Zeitung	4	1,405	10,209	176
1866	Neue Berliner Musikzeitung	8	1,054	7,805	116
1866	Fränkischer Kurier	6	1,809	12,204	302
1866	Pfälzer Zeitung	4	1,089	7,571	127
1866	Vossische Zeitung	31	4,739	33,499	1,129
1866	Weisseritz-Zeitung	8	818	5,609	114
1866	Kölnische Zeitung	420	249,483	2,133,614	17,054
1867	Hannoverscher Courier	4	1,994	13,563	226
1867	Neue preussische Zeitung	4	2,896	16,491	203
1852-1888	Special editions Kölnische Zeitung	20	831	8,044	134
1891	Bonner Zeitung	4	1,470	9,446	402
1924	Kölnische Zeitung	141	79,832	672,295	9,067
Sum		693	352,871	2,989,301	29,642

Table 2: Data-set sizes of this paper and related work in comparison.

Dataset	Pages
Chronicling Germany (ours)	693
Europeana (Clausner et al., 2015)	528
News Eye Finnish (Muehlberger & Hackl, 2021c)	200
Reichs- und Staatsanzeiger (UB-Mannheim, 2023b)	197
News Eye French (Muehlberger & Hackl, 2021a)	183
Neue Züricher Zeitung (Ströbel & Clematide, 2019)	167
News Eye Austrian (Muehlberger & Hackl, 2021b)	158
News Eye Competition (Michael et al., 2021)	100

(UB-Mannheim, 2023b) data set or the 167-page Neue Züriche Zeitung (Ströbel & Clematide, 2019; UB-Mannheim, 2023a) corpus. Ground truth lines are compared in Table 2. Our dataset adds a significant number of new lines. Consequently, we argue that progress has been made.

Our dataset features sections and elements that are especially challenging for OCR and baseline models. For example, advertisement pages mix large and small font sizes and include drop capitals, where the initial letter of an advertisement spans over multiple rows but is read as part of the first row. Both features are a challenge for the baseline detection task. Other challenges are fractions in stock exchange news and abbreviations in lists of casualties. Overall, 83,8% of our dataset scans fall into a range between 4500 and 5499 width times 6500 and 7499 height pixels. The largest and smallest width are 5375 and 1800, for the height we have a maximum of 7230 and a minimum of 2510 pixels. [3]

In summary, this paper makes the following contributions: (1) We introduce the *Chronicling Germany*-dataset. Its 693 manually annotated high-resolution pages make it the largest German-language historic newspaper dataset (see Table 1 for dataset details). (2) We establish a baseline recognition pipeline for the layout detection, text-line recognition, and OCR-tasks. (3) We verify generalization properties using 112 historic newspaper pages from the earlier 1785 - 1866 period, which we call the out-of-distribution test set. We observe good generalization performance for OCR tasks, despite layout generalization not being satisfactory. This is due to the fact, that for correctly reading text it is not necessary to correctly detect the class of a text segment. The dataset and code for our pipeline are freely available online. [4]

2 Related Work

Unfortunately, from a digital history perspective, many modern systems focus on recent data and suffer from poor performance in a historical setting.[5] The current situation has led to a large body of OCR error correction work (Carlson et al., 2023), highlighting the need for specialized data sets and software. Liebl & Burghard (2020), for example, combines existing open-source components for this task.

Related datasets include the Europeana corpus (Clausner et al., 2015), the Deutsche Reichsanzeiger (UB-Mannheim, 2023b), and the Neue Züricher Zeitung (UB-Mannheim, 2023a; Ströbel & Clematide, 2019). The Europeana dataset contains *528* annotated pages from European sources. The Reichsanzeiger and the Neue Züricher Zeitung sets consist of *197* and *174* annotated pages, respectively, but have so far only been used for OCR training but not in any layout training pipeline (cf. 2). The layout annotation of these two projects is comparable to ours but less granular, and we have annotated considerably more pages.[6] More recently Dell et al. (2024), published perhaps the largest American historical newspaper dataset to date. Their dataset also includes layout annotations. Our work complements these existing datasets by additionally providing compatible annotations for German historical newspapers that differ significantly from other Western European and American newspapers. Furthermore, we annotate advertisements in detail, which significantly add to the complexity of the OCR-task (Dell et al., 2024) and are not annotated in the Reichsanzeiger and the Neue Züricher Zeitung. Advertisements are particularly interesting to scholars of economic history who are interested in labour markets, for example. Globally, the Historical Japanese Dataset (Shen et al., 2020) , a collection of over 2,000 annotated pages of complex layouts from the Japanese Who is Who of 1953, is quite important.

2.1 Common processing pipeline elements

Layout Segmentation is a longstanding task in document processing. For example, dhSegment (Oliveira et al., 2018) proposes a UNet structure based on the popular ResNet50 architecture (He et al., 2016). As described by Ronneberger et al. (2015), the network features a contracting and an expanding part. The contracting subnetwork uses ResNet50 as an encoder, and an additional expansive

[3] Our dataset includes pages from 1866, when the Austro-Prussian War was raging in the German Bund.

[4] Repository links withheld for double-blind peer-review. We will restore the links once the review process has been completed.

[5] For an example see Figure 5 in A.2, alco compare Shen et al. (2020).

[6] We aim to combine those two datasets with Chronicling Germany in future work.

subnetwork produces segmentation maps at the resolution of the original input. Transformer-based solutions trained on modern documents are available for similar tasks (Davis et al., 2022). However, Convolutional Neural Networks (CNNs) are cheaper to run (Dell et al., 2024) and require less training data, making them a budget-friendly solution.

Baseline-detection or text-line detection, means finding the straight line that connects the base points from each letter. Early work employed quadratic splines for this task (Smith, 2007). Modern solutions often employ architectures devised for segmentation or object detection tasks. Kodym & Hradis (2021) for example, choose a U-Net. Object detection pipelines are alternatively used instead of baseline detection; e.g., Dell et al. (2024) work with YOLOv8. Following Kodym & Hradis (2021), we employ a U-Net to detect text baselines in this project. Our annotations are consistent with the Europeana-corpus from Clausner et al. (2015) and the work from UB-Mannheim (2023a;b) that also features Fraktur letters. This design choice allows combining our datasets in future work.

Optical Character Recognition (OCR) is an important tool in digital history. Liebl & Burghard (2020), successfully work with a topological feature extraction step followed by a classifier as described by Smith (2007) for the digitization of the *Berliner Börsen Zeitung*. Following Breuel (2007), Kiessling (2022) uses a Recurrent Neural Network (RNN) based system. Dell et al. (2024) apply the contrastive learning approach presented by Carlson et al. (2023). Using a vision encoder, characters are projected into a metric space. The system works because patches containing the same character will cluster together.

3 The Chronicling Germany Dataset

Our Dataset contains 693 pages from historic German newspapers, mostly from 1866, specifically from the period of the Austro-Prussian War. Of these 693 pages, 15 pages contain only advertisements with approx. 1,900 individual advertisement blocks. The backbone of the dataset is the *Kölnische Zeitung*, a large regional newspaper from Western Germany, but we also include newspaper pages from the entire German Empire. Overall, we consider the dataset to be a very good representation of the various layout styles of historical German newspapers. (For a more detailed description and justification of the composition of the dataset, see A.5).

We split our data into train, validation and test datasets (Table 3), where the test dataset consists of in distribution in distribution (id) and out of distribution out of distribution (ood). Id test pages are from the Kölnsche Zeitung, while ood pages are taken from other German newspapers (Table 1). The train and validation data splits only contain id data.

Polygons placed by our expert human annotators capture the layout for each page.[7] All annotations are stored in PAGE-XML files. The Polygons capture different text-region types. Subclasses can exist within these. Each region type has a unique XML tag: `TextRegion`, `SeparatorRegion`, `TableRegion` and `GraphicRegion`. Graphic regions are always assigned the class `image`. Within text regions, we include the following classes: `paragraph`, `header`, `heading`, `caption`, `inverted_text`. Within table regions, the only possible subclass is `table`. To facilitate correct reading order detection, we introduce the separator subclass `separator_vertical`, and `separator_horizontal`. Vertical separators highlight different columns of a page. Horizontal separators split the page into sections and are relevant for the reading order if they span over multiple columns. Otherwise, they are found at the beginning of a new article or between caption or header elements. The header category covers the newspaper's name, which appears at the top of the front pages. To the left and right of the newspaper name, historical newspapers often have smaller blocks with additional information, such as the name of the editor-in-chief, the publication date, or the subscription price. These polygons are annotated as captions. Polygons that cover paragraphs, headlines, and tables are annotated, respectively. See Figure 1 for an annotation sample. Overall, the dataset includes 29,642 polygon regions.

We primarily use a combination of the classes described above to annotate the historic advertisements. We have decided not to introduce new classes to avoid confounding the model's training. This applies, in particular, to the separator classes. Therefore, we use the classes `separator_vertical` and `separator_horizontal` for the annotation of separator regions around individual advertisements. Advertisements tend to use text blocks with bigger fonts. To be consistent with our

[7]The annotation process is documented in A.5, the annotation guidlines are reported in A.6.

Table 3: Dataset split (left), test data is divided into in distribution (id)- and out of distribution (ood)-data. The right hand side shows label distribution-percentages per pixel.

	pages	polygons	lines
train	492	21.819	279.097
validation	30	1.422	17.463
test id	59	3.014	33.586
test ood	112	3.387	22.725
sum	693	29.642	352.871

label	class	frequency
0	background	38.16%
1	caption	0.72%
2	table	2.88%
3	paragraph	53.61%
4	heading	0.97%
5	header	0.68%
6	separator vertical	0.62%
7	separator horizontal	0.60%
8	image	0.05%
9	inverted text	0.02%

annotations, we mark these as `heading` . For the same reason, the normal-sized text is annotated as `paragraph` . Additionally, we include the classes `inverted_text` and graphic elements as `image` . These are present, especially in the advertisement pages, as well as the 1924 pages. Table 3 illustrates this numerically. The two classes `inverted_text` and `image` are only present in a subset of the data, which explains its low share of pixels overall.

Regions of each page have a reading order number assigned to them. These numbers are assigned automatically and not corrected manually. Reading order is not the main scope of this dataset. Automatic assignment leads to satisfactory results for most pages. For advertisement pages, however, it does not. Yet, advertisements don't need a meaningful reading order, as they are comprised of elements that are independent of each other.

In addition to the layout data, we include transcribed text divided into text lines. In our dataset, each text line is comprised of a polygon, which contains all characters, as well as a baseline and the corresponding text. Baselines and text transcriptions are initially generated automatically using the pipeline proposed by Kodym & Hradis (2021) , and then corrected by expert annotators. Line polygons and baselines are only corrected when there are significant mistakes. This is especially the case within the advertisement pages, where some initial letters of advertisements span over more than one line. Correct drop capital detection is challenging for current text-line detectors. The correction process is ongoing. Overall, our dataset includes 352,871 text lines. The transcription follows the OCR-D guidelines, level 2 (Johannes Mangei, 2024) . This means the text is transcribed in a visual style, preserving, for example, the archaic 'long s' or ' ſ'. For a complete discussion, see supplemental section A.6.

4 Experiments and results

Data: All experiments work with fixed train, test and validation splits as outlined in Table 3.

Pipeline: Figure 2 presents a pipeline overview. Overall, we employ two U-Nets for layout recognition and text-line detection and, finally, a Long Short-Term Memory (LSTM) cell for OCR. The pixel-wise layout inference is converted into polygons during the post-processing step. We use targets like Kodym & Hradis (2021) for training the baseline U-Net. The model recognizes baselines, ascender-, descender-, and end-points, which are converted into line regions and baselines during post-processing. The post-processing code is an adapted version from Kodym & Hradis (2021). Contrary to their approach, we use the layout regions from the previous step to cut out parts of the image and identify all baselines for each region. These baselines are then used as input for the LSTM OCR model and the original image. The pipeline is sensitive to the character resolution. A small letter "a", for example, should be about 20x20 pixels in size. If the resolution deviates significantly (more than five pixels in either dimension), we rescale the input images accordingly.

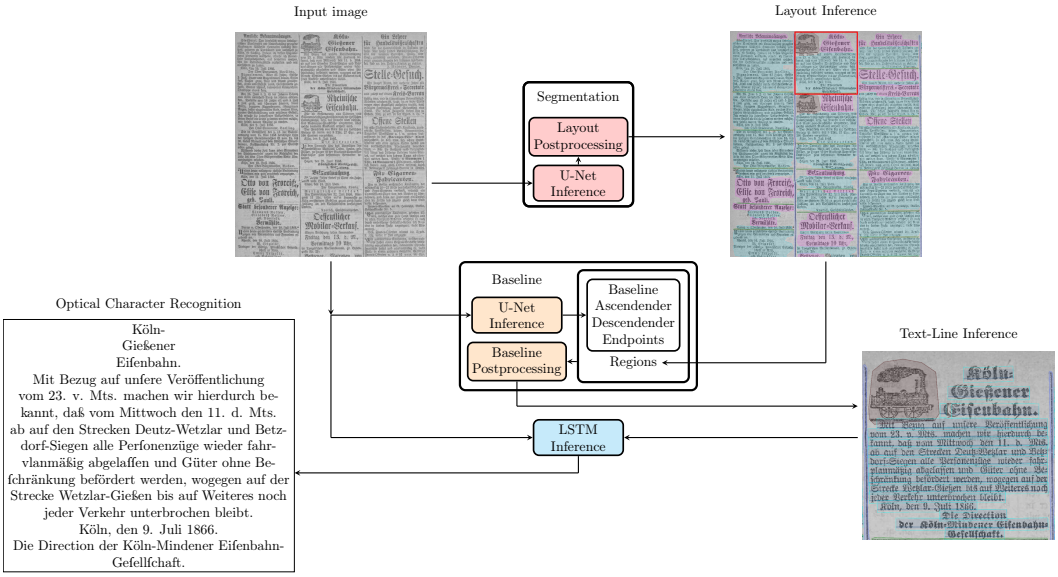


Figure 2: Flow chart of the entire prediction pipeline. The layout detection, text-line inference and Optical Character Recognition (OCR)-tasks use separate networks each. The output is machine-readable and can be processed further. For example in a machine translation step.

4.1 LAYOUT-SEGMENTATION

Training: Our layout segmentation setup follows Oliveira et al. (2018). For layout training, all pages are scaled down by a factor of 0.5 and split into 512 by 512-pixel crops. Cropping leads to 34,376 training crops overall. During training, we work with 24 crops per batch per graphics card. The training runs on a node with four graphics processing units (GPUs) . Consequently, the effective batch size is 96, with 358 training steps per epoch. Initially, optimization of the contracting network part can start from pre-trained ImageNet weights, while optimization of the expanding path has to start from scratch. The expanding subnetwork starts with the encoding from the contracting network and produces a segmentation output at the input resolution. To improve generalization, input crops are augmented using rotation, mirroring, gaussian blurring, and randomly erasing rectangular regions. An AdamW-Optimizer (Loshchilov & Hutter, 2017) trains this network with a learning rate of 0.0001, with a weight decay parameter of 0.001 for 50 Epochs in total, while using early stopping to save the best model. We use transfer learning via pre-training on the Europeana dataset (Clausner et al., 2015). We initialize the encoder using ImagNet weights, train on Europeana first and continue training on our data. We use only in distribution data for training and validation.

Results: Table 4 column two lists network performance on the test dataset, and column four lists performance on the id test data only. We compute F1 Score values for all individual classes on pixel level. Generally, we find good performance with id data, while ood data poses a challenge. Table 7 additionally shows overall test data compared to ood only data. Results of the ood only data show, that rarer classes like headlines and separators are a challenge, while the paragraph class shows good generalization. In all cases the especially rare classes `image` and `inverted_text` are not recognized as well. Figure 4 presents an id advertisement page from our test set with ground truth and prediction side by side.

We also compare our model results to the pipeline developed by Dell et al. (2024) . For this purpose, we evaluated our test set on their pipeline and report the results in columns three and five of table 4. Overall, our pipeline performs slightly better on the comparable classes of our test dataset then Dell et al. (2024). However, we do not fine-tune their model on our training data and there are significant differences between the Chronicling Germany dataset and the American Stories dataset, that distort the comparison. The most significant difference between the two datasets are the more detailed

Table 4: Layout detection results. This table lists F1 Score values for all individual classes. N/A values in columns (3) and (5) are due to differing annotations between our approach and Dell et al. (2024). In columns (2) and (4) we report our model results for in distribution (id) + out of distribution (ood) and in distribution (id) only data. (Out of distribution only results for layout in Table 7)

	F1 Score (id + ood)		F1 Score (id only)	
class	ours	Dell et al.	ours	Dell et al.
background	0.84 ± 0.001	0.80	0.96 ± 0.003	0.82
caption	0.36 ± 0.017	N/A	0.82 ± 0.031	N/A
table	0.46 ± 0.019	0.43	0.76 ± 0.037	0.49
paragraph	0.91 ± 0.003	0.88	0.99 ± 0.001	0.90
heading	0.63 ± 0.008	0.53	0.87 ± 0.009	0.60
header	0.46 ± 0.010	0.10	0.88 ± 0.028	0.10
separator vertical	0.27 ± 0.027	N/A	0.83 ± 0.009	N/A
separator horizontal	0.56 ± 0.019	N/A	0.89 ± 0.006	N/A
image	0.08 ± 0.016	N/A	0.25 ± 0.029	N/A
inverted text	0.02 ± 0.002	N/A	0.14 ± 0.035	N/A

headline annotations in the Chronicling Germany dataset.[8] Dell et al. (2024) seem to assign the headline class less frequently to headlines that do not stand out clearly. This leads to a significant amount of headline regions from the Chronicling Germany test-set to be classified as paragraph by the Dell-pipeline. Furthermore, the Chronicling Germany dataset includes annotated separator regions, while American Stories does not.[9] Moreover, Dell et al. (2024) treats the entire header of a newspaper front page as one class, while Chronicling Germany differentiates between the header itself and the captions that are typically left and right of the header.

4.2 Baseline Detection

Training: Following Kodym & Hradis (2021) we train an U-Net for the text-baseline prediction task. The raw input image as well as ground truth baselines serve as starting points for the optimization. The training process minimizes a joint text-line and text-block detection objective as introduced by Kodym & Hradis (2021) . We run an AdamW-optimizer (Loshchilov & Hutter, 2017) with a learning rate of 0.0001 and a batch size of 16. During training, inputs are randomly cropped to 256 by 256 images. To improve the robustness of the resulting network the input pipeline includes color jitter, gaussian blur, random grayscale and gaussian blur perturbations during training.

Results: We measure precision, recall, and F1 score (see Table 5). Generally, we observe values around 0.9. These observations are in line with Kodym & Hradis (2021), who observe similar numbers on the cBAD2019 dataset (Diem et al., 2017). Dell et al. (2024) do not provide baseline data, instead opting for extracting the text for each region as a whole employing a Yolo v8 architecture. Since we did not annotate the ground-truth text boxes, there is no adequate way to compare the two pipelines for this specific task. The historic newspaper community either works with baseline detection or direct text object detection pipelines. We decided to follow Kodym & Hradis (2021) and employ a U-Net to detect text baselines. This is consistent with other European projects like the Europeana-corpus from Clausner et al. (2015) and the work from UB-Mannheim that also features Fraktur letters. This choice allows combining these European datasets in future work, and is a key design decision, since we aim to boost performance in the Fraktur-subset of historical newspapers.

4.3 Optical Character Recognition (OCR)

Training Based on the Kraken-OCR-engine (Kiessling, 2022) we train a LSTM-cell for the OCR -task and employ baselines to extract individual line polygons. Alongside the annotations, which our

[8] Segmentation of headlines is of particular interest for historical research, as they allow for the layout based identification of and differentiation between individual articles.

[9] In the case of the Kölnische Zeitung identifying horizontal page spanning separators is of cruicial importance for the reading order, as they act like a page break. This means the reader should not continue reading down the current column but go back up to the next one.

Table 5: Baseline detection results. We measure performance in precision, recall and F1 score. Detected lines are matched with ground truth lines and are considered a true positive if the predicted line has an IoU score of more than 0.7 compared to the corresponding ground truth line. Results are averaged over all test pages.

Model	precision	recall	F1 score
UNet	0.934 ± 0.008	0.892 ± 0.01	0.911 ± 0.008

Table 6: Optical Character Recognition (OCR) results. Levenshtein distance per character appears in the first column. We computed the percentage of completely error-free lines for each model. The second column lists these results. Finally, we consider a line to have many errors if we observe a Levenshtein distance of more than 0.1 per character. We report the percentage of many error lines in the final column. We list the mean and standard deviation for multiple seeds.

Model	Data	Levenshtein-Distance	fully correct [%]	many errors [%]
LSTM (UBM 2024)	id + ood	0.02	43.5	7.2
	id only	0.01	48.9	3.2
	ood only	0.03	35.5	13.0
LSTM finetuned (ours)	id + ood	0.02 ± 0.001	60.5 ± 0.34	8.1 ± 0.66
	id only	0.01 ± 0.001	71.3 ± 0.21	2.9 ± 0.19
	ood only	0.04 ± 0.004	44.6 ± 1.27	15.8 ± 1.96
Transformer (ours)	id + ood	0.04 ± 0.01	56.2 ± 1.3	12.5 ± 2.3
	id only	0.04 ± 0.01	66.1 ± 1.64	9.7 ± 2.7
	ood only	0.04 ± 0.01	41.7 ± 0.89	16.6 ± 2.13

human domain experts have checked, these serve as input and ground truth pairs. Adam (Kingma & Ba, 2015) optimizes the network with a learning rate of 0.001. Optimization runs for eight epochs with a batch size of 32 sequences. We use early stopping to prevent the model from overfitting and include pixel-dropout, blur, rotation and see-through-like augmentations during training to improve generalization.

Results: Compared to the model trained by the Universitätsbibliothek Mannheim (Jan Kamlah, 2024) we observe improved results after finetuning (see: Table 6). We also find the OCR-transformer proposed by Kodym & Hradis (2021) in their pero-application with our LSTM results still being slightly better. The antiqua-pretrained OCR model from Dell et al. (2024) does not generalize well to the Fraktur-texts. For this pipeline, we observe an average Levensthein distance of 0.58 on the test set (not included in Table 6).[10]

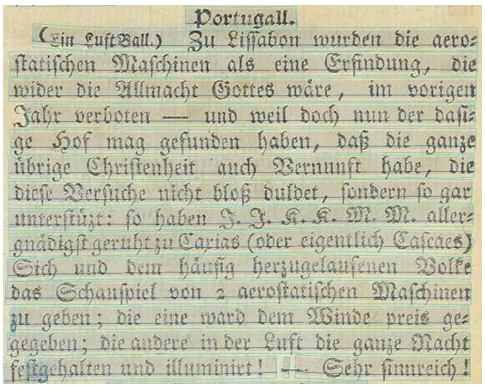
4.4 Overall pipeline performance

So far, we have evaluated components individually using ground truth inputs from previous steps. We additionally evaluate the complete pipeline on the test set (Table 9). We choose the best model of each component, according to the validation set (30 pages), to use in our pipeline. Then, we evaluate the resulting transcription with our ground truth. All predicted and ground truth lines are matched based on the intersection over the minimum of the corresponding text lines. Lines without a match were paired with an empty string. Our pipeline achieves an overall Levenshtein distance per character of 0.03 across the entire test set.

5 Pipeline-Generalization

Test-Data : We train only on in distribution data from the Kölnsche Zeitung, as this is the by far largest part of our data. To verify generalization, our test dataset contains 112 out of distribution

[10]Please note that we cannot fine-tune the OCR engine proposed by Dell et al. (2024) on our Fraktur-data because of differences in the text detection step (see above).



Portugall.
(Ein LuftBall.) Zu Liſſabon wurden die aeroſtatiſchen Maſchinen als eine Erfindung, die wider die Allmacht Gottes wäre, im vorigen Jahr verboten — und weil doch nun der daſige Hof mag gefunden haben, daß die ganze übrige Chriſtenheit auch Vernunft habe, die dieſe Verſuche nicht bloß duldet, ſondern ſo gar unterſtützt: ſo haben J. J. K. K. M. M. allergnädigſt geruht zu Caxias (oder eigentlich Caſcaes) Sich und dem häufig herzugelaufenen Volke das Schauſpiel von 2 aeroſtatiſchen Maſchinen zu geben; die eine ward dem Winde preis gegegeben; die andere in der Luft die ganze Nacht feſtgehalten und illuminirt! — Sehr ſinnreich!

Dortugall.
(Ein LuftBall.) Zu Liſſ abon wurden die gero ſtatiſchen Ma ſchinen alſ eine Erfindung, die wider die Allmacht Gotteſ wäre, im vorigen fahr verboten — und weil doch nun der daſige Hof mag gefunden haben, daß die ganze übrige Chriſtenheit auch Vernunft habe, die dieſe Verſuche nicht bloß duldet , ſondern ſo gar unrerſtüzt: ſ o haben J. J. K. K. M. M. allergnadigſi geruht zu Caxiaſ (oder eigentlich Caſcaef) ſich und dem haufig herzugelaufenen Volke daſ ſchau ſpiel von 2 acroſtati ſchen Maſchinen zu geben; die eine ward dem Winde preiſ gegegeben; die andere in der Luft die ganze Nacht tgehalten und illuminirt!
— fehr finnreich!

Figure 3: Generalization test set sample image. This figure shows a page element with detected baselines on the left. The right side presents the automatically created transcription.

pages from many different papers and time periods (Table 1 and Table 3). These contain also a small amount of out of domain antiqua font, enabling the evaluation not only on the trained Fraktur font.

Inference : Additional to the full test set evaluation, we run the entire pipeline on the out of distribution data only (Table 9). Overall, we measure a Levenshtein distance per character of 0.06. Figure 3 presents an example taken from a 1785 issue of the Schwäbischen Merkur. The sample is a report from Portugal. Readers learn that hot-air balloons or "aeroſtatiſche Maſchinen" where banned "last year" because hot-air balloons were initially deemed to be "incompatible with the omnipotence of god". Later, however, the court changed it's mind and bought two for a demonstration. Linguistically, the sample is close enough to modern German to be machine-translated.

6 Conclusion and Future Work

This work introduces the *Chronicling Germany* -dataset, the currently largest dataset of German language historic newspaper pages. In addition to the dataset, it presents a neural network-based processing baseline with test-set OCR-accuracy results. Our paper creates a starting point for researchers who wish to improve historical newspaper processing pipelines or are looking for a low-resource computer vision challenge. To create the dataset, history students spent 1,500 hours annotating the layout. The dataset's 693 pages, make it the largest fully annotated collection of historic German newspaper pages. The dataset includes 1,900 individually annotated advertisements. Furthermore, the out of distribution part of our test set includes 112 pages from historic newspapers that are not part of the training set. We verify baseline pipeline performance on the out-of-distribution pages. By following the OCR-D annotation guidelines (Johannes Mangei, 2024) we ensure our annotations' compatibility with concurrent and future work.

References

Nathalie Abadie, Edwin Carlinet, Joseph Chazalon, and Bertrand Duménieu. A benchmark of named entity recognition approaches in historical documents application to 19 th century french directories. In *International Workshop on Document Analysis Systems*, pp. 445–460. Springer, 2022.

Oliver Adams, Adam Makarucha, Graham Neubig, Steven Bird, and Trevor Cohn. Cross-lingual word embeddings for low-resource language modeling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 937–947, 2017.

Brian Beach and W. Walker Hanlon. Historical newspaper data: A researcher's guide. *Explorations in Economic History*, 2023.

Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents. *arXiv preprint arXiv:2308.13418*, 2023.

Cameron Blevins. Space, nation, and the triumph of region. a view of the world from houston. *Journal of American History*, 2014.

Thomas Breuel. Announcing the ocropus open source ocr system. *The official Google Code Blog entry, April*, 9, 2007.

Jacob Carlson, Tom Bryan, and Melissa Dell. Efficient ocr for building a diverse digital history. *arXiv preprint arXiv:2304.02737*, 2023.

Christian Clausner, Christos Papadopoulos, Stefan Pletschacher, and Apostolos Antonacopoulos. 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.

Brian Davis, Bryan Morse, Brian Price, Chris Tensmeyer, Curtis Wigington, and Vlad Morariu. End-to-end document recognition and understanding with dessurt. In *European Conference on Computer Vision*, pp. 280–296. Springer, 2022.

Melissa Dell, Jacob Carlson, Tom Bryan, Emily Silcock, Abhishek Arora, Zejiang Shen, Luca D'Amico-Wong, Quan Le, Pablo Querubin, and Leander Heldring. American stories: A large-scale structured text dataset of historical us newspapers. *Advances in Neural Information Processing Systems*, 36, 2024.

Markus Diem, Florian Kleber, Stefan Fiel, Tobias Grüning, and Basilis Gatos. cbad: Icdar2017 competition on baseline detection. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pp. 1355–1360. IEEE, 2017.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. Data augmentation for low-resource neural machine translation. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pp. 567–573. Association for Computational Linguistics, 2017. doi: 10.18653/V1/P17-2090. URL https://doi.org/10.18653/v1/P17-2090.

Andreas Ferrara, Joung Yeob Ha, and Randall Walsh. Using digitized newspapers to address measurement error in historical data. *The Journal of Economic History*, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. A survey on recent approaches for natural language processing in low-resource scenarios. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pp. 2545–2568. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.NAACL-MAIN.201. URL https://doi.org/10.18653/v1/2021.naacl-main.201.

Jan Kamlah. Universitätsbibliothek mannheim, german newspapers ocr model. https://github.com/JKamlah/german-newspapers-ocr-model/tree/main/data/kraken/text/german_newspapers_topologies/kraken, 2024. Online; accessed 4 June 2024.

Johannes Mangei. Ground truth guidelines. https://ocr-d.de/en/gt-guidelines/trans/, 2024. Online; accessed 5 June 2024.

Benjamin Kiessling. The Kraken OCR system, April 2022. URL https://kraken.re.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Oldrich Kodym and Michal Hradis. Page layout analysis system for unconstrained historic documents. *CoRR*, abs/2102.11838, 2021. URL https://arxiv.org/abs/2102.11838.

Bernhard Liebl and Manuel Burghard. From historical newspapers to machine-readable data: The origami ocr pipeline. *Workshop on Computational Humanities Research, November 18–20, 2020, Amsterdam, The*, 2020.

Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017. URL http://arxiv.org/abs/1711.05101.

Johannes Michael, Max Weidemann, Bastian Laasch, and Roger Labahn. Dataset of ICPR 2020 Competition on Text Block Segmentation on a NewsEye Dataset, June 2021. URL https://doi.org/10.5281/zenodo.4943582.

Guenter Muehlberger and Guenter Hackl. NewsEye / READ AS training dataset from French Newspapers (19th, early 20th C.), November 2021a. URL https://doi.org/10.5281/zenodo.5654841.

Guenter Muehlberger and Guenter Hackl. NewsEye / READ AS training dataset from Austrian Newspapers (19th, early 20th C.), November 2021b. URL https://doi.org/10.5281/zenodo.5654907.

Günter Muehlberger and Günter Hackl. NewsEye / READ AS training dataset from Finnish Newspapers (19th C.), November 2021c. URL https://doi.org/10.5281/zenodo.5654858.

Konstantina Nikolaidou, Mathias Seuret, Hamam Mokayed, and Marcus Liwicki. A survey of historical document image datasets. *International Journal on Document Analysis and Recognition (IJDAR)*, 25(4):305–338, 2022.

Sarah Oberbichler and Eva Pfanzelter. Topic-specific corpus building: A step towards a representative newspaper corpus on the topic of return migration using text mining methods. *Journal of Digital History*, 2021.

Sofia Ares Oliveira, Benoit Seguin, and Frederic Kaplan. dhsegment: A generic deep-learning approach for document segmentation. In *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 7–12. IEEE, 2018.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241. Springer, 2015.

Zejiang Shen, Kaixuan Zhang, and Melissa Dell. A large dataset of historical japanese documents with complex layouts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

Ray Smith. An overview of the tesseract ocr engine. In *Ninth international conference on document analysis and recognition (ICDAR 2007)*, volume 2, pp. 629–633. IEEE, 2007.

Phillip Ströbel and Simon Clematide. Improving ocr of black letter in historical newspapers: The unreasonable effectiveness of htr models on low-resolution images. In *Proceedings of the Digital Humanities 2019, (DH2019)*, 2019. accepted.

UB-Mannheim. Ground truth for neue zürcher zeitung black letter period. https://github.com/UB-Mannheim/NZZ-black-letter-ground-truth, 2023a.

UB-Mannheim. reichsanzeiger gt. https://github.com/UB-Mannheim/reichsanzeiger-gt/, 2023b.

Yunhua Zhang, Hazel Doughty, and Cees GM Snoek. Low-resource vision challenges for foundation models. *Computer Vision and Pattern Recognition Conference*, 2024.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. Transfer learning for low-resource neural machine translation. In Jian Su, Xavier Carreras, and Kevin Duh (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pp. 1568–1575. The Association for Computational Linguistics, 2016. doi: 10.18653/V1/D16-1163. URL https://doi.org/10.18653/v1/d16-1163.

A SUPPLEMENTARY

A.1 ACRONYMS

CNN Convolutional Neural Network

GPU graphics processing unit

id in distribution

LSTM Long Short-Term Memory

NLP Natural Language Processing

OCR Optical Character Recognition

ood out of distribution

RNN Recurrent Neural Network

A.2 ADDITIONAL FIGURES

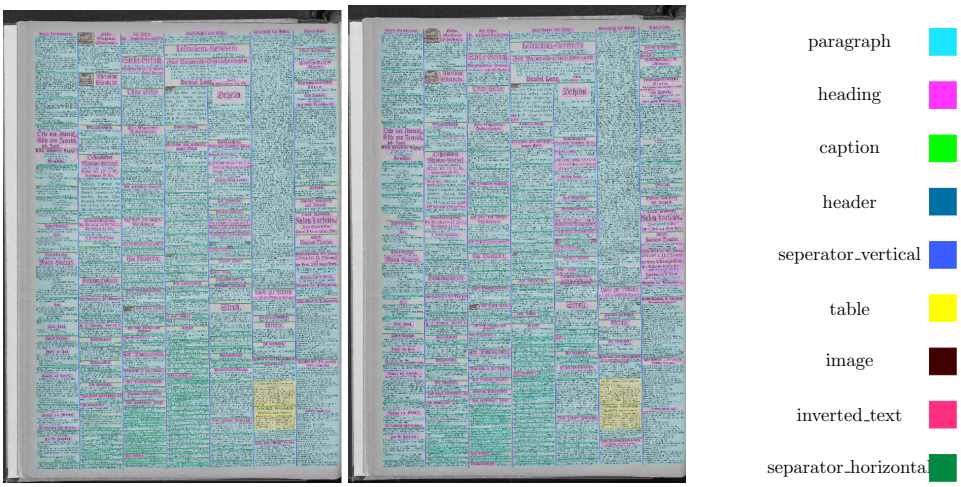
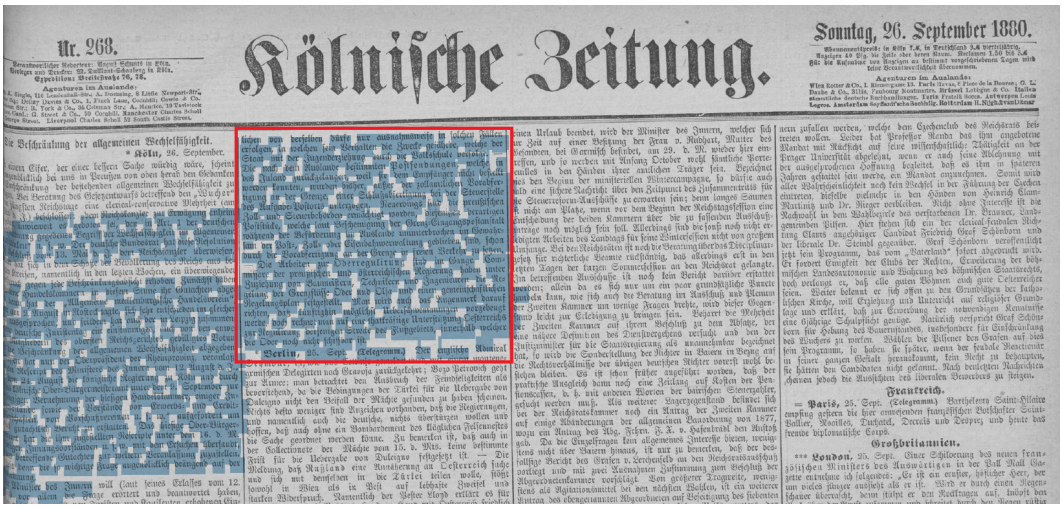


Figure 4: Target labels on the left and segmentation prediction on the right. The top left part of this advertisement page also appears in Figure 2.

A.3 PROJECT LIMITATIONS AND SOCIAL IMPACT

This dataset contains newspaper pages set in Fraktur-letters. The font is very different from modern fonts. The 'long s' or 'ſ', for example, is completely foreign to modern eyes. While our generalization dataset also includes four pages in Antiqua font, which have been predicted with sufficient accuracy, networks trained exclusively on our dataset are not likely to outperform more specialized networks on modern newspaper pages. There still exist some limitations of our pipeline. This includes the correct recognition of the drop-capitals present in advertisement pages and abbreviations and fractions in the market and stock exchange reports of the newspapers.

Ideally, our work will enable the processing of millions of pages of historical data, making vast resources easily available to future researchers who can then build upon the transcribed source material, for example, with machine translation and Natural Language Processing (NLP) pipelines. Countless research questions concerning economic, societal, political and scientific development can be addressed with such data. For a more detailed description of the relevance of such data for historical research, see Supplementary Section A.4 . We hope this dataset will help to improve our understanding of the past. We therefore expect a positive impact on society as a whole.


Nr. 268.

Kölnische Zeitung.

Sonntag, 26. September 1880.

Figure 5: Layout recognition error in the *Kölnische Zeitung*. A researcher tried to select text in the first column on the very left but the column layout it not understood correctly. This page was published on September 26, 1880. Digital versions are available at the *zeit.punkt NRW* website. Layout recognition and transcription generated by Transkribus.

A.4 Machine learning is important for the study of history

Figure 1 illustrates the breakthrough of the newspaper industry in the 19th century.[11] While the number of newspapers listed in the *Deutsche Zeitungsportal* grew at a rate of 2.9 percent p.a. in the first two-thirds of the 19th century, the increase rose to 3.4 percent p.a. after the foundation of the German Empire. There were three main reasons for this increase: firstly, the literacy of the population increased over the century. Secondly, considerable technological advances made it easier to produce a newspaper. Thirdly, state control of newspapers declined from the middle of the century. A significant milestone was the Press Act of 1874, which finally abolished censorship (although some restrictions remained so that even after 1874, there was no complete freedom of the press in the German Empire). Nevertheless, no later than the last third of the 19th century, a mass market for print media had emerged in Germany, which was served by many newspapers whose content and political orientation were very heterogeneous.

Historical newspapers contain a wealth of information about past societies. They provide information about the spatial occurrence of events, about contemporary perceptions of social and economic change, and allow tracing of cultural change. Blevins (2014), for example, uses the mentioning of place names in the *Houston Post* to draw a mental map of the Nation around 1900. Measured by the mention of place names, the region west of Houston was deeply rooted in the newspaper and its readership. The East Coast and the Midwest were also present in the imagination of contemporaries. However, the Southwest, the Northwest, and California hardly appear on this mental map. Based on the newspaper's coverage, one could argue that readers of the *Houston Post* around 1900 were barely aware of the Nation as a geographical entity. In economic history, historical newspapers have recently been used to identify treatments or measure variables of interest. Beach & Hanlon (2023) give an overview of the recent use of historical newspaper data in economic history. An interesting recent example is Ferrara et al. (2024) , who used digitized newspaper archives to measure a county's exposure to the boll weevil around 1900. The boll weevil is a pest of cotton that hit the American South between 1892 to 1922. The pest reduced cotton production and, consequently, hastened social changes in the primarily Black rural communities, like the fertility transition and higher investment in education.

[11] Note that the Deutsche Zeitungsportal does not collect all historical newspapers. There is probably a selection bias towards more prominent outlets with extended publication periods. However, on the whole, figure 1 should reflect the development of the newspaper market in Germany quite well.

Even though newspaper portals are an essential source for historians and other disciplines interested in history, such as economics, their potential has not yet been fully realized (Beach & Hanlon, 2023) . Firstly, researchers have so far mainly used US-American portals. The reason for this bias may be these portals have been established longer than in other regions of the world. Secondly, the mass utilization of newspaper data is often limited to a keyword search, which usually only covers the entire page and does not discriminate between articles. Therefore, the joint occurrence of two or more search terms is recorded for the page, not the article, and information retrieval is thus still very imprecise (Oberbichler & Pfanzelter, 2021) . Thirdly, the text cannot always be downloaded easily, which makes further processing by researchers more difficult. On the other hand, the image files of individual newspaper pages are easy to obtain via the portals. Deep learning algorithms that recognize the layout of a newspaper page and capture the text at the article level, therefore, promise great benefits for historical research. The *Chronicling Germany* data set presented here, comes with layout annotations for every page. It is intended to stimulate the further development of deep learning algorithms and to promote the increased use of non-American newspaper portals.

In addition to more accurate and straightforward information retrieval, downloadable article-level data will also allow scholars of history to apply advanced NLP-methods in the future, including document and text embedding techniques and fine-tuning large language models to 19th-century German.

A.5 Dataset description

A.5.1 Construction of the dataset

The Chronicling Germany dataset includes 17 newspapers from all over Germany (Figure: 6). Most of the data (581 pages) is from the Kölnische Zeitung but we have added 112 pages from other newspapers covering the time period 1785-1891 (Table 1). Currently, these 17 newspapers are completly in the test set. However, in an updated version we will also include different newspapers into the training pipeline. Overall, we have annotated 688 pages including over 2.9 million words and almost 30,000 region polygons. Aside from availability and representation, we selected these newspapers for the following reasons: Our focus is on 1866, the year of the Austro-Prussian War. Aside from its historical importance as the second of the three unification wars and the decisive turning point towards the "Kleindeutsche Lösung", this year gives certain advantages to make our data more diverse: During this year, most newspapers across Germany reported lists of the fallen, missing and deserted as well as reports on military careers of officers. These are usually printed in a considerably different layout, thus diversifying our data. Additionally, most states - particularly during wartime - obliged newspapers in their territory to publish "Öffentliche Bekanntmachungen" or official notices. In 1866, all states handled this differently, resulting in more diverse newspaper layouts across Germany (compared to 1871). Also , focusing on this year allows users of our dataset to evaluate separately how well a model generalizes to different newspapers of the same time and how well it generalizes to newspapers from other decades. When no newspapers from 1866 were available, we sometimes included an issue from 1867. The different newspapers have been chosen to maximize variation between them. We include newspapers from various regions of (past) Germany, like Berlin, Eastern Prussia, the Rhineland, Lower Germany (Hannover), Bavaria, The Palatinate, and Saxony. We also take care to include larger national as well as regional newspapers, and newspapers with a special non-political interst, like the Neue Berliner Musikzeitung (New Berlin Musics Paper) and Der Bazar (a paper on "women's topics" - mostly written by men). We dedicated extra attention to the Vossische Zeitung, because it is one of the most-read newspapers of its age and - due to its bad printing quality - it is particularly difficult for layout detection and OCR. The amount of pages per newspaper varies, since we include full issues of each newspaper, regardless of their length. This is done to ensure that the entire diversity in layout and font across different sections of the newspaper is represented in the dataset. The years from 1924 onwards constitutes a natural end for the dataset, since German newspapers gradually started using latina fonts instead of Fraktur during that period.

A.5.2 Annotation process

The annotation process follows the annotation guidlenes in A.6 . A human domain expert carried out all layout annotations that were then cross-checked by another human domain expert. Annotating and correcting text is extremely time-consuming. For the moment an automatic transcription is checked and corrected by a single human domain expert. Currently, we have corrected 446 pages in of the

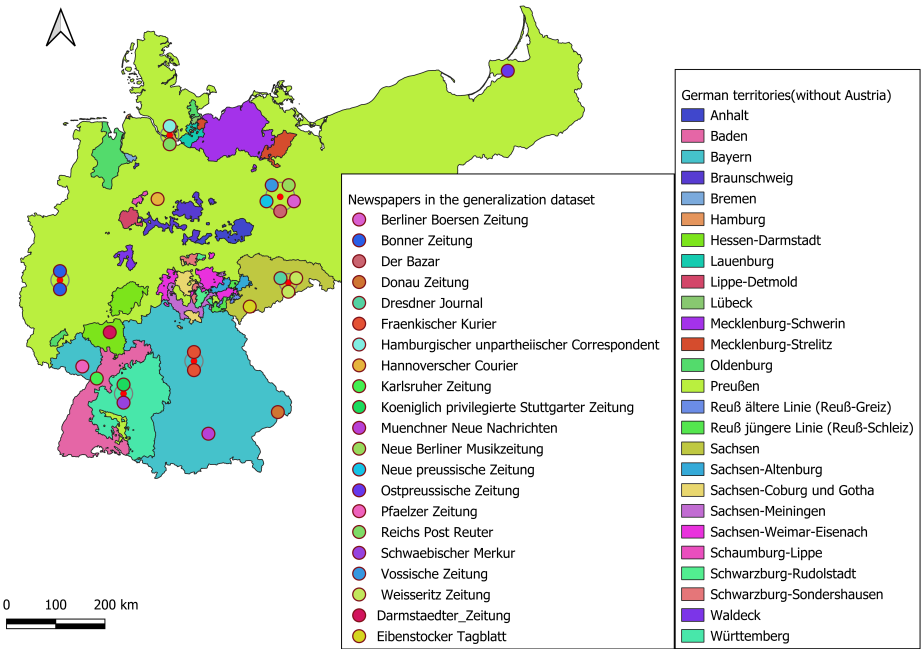


Figure 6: Map of historic Germany from 1867 with labeled regions and regions of newspaper origins beyond the Kölnische Zeitung.

Kölnische Zeitung, and 112 pages in the generalization part of the dataset. To improve quality further we will run a second correction round, where all lines will again by proofread by different annotators.

A.6 Annotation Guidelines

A.6.1 Introduction

These annotation guidelines are an adaptation of the OCR-D rules (`https://ocr-d.de/en/gt-guidelines/trans/transkription.html`). We outline additional rules, we created to ensure consistency of the *Chronicling Germany* dataset.

A.6.2 Page types and type area

The OCR-D guidelines provide for a distinction to be made between page types and the type area during layout analysis. The type area usually contains the text body, but not elements such as the page number. In the *Chronicling Germany* data set, these steps are currently not taken into account.

A.6.3 Regions

Region-types The OCR-D guidelines distinguish between different types of regions, such as text, image and separator regions. In the Bonn Newspaper dataset, the regions are generally recorded in accordance with OCR-D page region level 1 (`https://ocr-d.de/de/gt-guidelines/trans/ly_level_1_5.html`). However, tables are also recorded as a separate region and no distinction is made between images and drawings; instead, all images, photos, illustrations and drawings are grouped together under the GraphicRegion. The entire contiguous region is always marked as a block. For text regions, this applies to contiguous blocks of the same class.

- `TextRegion`: All texts that are not tables. Table headings are not marked as a text region.
- `TableRegion`: All parts of the page that contain tabular information. These are often, but not always, clearly recognizable as tables by small separators. Text that is only separated by separators does not count as a table, but a structure must be recognizable that assigns

certain meanings to rows and columns. Table headings are included with corresponding tables.

- `SeparatorRegion`: All dividing lines are marked as SeparatorRegion. This also includes decorative elements that, like other separator lines, separate areas from each other and are not purely cosmetic in nature. The separators are divided into vertical and horizontal separators and marked with "separator_vertical" and "separator_ horizontal".
- `GraphicRegion`: All graphics, images, photos, illustrations, and drawings.

A.6.4 TEXTREGION SUBTYPES

TextRegions are divided into different subtypes. The subdivision corresponds to the OCR-D guideline for text regions (`https://ocr-d.de/de/gt-guidelines/trans/lytextregion.html#textregionen__textregion_`). However, drop capitals are treated differently from OCR-D. These are counted as part of the paragraph instead of being marked as a separate text region so that models trained on this data will include them in the correct position in their text output. In addition, headlines (caption) and inverted text (inverted-text) are also recorded in the *Chronicling Germany* data set. Instead of annotating advertisements separately, the classes created for other newspaper pages are applied to the advertisements as far as possible. Because headlines should be visually identified, this leads to a large number of text in the advertisements marked as headlines, which contradicts a semantic definition of a headline. Therefore, it makes sense to treat these pages separately in practice and not differentiate between headings and other text. The following elements from the OCR-D guidelines are not represented in the *Chronicling Germany* dataset due to lack of occurrence:
page-number, marginalia, footnote, signature-mark, catch-word, floating, TOC-entry

We discuss the definition for the text subclasses below:

- **paragraph:** Standard text type that includes paragraphs. These are usually kept compact to accommodate as much text as possible in the available space. If a text region cannot be assigned to any other type, it falls under the paragraph label.
- **heading:** Headings that can be clearly distinguished visually from the rest of the text. This is achieved by using a significantly larger or bold font and centered text, which is clearly different from the block layout of paragraphs. A heading is located above a paragraph and is sometimes separated from the previous text by a separator. A thin separator between the heading and the text can occur. However, if there is too much space between them or a thick separator, the two texts no longer count as belonging together in the sense of heading and paragraph. If a text is not superordinate to a paragraph, it cannot be a heading.
- **header:** Page or column titles that appear prominently above the entire page. These are centered at the top of the page and can appear in different font sizes.
- **caption:** Title lines that are located to the right and left of a page heading or text heading. They often contain information such as the date.
- **inverted-text:** Text that is printed white on black. This is often part of decorative elements but is not marked as a graphic element.

A.6.5 OCR

A prerequisite for text recognition is baseline or text-line recognition. Both the baseline itself and a polygon around the text line are annotated. These are generated automatically and only corrected if the baseline connects non-contiguous text passages. Lines that have been divided into two baselines are not corrected. Tables and inverted text are not given baselines.

The text is corrected according to its optical appearance. What is written on the page is transcribed, even if there are errors in the print or scan. Completely illegible passages are not transcribed, passage that are largly illegible are transcribed but marked with the "unknown" tag of Transkibus. Thes passage are not used in training.

The transcription is carried out according to level 2 of the OCR-D guidelines (`https://ocr-d.de/en/gt-guidelines/trans/level_2_2.html`). This includes the transcription of special characters such as the 'long s' (U+017F) or long hyphens (U+2014, em dash). Consistency

with the rest of the data is important here. As these were generated automatically, it is best to look for another example and adopt that version if the special characters are unclear.

Unlike in the OCR-D guideline, fractions are not transcribed with special characters. Instead, the fraction is represented with a slash:
$1\frac{3}{4}$ = 1 3/4.
In this case, it is important to separate the whole number from the fraction with a space. The same applies to times with an underscore. Example for clock times: 11_{45} =11_45 or $11._{45}$ =11._ 45. (For both, use non-breaking spaces in future (U+202F))

The case of a number that is followed by a unit (e.g. 100M) is not dissolved in the OCR-D guidelines. We always add a space between the number and the unit (e.g. 100M becomes: 100 M)

Transkribus allows the selection of special characters with a virtual keyboard. However, it must be ensured that the character used is unique. For example, U+2014 and U+2015 are visually indistinguishable. U+2014 must be used for long hyphens. If the characters are unclear, the OCR-D guidelines, which include tables for the use of special characters, can also be consulted:

- `https://ocr-d.de/en/gt-guidelines/trans/trLigaturen2.html`
- `https://ocr-d.de/en/gt-guidelines/trans/trFremdsprache.html`
- `https://ocr-d.de/en/gt-guidelines/trans/ocr_d_koordinationsgremium_codierung.html`
- `https://ocr-d.de/en/gt-guidelines/trans/trBeispiele.html`
- `https://ocr-d.de/en/gt-guidelines/trans/tr_level_1_3.html`
- `https://ocr-d.de/en/gt-guidelines/trans/trAnfZeichen.html`
- `https://ocr-d.de/en/gt-guidelines/trans/trGedankenstrich.html`

A.7 Further Results

A.7.1 Out of Distribution Layout results

Table 7: Layout detection results on the out of distribution test set. This table lists F1 Score values for all individual classes. N/A values for Dell et al. (2024) are due to annotation differences. Since YOLOv8 detects bounding boxes, we do not require seperator detection. (Main results for layout in Table 4)

	F1 Score (ood only)		
class	ours	YOLOv8	Dell et al.
background	0.78 ± 0.011	0.716 ± 0.013	0.74
caption	0.44 ± 0.005	0.112 ± 0.021	N/A
table	0.31 ± 0.039	0.285 ± 0.042	0.21
paragraph	0.91 ± 0.006	0.768 ± 0.028	0.86
heading	0.41 ± 0.014	0.405 ± 0.021	0.42
header	0.18 ± 0.034	0.164 ± 0.027	0.1
separator vertical	0.38 ± 0.036	N/A	N/A
separator horizontal	0.40 ± 0.024	N/A	N/A
image	0.1 ± 0.022	0.117 ± 0.050	N/A
inverted text	0.15 ± 0.01	0.000 ± 0.000	N/A

A.7.2 YOLOv8 Layout results

Table 8: Layout detection results. This table lists F1 Score values for all individual classes form a YOLOv8 detection model trained on our dataset. Since YOLOv8 detects bounding boxes, we do not require seperator detection.

class	F1 Score (id + ood)	F1 Score (id only)	F1 Score (ood only)
background	0.786 ± 0.009	0.830 ± 0.008	0.716 ± 0.013
caption	0.575 ± 0.005	0.947 ± 0.002	0.112 ± 0.021
table	0.696 ± 0.037	0.881 ± 0.007	0.285 ± 0.042
paragraph	0.859 ± 0.013	0.916 ± 0.007	0.768 ± 0.028
heading	0.673 ± 0.007	0.827 ± 0.014	0.405 ± 0.021
header	0.622 ± 0.050	0.842 ± 0.071	0.164 ± 0.027
image	0.178 ± 0.036	0.503 ± 0.191	0.117 ± 0.050
inverted text	0.157 ± 0.016	0.420 ± 0.078	0.000 ± 0.000

A.7.3 Pipeline OCR results

Table 9: Optical Character Recognition (OCR) results for running the entire pipeline on our dataset. Levenshtein distance per character appears in the first column. We computed the percentage of completely error-free lines for each model. The second column lists these results. Finally, we consider a line to have many errors if we observe a Levenshtein distance of more than 0.1 per character. We report the percentage of many error lines in the final column. All predicted and ground truth lines are matched based on the intersection over the minimum of the corresponding text lines.

Model	Data	Levenshtein-Distance	fully correct [%]	many errors [%]
Complete Pipeline	id + ood	0.03	49.9	21.8
	id only	0.03	63.1	9.9
	ood only	0.06	34.6	35.4