
# Converting scanned documents to TEI XML

Vaishali Burge
(Matr.No: 221202802)
*vburge@uni-koblenz.de*

Harshani Reddy
(Matr.No: 221202816)
*harshani@uni-koblenz.de*

Jyothsna Kalyanaraman
(Matr.No: 221202825)
*jyothsnakalyan@uni-koblenz.de*

April 12, 2024

Under the Guidance Of: Dr. Jens Dörpinghaus
Mathematical Institute, University of Koblenz, Germany
Federal Institute for Vocational Education and Training (BIBB), Bonn, Germany

**Abstract**

This research addresses the crucial task of information retrieval from historical documents, specifically those from the 19th century. Optical Character Recognition (OCR) technology serves as a foundation for this process, enabling the conversion of scanned documents into machine-readable text format. A prominent OCR solution, Tesseract, was employed in this study due to its capability to recognize a vast array of languages. This facilitated the extraction of valuable information and metadata from the digitized historical materials. Furthermore, standardized OCR-D processors were utilized to convert the raw data from its original PDF format into a structured TEI XML format. This transformation ensures the confidentiality and controlled access of the historical data for future research endeavors and analytical purposes. The implemented methodology offers a systematic approach to information gathering and retrieval from historical documents. This approach aligns with the broader efforts of digitalization and archival preservation, fostering continued exploration and understanding of historical records.

# Contents


**1 Introduction** **1**

**2 Background and Related Work** **2**
2.1 Importance of Digitizing Historical Documents . . . . . 2
2.2 Challenges of Digitizing Historical Documents . . . . . 3
2.3 Benefits of TEI XML for Historical Documents . . . . . 3
2.4 TEI XML Applications in Digital Humanities . . . . . 3
2.5 Previous Digitization Efforts in Germany . . . . . 4
2.5.1 PHASE 1: Insights from a User Survey . . . . . 4
2.5.2 PHASE 2: Module Project . . . . . 4

**3 Approach and Implementation** **5**
3.1 Role of OCR Technology . . . . . 5
3.2 Specific OCR Tools and Workflow . . . . . 6
3.3 TEI XML Markup . . . . . 6
3.4 TEI Elements and Metadata . . . . . 7
3.5 Structural Analysis and Metadata Incorporation . . . . . 7
3.6 Implementation . . . . . 7
3.6.1 Process Flow . . . . . 7
3.7 Image Acquisition . . . . . 8
3.7.1 Ensuring Optimal Pixel Density . . . . . 8
3.7.2 Handling Multi-Page Images following OCR-D Specification 8
3.8 Pre-processing . . . . . 9
3.9 Page and Line Segmentation . . . . . 10
3.10 Dewarping . . . . . 11
3.11 Text Recognition . . . . . 11
3.12 hOCR Conversion . . . . . 11
3.13 TEI XML File . . . . . 11
3.13.1 Algorithm of the code . . . . . 12

**4 Results** **13**
4.1 Processing Stages . . . . . 13
4.1.1 Preprocessing . . . . . 13
4.1.2 Image Acquisition . . . . . 13
4.1.3 Page and Line Segmentation . . . . . 14
4.1.4 Text Recognition . . . . . 14
4.1.5 METS File to TEI XML Conversion . . . . . 15

**5 Conclusion** **15**

**6 Limitations and Future Work** **16**
6.1 Limitation of Tesseract: . . . . . 16
6.2 Identifying Certain Letters . . . . . 16
6.3 Future Work . . . . . 17

# List of Figures


1 Process Flow . . . . . 8
2 Document 141739. . . . . 9
3 Document edelmetallpruefer_1938_pruefungsanforderungen. . . . 10
4 Preprocessing Techniques: (a) Original Scan (b) After Binarization 14
5 Outputs of Page and Line Segmentation Process . . . . . 14
6 TEI XML Output of single page . . . . . 15
7 Sometimes identifying "ch" as "&lt" . . . . . 17
8 Error in identifying letters "Berleg" as "Verlag" and "Druck" as "Deu&gt" . . . . . 17

# 1 Introduction

Large-scale digitization of historical documents is a major issue in the field of digital humanities. This is particularly true for materials originating from the 1930s, a pivotal era in German history characterized by a wealth of documents reflecting crucial cultural, political, and social developments. So far, only a small fraction of these sources have been made accessible and considered for research [1]. Modern text recognition technology struggles to read older texts. Even though these programs are trained on a lot of modern fonts, they can't handle the different fonts used in the past. This is especially true for 19th-century documents, which often used fancy and uncommon fonts. Because of this, commercial software for reading old documents often makes mistakes (especially for the multitude of highly variable broken blackletter typefaces)[2]. Transforming these scanned documents into structured, machine-readable formats such as TEI XML necessitates a meticulous and well-defined methodology to ensure efficiency.

Optical Character Recognition (OCR) technology automates the conversion of captured images of text, both handwritten and printed, into a machine-readable format [3]. Despite its advancements since its 19th-century inception, OCR remains a cornerstone in the digitization workflow, enabling the conversion of scanned text into a searchable and editable format. (citation required)While OCR engines like Tesseract offer valuable support for German documents, achieving accuracy remains a challenge, especially when dealing with historical fonts and the specificities of the language. *Our research question centres on investigating how can we efficiently generate TEI XML from scans of German documents from the 1930s.* Specifically, we aim to extract old German documents from the 1930s using OCR processor technology while converting the XML (the final XML file generated by the processor, which has lines and paragraphs) files to hOCR and generating TEI XML, which extracts metadata such as headings, paragraphs, footnotes, and publication places from documents.

Once the OCR process, including pre-processing and post-corrections, is finished, the resulting text can be transformed to hOCR (HTML-based OCR) format. After completing this intermediate step, the text has a structured representation that may be further processed for formatting and analysis. The next step is to convert the data from the hOCR format to the TEI XML structure. This process involves mapping the structural elements identified in the hOCR output to their corresponding elements defined in the TEI XML format. In order to provide crucial context and background information, metadata such as author information, publication date, and source details may also be included in the TEI header. The transformation process extends beyond the generation of plain text.Adding meaningful structure to raw text through TEI XML involves a series of procedures. When it comes to recognizing and identifying parts such as headings, paragraphs, and footnotes, structural analysis is essential. Moreover, special components like tables, drawings, and handwritten portions require special consideration when modifying TEI encoding. Ultimately, adding

rich metadata to the TEI header improves the contextual comprehension of the scanned pages.

In summary, the efficient generation of TEI XML from scanned German documents from the 1930s demands a rigorous and systematic approach encompassing OCR, error correction, structural analysis, TEI markup, and metadata inclusion. This process not only preserves the integrity of historical documents but also facilitates their accessibility and analysis within the digital realm of scholarship and research in the digital humanities domain. The solution we present here will serve as the groundwork for a large-scale information retrieval system from historical documents. These documents, specifically focusing on job postings from the late 19th century, hold rich historical data. By converting them into TEI XML, we aim to create a structured and searchable resource that empowers researchers to delve into this valuable information.

This paper is divided into six sections. The first section introduces the challenges of digitizing historical documents from the 1930s and outlines our research question. The second section reviews related work on OCR technology, TEI XML markup, and previous digitization efforts. In the third section, we detail our methodology for digitization, OCR, and TEI XML markup. The fourth section presents our results. The fifth section defines our results. Finally, the sixth section discusses limitations and suggests future research directions.

# 2 Background and Related Work

This section explores the importance of digitizing historical documents for research and preservation. We then delve into the challenges of this process, including degraded text and limitations of OCR. To overcome these hurdles, we next go through and examine the benefits of TEI XML for structured and searchable digital documents. Additionally, We will examine how TEI XML is currently utilized in digital humanities research, and then Finally, we'll review previous digitization efforts in Germany to gain insights into past obstacles and potential future directions.

## 2.1 Importance of Digitizing Historical Documents

Historical and archival documents serve as windows into our past [4]. They contain essential information about our past, cultural background, and evolution. However, since they are frequently preserved in physical form, these records are at risk for destruction from time and external factors. Digitization preserves these records for future generations by protecting them from possible loss. Additionally, it improves usability by providing these materials easily accessible with a few clicks by anyone, wherever in the entire world. This helps linguistic, anthropological, and historical study in addition to democratizing information [4]. Governmental organizations have a strong reason to keep lots of old paper documents that used to be available only in paper form. Documents from the

military, government, and other sources provide a wealth of examples and use cases.

## 2.2 Challenges of Digitizing Historical Documents

Digitizing historical documents is challenging because of issues such as degraded text, different handwriting styles, ancient scripts, and the need for top-notch scanning quality. Although Calmalari provided a possible starting point, we faced some challenges with recognizing the font. On the other hand, Tesseract, a well-respected model that succeeds at dealing with German papers and fonts, generated positive results. Still, Tesseract showed its drawbacks regarding general accuracy, sensitivity to document quality, and font identification. These limitations reflect well-researched problems OCR technology has faced, especially when handling old documents. Processing papers with damaged text, such as faded ink or cracking paper is one of these obstacles.

## 2.3 Benefits of TEI XML for Historical Documents

Traditionally, museum objects come with well-organized descriptive information (metadata) in catalog entries. However, historical documents like letters, diaries, and travelogues are often unstructured and lack organization. These primary source documents are typically written in continuous prose, sometimes fragments. TEI provides a solid framework for developing metadata organization and management solutions for these documents [5].

XML offers several advantages for handling historical documents. Firstly, it prioritizes descriptive markup, making the content readily understandable by humans without programming expertise. Secondly, XML separates validity from correctness. Documents can be well-formed but lack proper meaning or structure. TEI, a specific set of XML rules for historical documents, ensures data adheres to defined standards, enhancing accuracy and consistency. In conclusion, XML's platform independence ensures that historical documents remain accessible and usable with different tools and systems, ensuring their preservation for the future. These combined strengths make XML, particularly when used with TEI, a powerful solution for managing and preserving historical documents in a structured and enduring way [6].

## 2.4 TEI XML Applications in Digital Humanities

Humanities research increasingly relies on digital tools. Researchers need ways to use their archives, organize collections, and share reusable materials in different formats. However, existing tools can be complex, making it difficult for non-programmers to manage historical documents [7]. This limits the general public's ability to access crucial resources. To address this difficulty, our project makes use of Python's features to generate an approachable outcome. Although we've utilized Python code to automate work effectively, users can interact with

it without any prior knowledge of programming. We've created an easy-to-follow workflow that leads people through the procedure.

## 2.5 Previous Digitization Efforts in Germany

In a 2016 survey conducted by the BSB on the OCR-D project website, 139 participants, mainly humanities scholars, were asked about their usage of OCR texts. While not all questions were answered, some responses were deemed unusable [8].

### 2.5.1 PHASE 1: Insights from a User Survey

The summary from the Phase 1 is [8] :

- A user survey revealed the widespread use of OCR texts in humanities research.
- Researchers employ them for searching large datasets and conducting text analysis.
- Interestingly, 60% tolerate errors in OCR texts. Historians, in particular, value them for finding information.
- While OCR texts are helpful for initial citations, librarians still prefer original images. Opinions on versioning OCR texts are mixed, as researchers acknowledge its importance but are divided on needing access to past versions

### 2.5.2 PHASE 2: Module Project

The insights from the Module Project are [8]:

- Image Preprocessing (DFKI): Enhanced tools for cropping, deskewing, and dewarping to improve image quality for OCR.
- Layout Recognition (DFKI): Extracted document structure for efficient text recognition, including text/non-text segmentation and block classification.
- Layout Analysis and Segmentation (U Würzburg): Developed a tool using Convolutional Neural Networks (CNNs) to separate text from images and segment pages.
- OCR Post-correction (U Leipzig): Implemented a system combining neural networks and finite-state transducers to improve OCR accuracy.
- Tesseract Integration (U Mannheim): Integrated Tesseract OCR engine into the OCR-D workflow and enhanced its stability, code quality, and performance.

- Automatic Post-correction (U Munich): Created a system for automatic post-correction of historical OCR outputs, with an optional interactive correction tool.
- Font Recognition and Model Repository (U Leipzig et al.): Developed tools for automatic font recognition and built a repository of font-specific OCR models for improved accuracy.
- Long-term Archiving (SUB Göttingen et al.): Created a prototype for long-term archiving and persistent identification of OCR results for historical prints.

# 3 Approach and Implementation

As discussed in the introduction, our research focuses on retrieving information from historical documents, particularly those from the 19th century. We rely on Optical Character Recognition (OCR) technology to convert scanned documents into text format. We efficiently extract important information and metadata from digitized historical materials using Tesseract, a widely used OCR tool known for recognizing various languages. Additionally, we employ standardized OCR-D processors and Python scripts to convert the raw data from TIFF format into structured TEI XML. This conversion ensures that historical data is kept secure and can be accessed for future research and analysis. Our method offers a systematic approach to gathering and retrieving information from historical documents, supporting digitalization and archival preservation efforts while enabling further exploration and understanding of historical records in the digital humanities field. The research [9] explores the evolution of techniques, tools, and trends in historical document processing over the past twenty years, with a specific focus on the last decade. It delves into various approaches to digitizing historical handwritten balance sheets.

- Manual input into a spreadsheet is the least technically demanding approach for digitizing historical documents, but it is slow and lacks scalability.
- Using off-the-shelf commercial OCR software reduces manual review time by about 60%, but it is not feasible for large-scale projects due to extensive manual work [9].
- Combining OCR with pre- and post-processing steps is the recommended approach for large-scale datasets, significantly reducing the time required for digitization and improving data quality.

## 3.1 Role of OCR Technology

From the above approaches, we chose to utilize Optical Character Recognition (OCR) technology as it plays a crucial role in converting scanned text

images into machine-readable format. By applying advanced algorithms and techniques, OCR software can accurately identify and extract the textual content from digital images, enabling the conversion of printed or handwritten documents into editable and searchable digital text. This process is essential for various applications, such as document digitization, text mining, and information retrieval, as it allows for the efficient processing and analysis of large volumes of textual data [9].

To address the challenges of 19th-century historical documents as discussed in the Background and Related Works section, This research has adopted OCR Technology. We aim to achieve a high degree of accuracy in text extraction, enabling further analysis and preservation of this valuable historical record.

### 3.2 Specific OCR Tools and Workflow

The proposed approach will employ a combination of specialized OCR tools and techniques tailored to the characteristics of the historical documents under investigation [10]:

- Preprocessing: Binarization using the Sauvola-MS-Split algorithm and Image cropping to remove unnecessary margins.
- Denoising and Deskewing: Enhancing the textual content quality and improving subsequent processing steps accuracy.
- Text Segmentation: Identify and extract the textual content at the line level using the OCR-D-Tesseract-Segment tool.
- Text Dewarping: Straighten and normalize the text using the OCR-D-CIS-Ocropy-Dewarp tool.
- Text Recognition: Leverage the Fraktur+deu model to accurately identify the textual content, taking into account the language-specific characteristics of the historical documents using the OCR-D-Tesseract-Recognize tool.

By employing this tailored workflow of OCR tools and techniques, the proposed approach addresses the challenges of historical fonts and language specificities. This ensures accurate and reliable text extraction from the scanned images of historical documents.

### 3.3 TEI XML Markup

Text Encoding Initiative (TEI) XML is a widely adopted standard for the digital representation and encoding of textual data [11]. TEI XML provides a comprehensive set of guidelines and a flexible markup language that allows for the structured and semantic encoding of various types of textual materials, including historical documents, literary works, and scholarly editions [6].

## 3.4 TEI Elements and Metadata

The TEI XML markup process for the scanned German documents will involve the inclusion of a range of TEI elements and metadata to capture the rich textual and contextual information [6]:

- Structural Elements: , <p>, <head>, to represent the logical structure of the text as sections, paragraphs, and headings respectively.
- Bibliographic Metadata: <teiHeader>, <titleStmt>,<publicationStmt>, <sourceDesc>to provide comprehensive metadata about the document, including title, author, publication details, and source information.
- Named Entities: <persName>, <orgName>, <placeName>to identify and mark up relevant named entities, such as personal names, organizations, and locations.

By incorporating this set of TEI XML elements and metadata, the proposed approach will ensure the comprehensive and structured representation of the scanned German documents, enabling more effective preservation, search, and analysis of the textual content.

## 3.5 Structural Analysis and Metadata Incorporation

Following text extraction with OCR, a custom Python script analyzes the document structure. This script leverages BeautifulSoup to process the initial Tesseract output (typically an XML file with recognized text and layout information).

The script first converts the Tesseract file to hOCR format, which is better suited for structural analysis. By analyzing hOCR data (text and layout coordinates), the script identifies and encodes various document elements like headings, paragraphs, and lists.

Additionally, the script attempts to extract metadata (publication date, place, title, organization) directly from the scanned document. Users are prompted to provide any missing information. This enriched TEI XML output not only captures the text but also provides valuable information about the document's structure and origin, facilitating future research and analysis.

## 3.6 Implementation

### 3.6.1 Process Flow

In this part of the Implementation section let's look at the process flow in detail, see Fig. 1

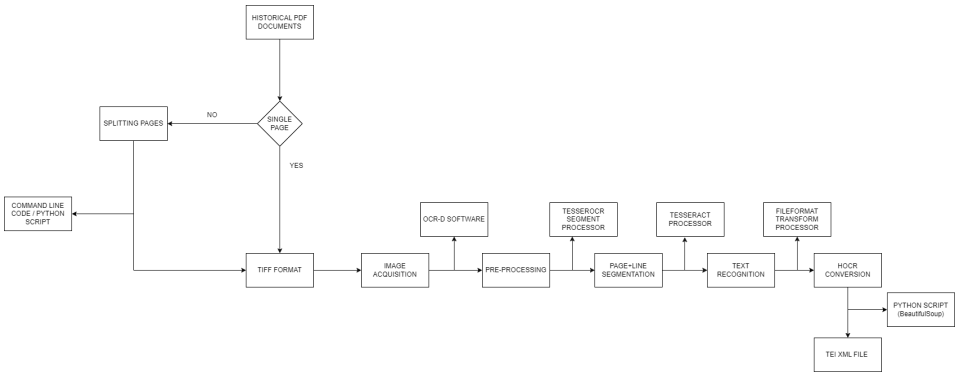



**Figure 1:** Process Flow

## 3.7 Image Acquisition

The initial phase of the workflow involves preparing the images for subsequent processing. Since there are PDF formats of historical documents, each possessing unique structural characteristics, several pre-processing considerations must be meticulously addressed before commencing document processing.

### 3.7.1 Ensuring Optimal Pixel Density

The original input photos must have a pixel resolution higher than 150 ppi. It's essential to ensure that the density is explicitly adjusted during the digitalization process for any procedure that generates new images or modifies their size. If images lack adequate pixel density metadata, processors should assume a resolution of 300 ppi [12].

### 3.7.2 Handling Multi-Page Images following OCR-D Specification

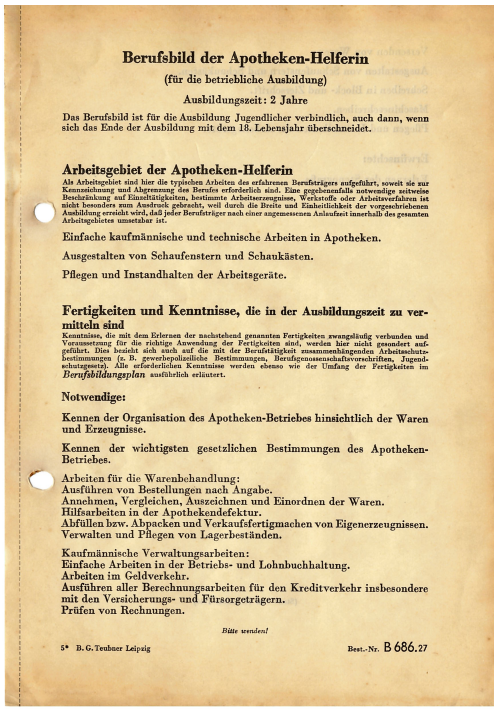
The OCR-D specification mandates that data providers must supply single-image TIFF files, and OCR-D processors must raise exceptions if they encounter multi-image TIFF files [12].

To adhere to the OCR-D specification, users must provide single-image TIFF files. However, if the images are initially provided in other formats, such as PDF, conversion to TIFF format is necessary. This can be achieved using Python code or available online tools. Once converted, users can provide the single-page TIFF images for further processing. If the original image is multi-page, splitting of images is required as OCR-D processors cannot handle multiple pages simultaneously. For image splitting, users can execute the following command via the command line `convert combined_image.jpg -crop 50%x100% +repage page%d.jpg` or Python code provided within the GitHub repository.

## 3.8 Pre-processing

The first step is to prepare the images for processing by setting up a workspace. This involves a series of standard procedures, including image acquisition, quality assessment, and formatting adjustments, to ensure the images are ready for further analysis and manipulation [13]. Once the necessary software, including dependencies and `ocrd_all`, is installed, the next step is to set up a workspace for processing the files.

Next, we employ processors to pre-process the images, following a streamlined and definitive workflow that covers essential pre-processing steps. Depending on the structure of the document, the steps differ; for example, a well-defined document like **141739** (see Fig. 2) requires fewer processing steps, while a document like **edelmetallpruefer_1938_pruefungsanforderungen** (see Fig. 3) necessitates more pre-processing due to its complexity. The document **141739** (Fig. 2) below is simple and well-defined, requiring minimal preprocessing, and does not require steps such as deskewing or denoising.



**Berufsbild der Apotheken-Helferin**

(für die betriebliche Ausbildung)

Ausbildungszeit: 2 Jahre

Das Berufsbild ist für die Ausbildung Jugendlicher verbindlich, auch dann, wenn sich das Ende der Ausbildung mit dem 18. Lebensjahr überschneidet.

**Arbeitsgebiet der Apotheken-Helferin**

Als Arbeitsgebiet sind hier die typischen Arbeiten des erfahrenen Berufsträgers aufgeführt, soweit sie zur Kennzeichnung und Abgrenzung des Berufes erforderlich sind. Eine gegebenenfalls notwendige zeitweise Beschränkung auf Einzeltätigkeiten, bestimmte Arbeitserzeugnisse, Werkstoffe oder Arbeitsverfahren ist nicht besonders zum Ausdruck gebracht, weil durch die Breite und Einheitlichkeit der vorgeschriebenen Ausbildung erreicht wird, daß jeder Berufsträger nach einer angemessenen Anlaufzeit innerhalb des gesamten Arbeitsgebietes umsetzbar ist.

Einfache kaufmännische und technische Arbeiten in Apotheken.

Ausgestalten von Schaufenstern und Schaukästen.

Pflegen und Instandhalten der Arbeitsgeräte.

**Fertigkeiten und Kenntnisse, die in der Ausbildungszeit zu vermitteln sind**

Kenntnisse, die mit dem Erlernen der nachstehend genannten Fertigkeiten zwangläufig verbunden und Voraussetzung für die richtige Anwendung der Fertigkeiten sind, werden hier nicht gesondert aufgeführt. Dies bezieht sich auch auf die mit der Berufstätigkeit zusammenhängenden Arbeitsschutzbestimmungen (z. B. gewerbepolizeiliche Bestimmungen, Berufsgenossenschaftsvorschriften, Jugendschutzgesetz). Alle erforderlichen Kenntnisse werden ebenso wie der Umfang der Fertigkeiten im *Berufsbildungsplan* ausführlich erläutert.

**Notwendige:**

Kennen der Organisation des Apotheken-Betriebes hinsichtlich der Waren und Erzeugnisse.

Kennen der wichtigsten gesetzlichen Bestimmungen des Apotheken-Betriebes.

Arbeiten für die Warenbehandlung:
Ausführen von Bestellungen nach Angabe.
Annehmen, Vergleichen, Auszeichnen und Einordnen der Waren.
Hilfsarbeiten in der Apothekendefektur.
Abfüllen bzw. Abpacken und Verkaufsfertigmachen von Eigenerzeugnissen.
Verwalten und Pflegen von Lagerbeständen.

Kaufmännische Verwaltungsarbeiten:
Einfache Arbeiten in der Betriebs- und Lohnbuchhaltung.
Arbeiten im Geldverkehr.
Ausführen aller Berechnungsarbeiten für den Kreditverkehr insbesondere mit den Versicherungs- und Fürsorgeträgern.
Prüfen von Rechnungen.

*Bitte wenden!*

5* B. G. Teubner Leipzig Best.-Nr. B 686.27

**Figure 2:** Document 141739.

- **ocrd-preprocess-image**: This processor helps with enhancing your images, which can be vital for the following binarization. In this processing step, the raw image is taken and enhanced by e.g. grayscale conversion, brightness normalization, noise filtering [10].

- **ocrd-anybaseocr-crop**: Employing parameters such as 'marginTop', 'marginBottom', 'marginLeft', and 'marginRiretainsthis processor crops the image to eliminate extraneous margins and retain relevant content [10].

However, documents like **edelmetallpruefer_1938_pruefungsanforderungen** (Fig. 3) depicted in the figure below, require all preprocessing steps due to unclear images and challenging font recognition.

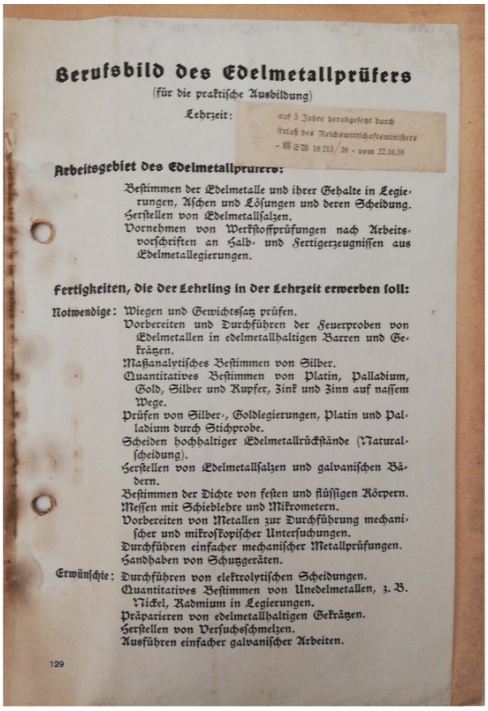



**Figure 3:** Document edelmetallpruefer_1938_pruefungsanforderungen.

In addition to the previously mentioned pre-processing steps, extra steps like denoising and deskewing are also necessary.

- **ocrd-skimage-denoise**: This processor is used to denoise the image, removing any noise or artifacts that could negatively impact OCR accuracy [10].
- **ocrd-tesserocr-deskew**: This processor deskews the image, correcting any skew or rotation present in the text [10].

## 3.9 Page and Line Segmentation

Following pre-processing steps, layout analysis segments the document image. This stage involves:

- **Region Segmentation:** Dividing the image into distinct areas containing text, graphics, tables, etc.
- **Line Segmentation:** Separating text regions into individual lines for accurate Optical Character Recognition (OCR) processing.

Layout analysis also corrects image distortions like skewing or rotation. The `ocrd-tesserocr-segment` processor efficiently handles both page and line segmentation simultaneously.

### 3.10 Dewarping

The next step after segmentation is Dewarping, In this processing step, the text line images get vertically aligned if they are curved [10].

### 3.11 Text Recognition

This marks the concluding phase of document processing. Here, the OCR processor examines segmented text regions to transform text into a machine-readable format. It identifies characters, words, and sentences, considering font styles and image quality. The usual result is a Mets file containing the recognized text content. There are two processors available for this purpose:

- `ocrd-calamari-recognize`
- `ocrd-tesserocr-recognize`

We have utilized the **ocrd-tesserocr-recognize** processor. This model is favored over Calamari due to its efficient use of internal Tesseract iterators, resulting in more accurate text recognition, particularly in complex layouts or degraded images. This enhancement significantly improves overall OCR performance [10].

### 3.12 hOCR Conversion

In this part of the process, the generated Mets file is converted to an hOCR file using the processor **ocrd-fileformat-transform**.

hOCR files contain detailed information about the location and attributes of recognized text elements, facilitating post-processing and analysis. In contrast, METS files primarily serve as metadata containers for organizing and describing digital objects, lacking the detailed text information present in hOCR files. Therefore, for OCR tasks focusing on text extraction and analysis, hOCR files offer better usability and efficiency compared to METS files.

### 3.13 TEI XML File

In this section, we discuss the algorithm for converting hOCR to TEI XML files. Our Python script leverages BeautifulSoup for parsing hOCR files containing

OCR results. By extracting specific text elements using bounding box span IDs as text identifiers and organizing them based on position and format within the document, we construct a TEI XML structure.

The TEI XML output captures metadata such as title, organization name, place, and date, along with structured content like headings, paragraphs, and lists. This standardized format facilitates archival and further document analysis.

### 3.13.1 Algorithm of the code

This algorithm outlines the steps to process an hOCR file and generate a TEI XML document. **The Steps are:**

- **Import necessary libraries:**
  - BeautifulSoup for HTML/XML parsing.
  - "re" for regular expressions.
- **Read the hOCR file:** Open and read the hOCR file's content.
- **Parse the hOCR content:** Use BeautifulSoup to parse the hOCR content (HTML-based).
- **Extract key information:**
  - Extract title text, date text, place names, and organization names from specific bounding boxes.
- **Extract text elements:** Retrieve text from different divs with specific bounding boxes.
- **Construct the TEI XML document:**
  - Define the TEI XML structure with extracted data such as title, author, publication information, and text body with paragraphs and headings.
  - Incorporate the extracted text into the appropriate TEI elements.
- **Write the TEI XML document to a file:** Save the constructed TEI XML document content to an output file.

# 4 Results

The conversion pipeline successfully transforms scanned PDF documents into TEI XML format, facilitating advanced analysis and manipulation of document data. Subsequent subsections detail the results obtained at each stage of the conversion process, accompanied by visual representations of intermediate outputs.

## 4.1 Processing Stages

The conversion process consists of several key stages:

### 4.1.1 Preprocessing

The preprocessing stage is essential for accurate text recognition in scanned PDFs. It involves segmenting multi-page PDFs into individual pages and converting them to compatible formats, typically TIFF. Various image processing techniques like binarization and cropping are then applied to enhance image quality and facilitate precise text extraction. Optionally, deskewing techniques may be used to correct skewed content. By customizing these steps, the images are optimized for efficient and accurate text recognition

### 4.1.2 Image Acquisition

The Image acquisition phases play a critical role in ensuring accurate text recognition and analysis. TIFF images serve as the foundation for subsequent processing stages, and these stages rely heavily on high-quality images. After image acquisition and conversion to TIFF format, we selectively apply preprocessing techniques like binarization, see Fig. 4b cropping, and deskewing as needed to enhance image quality. We tailor these techniques to the characteristics of each document, ensuring optimal conditions for accurate text recognition in subsequent stages of the conversion pipeline.


**(a)** Original Image **(b)** Binarization

**Figure 4:** Preprocessing Techniques: (a) Original Scan (b) After Binarization

### 4.1.3 Page and Line Segmentation

OCR-D processors segmented these pre-processed images into individual pages and lines, generating output in the METS format. In the example you see, there are many lines on the page. However, to simplify things, we're only showing three lines here, see Fig. 5.


**Figure 5:** Outputs of Page and Line Segmentation Process

### 4.1.4 Text Recognition

Following page and line segmentation, the Tesseract OCR engine extracts text from the segmented images. The recognized text is stored in a METS file, which contains structured metadata.

### 4.1.5 METS File to TEI XML Conversion

The generated METS file is converted to hOCR format and then parsed by Python code to produce the final TEI XML file, see Fig. 6. The div tags encompass data from subsequent pages, although they are currently hidden to provide an overview.

```
<TEI>
  <teiHeader>
    <fileDesc>
      <titleStmt>
        <title>BERUFSAUSBILDUNG IN DER INDUSTRIEJugendlichen-Ausbildung</title>
        <author>
          <orgName>B.G.Teubner</orgName>
        </author>
      </titleStmt>
      <publicationStmt>
        <publPlace>Leipzig and Berlin</publPlace>
        <date>25. März 1942</date>
      </publicationStmt>
    </fileDesc>
  </teiHeader>
  <text>
    <body>
      <div>
        <!-- Heading -->
        <head>Berufseignungsanforderungen</head>
        <!-- Subtext -->
        <head>für den Eintritt in den Anlernberuf</head>
        <!-- Heading -->
        <head>Asphaltabdichter</head>
        <p>bearbeitet vom</p>
        <p>Reichsinstitut für Berufsausbildung in Handel und GewerbeGemeinschaftsorgan der
           Organisation der gewerblichen Wirtschaftund der Deutschen Arbeitsfront</p>
        <p>Im Sinne der Erlasse des Reichswirtschaftsministersvom 13. 2. 1939 und 5. 7. 1939im
        Einvernehmen mit derArbeitsgemeinschaft der Industrie- und Handelskammernin
        der Reichswirtschaftskammerfür die Ausbildung in industriellen Betriebenam 8. September 1942
        anerkannt durch dieReichsgruppe Industrie</p>
        <p>Stand vom 25. März 1942</p>
        <p>Verlag und Druck von B.G.Teubner in Leipzig und Berlin</p>
        <p>Best.Nr. E 661.11</p>
      </div>
      <div>
      <div>
      <div>
    </body>
  </text>
</TEI>
```

**Figure 6:** TEI XML Output of single page

# 5 Conclusion

OCR-D offers a powerful framework for processing historical documents. While current implementations address many challenges, ongoing efforts to enhance character recognition accuracy, broaden processor options, and refine workflows will ensure continued improvement in the quality and efficiency of historical document digitization.

Through OCR-D, we have successfully processed images and converted them into structured formats like hOCR and TEI XML, marking a significant advancement in document digitization. However, minor issues persist, such as character identification accuracy, indicating areas for improvement. To address these challenges, further experimentation and optimization efforts are needed.

By conducting additional experiments and refining algorithms, we can improve character identification accuracy and optimize processing pipelines. These efforts will contribute to enhancing the overall performance and reliability of OCR-D workflows.

These developments not only enhance document processing skills but also open up exciting possibilities for information retrieval and analysis in a variety of fields. OCR-D has the potential to significantly progress document digitization and open the door to more effective and accessible digital libraries and archives with further testing and development.

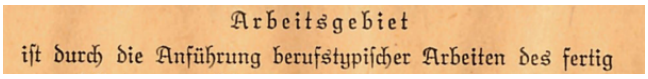
# 6 Limitations and Future Work

## 6.1 Limitation of Tesseract:

- **Preprocessing Dependency:** Tesseract necessitates meticulous preprocessing for optimal performance, which is often challenging due to varying image quality.
- **Scanned Images:** Less effective with scanned documents due to artifacts and skewed text.
- **Complex Layouts:** Struggles with intricate layouts, multi-column text, and unconventional arrangements.
- **Handwriting Recognition:** Tailored for printed text, thus challenging for handwritten content.
- **Language and Fonts:** Performance fluctuations with less common languages and fonts.
- **Gibberish Output:** May generate gibberish, affecting data accuracy [14].

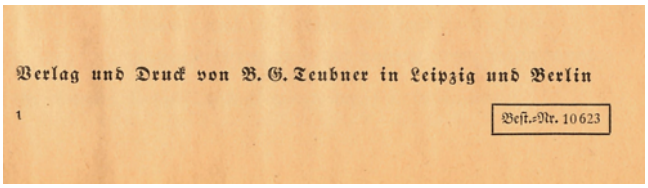
## 6.2 Identifying Certain Letters

Such as the letter "ch" fonts are being identified as "&alt" and some are interpreted wrongly when applied to the Tesseract model hOCR conversions.




```
</span>ift dur&lt; die Anführung berufstypifcher Arbeiten des fertig</span>
```

**Figure 7:** Sometimes identifying "ch" as "&lt"




```
></span>Verlag und Deu&gt; von B. G., Teubner in Leipzig und Berlin</span>
```

**Figure 8:** Error in identifying letters "Berleg" as "Verlag" and "Druck" as "Deu&gt"

## 6.3 Future Work

- Enhancing Text Recognition Accuracy: Refining OCR algorithms, exploring advanced preprocessing techniques, and incorporating domain-specific knowledge can improve recognition accuracy, especially for complex layouts and degraded text.
- Pipeline Automation and Integration: Streamlining the pipeline's automation and integration is a goal. Efforts will target automating multi-page PDF handling and improving table recognition algorithms. These enhancements will not only reduce manual work but also significantly improve workflow efficiency.
- Advanced Preprocessing and Domain-Specific Knowledge: Exploring advanced pre-processing techniques and incorporating domain-specific knowledge can further enhance recognition accuracy.
- Page Number Detection: Detecting page numbers presented a challenge, as accurately identifying and extracting them proved difficult. This limitation affects document indexing and retrieval, disrupting the sequential organization of pages.

# References


[1] David Fleischhacker, Roman Kern, and Wolfgang Göderle. Improving ocr quality in 19th century historical documents using a combined machine learning based approach.

[2] Uwe Springmann. The ethics of digital editing in the humanities. *Digital Humanities Quarterly*, 11(2), 2014.

[3] Herbert F. Schantz. *The History of OCR, Optical Character Recognition*. Recognition Technologies Users Association, Manchester Center, Vt., 1982.

[4] CharacTell. The role of ocr in digitizing historical and archival documents. https://www.charactell.com/resources/the-role-of-ocr-in-digitizing-historical-and-archival-documents/, unknown.

[5] Félix Rodríguez Moure and Sergio Escalera Romero. Towards the automatic creation of textual critical editions. *Journal of the Text Encoding Initiative*, 10, 2019.

[6] TEI Consortium. TEI P5: Guidelines for Electronic Text Encoding and Interchange. https://tei-c.org/release/doc/tei-p5-doc/en/html/SG.html.

[7] Floriane Chiffoleau. Keeping it open: a tei-based publication pipeline for historical documents. https://hal.science/hal-04357295.

[8] Ocr-d user survey. https://ocr-d.de/en/user_survey.

[9] Tabrizi Nasseh Philips, James. Historical document processing: A survey of techniques, tools, and trends. *cornell university*, 2020.

[10] ocrd. ocrd workflow guide. https://ocr-d.de/en/workflows.

[11] TEI Consortium. Projects Using the TEI. https://tei-c.org/release/doc/tei-p5-doc/en/html/SG.html.

[12] Ocr-d specification. https://ocr-d.de/en/spec/mets.

[13] OCR-D. OCR-D setup guide. https://ocr-d.de/en/setup.

[14] Docsumo. Introduction to tesseract ocr.