## Methode

Wissensgraphen können auf unterschiedliche Arte und Weise erstellt werden. In der Praxis sind die häufigste Variante: manuelle Erstellung, Erstellung aus (semi)strukturierten und unstrukturierten Quellen (vgl. Heist et al. Ege/Paschke 2021: 129). Außerdem gewährleisten neuronale Modelle des Textverstehens wie Large Language Models (LLM) ein großes Potential für die Wissenserfassung (vg. Hamann Hinkelmann/Hoppe/Humm 2025: 68).

Bei der manuellen Erstellung von Wissensgraphen werden die Fakten und Entitäten auf Grundlage von Erfahrung einer oder eines Experter:in definiert. Dies sichert die hohe Qualität des Wissens, aber dieser Ansatz ist zeitaufwendig und arbeitsintensiv (vg. Hamann Hinkelmann/Hoppe/Humm 2025: 74).

Eine Mischform aus manuellem Ansatz und KI-Modellen könnte ein optimaler Ansatz sein. Mithilfe von LLM wird das Korpus nach Entitäten und Fakten durchgesucht, während der manuelle Ansatz zu der Sicherstellung aller relevanten und qualitativ hochwertigen Informationen beiträgt (vg. Hamann Hinkelmann/Hoppe/Humm 2025: 74).

Im Vergleich zu dem manuellen Ansatz hat ein automatisierter Ansatz mehrere Vorteile. Er ist kosteneffizient, skalierbar, flexibel und aktuell. Ein Nachteil des automatisierten Ansatzes ist die Notwendigkeit von ausreichende annotierte Daten für das Modelltraining (vg. Hamann Hinkelmann/Hoppe/Humm 2025: 75).

Die Umsetzung der Implementierung erfolgte in Python unter Verwendung der Standardbibliotheken SpaCy, ….

## Arbeitsmaterial

Das Arbeitsmaterial umfasst insgesamt 20 Lebensbeschreibungen[[1]](#footnote-1) aus Periodika der Herrnhuter Brüdergemeine. Die *Nachrichten aus der Brüdergemeine* (NBG) wurden zwischen 1819–1894 im Druck veröffentlicht und sind als Primärdigitalisate im digitalen Archiv der [Memorial University of Newfoundland](https://dai.mun.ca/digital/nachrichten/) zugänglich. Die Auswahl des geeigneten Materials beruhte auf mehreren Prämissen: Einerseits mussten die betreffenden Personen in Südafrika leben und wirken, andererseits sollten diese Menschen in ihren Lebensläufen andere Personen erwähnen. Die Digitalisate wurden mit Python-Tesseract für Optical Character Recognition (OCR) erschlossen. Darüber hinaus dient eine Tabelle als Arbeitsmaterial, die alle Personennamen sowie deren Geburts- und Sterbeorte enthält, basierend auf NBD. Diese Tabelle wurde von der SLUB bereitgestellt.

## OCR-Erkennung

Das gesamte Material wurde per Python-Bibliothek *pytesseract* erfasst. Open-Source *Tesseract* verwendet die Sprachmodelle für die Textextraktion (vgl. Baierer/Zumstein 2016: 73). Die *pytesseract*-Bibliothek von Tesseract-OCR Engine von Google und unterstützt viele Bildformate (vgl. [Python-tesseract](https://pypi.org/project/pytesseract/)).

Je nach verwendeter Methode[[2]](#footnote-2) zeigt OCR-Erkennung unterschiedliche Ergebnisse. Um die Verbesserung dieser Ergebnisse zu erzielen, wurden die Bilder zuerst vorbereitet (Abbildung 5.1). Dies erfolgt mithilfe einer Python-Bibliothek *OpenCV.*

Ein Bild, das Text, Schrift, Screenshot enthält.

KI-generierte Inhalte können fehlerhaft sein.

Abb. 5.1

Zunächst wurden alle JPG-Dateien invertiert, indem die Helligkeitswerte jedes Pixels umgekehrt wurde, sodass helle Pixel dunkel und dunkle Pixel hell werden. Dies erhöht den Kontrast zwischen dem Text und dem Hintergrund. Um verschiedene Grauabstufungen darzustellen, wurde auf die invertierten Bildern Grayscaling angewendet. Dieser Schritt reduziert die Komplexität des Bildes und erleichtert die Extraktion, da es den Fokus auf die Helligkeitswerte legt. Ein Computer kann normalerweise bis zu 256 Graustufen darstellen (vgl. Rawat/ Sharma/ Gusain 2021: 2590). Anschließend wurde das Graustufenbild in ein binäres Bild umgewandelt. Der Binarisierung erfolgt durch Festlegung eines Schwellenwerts. So kann ein hoher Wert bspw. Zeichenstriche entfernen, während ein zu niedriger Wert zu unscharfen Randbereichen führen kann (vgl. ebd.: 2590). Die [CV2](https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html)-Bibliothek bietet verschiedene Typen von Binarisierungsmethoden zur Auswahl. Nach verschiedenen Erprobungen mit unterschiedlichen Einstellungen wurde der am besten passenden Schwellenwert ermittelt und für die THRESH\_BINARY-Methode Entschieden. Bei dieser Methode wurden alle Pixel mit einer Helligkeit über 120 auf weiß 255 gesetzt, während Pixel mit niedriger Helligkeit schwarz 0 wurden. Diese Einstellungen erwiesen sich bei allen Bildern als besonders effektiv und liefern gute Resultate. Zum Schluss wurde der Kontrast verstärk und dadurch wurde Text deutlicher hervorgehoben (vgl. [OpenCV](https://docs.opencv.org/4.x/d7/d1b/group__imgproc__misc.html#ggaa9e58d2860d4afa658ef70a9b1115576a147222a96556ebc1d948b372bcd7ac59)). Danach kam OCR zum Einsatz (Abbildung 5.2). Dabei wurden zwei Sprachmodelle kombiniert: Deutsch und Fraktur.



Abb. 5.2

Nach der OCR-Erschließung wurden OCR-Fehler analysiert, um mögliche Korrekturschritte abzuleiten. Es wurde festgestellt, dass der größte Teil von Fehler auf falsch erkannten Zeichen liegt, z.B.: *ic), auc), nad, cr, ſc<wer* usw.anstatt *ich, auch, nach, er, ſchwer*. Diese Fehler wurden automatisch durch die korrekten Zeichen ersetzt. Außerdem wurden Silbentrennung am Zeilenende und überflüssige Zeichnen im Text entfernt.

## TEI-XML- Modellierung

Für die vorlegende Arbeit wurde XML nach den Rechtlinien der TEI als semantischen Ebene eingesetzt. Dabei dient es zu der Erfassung der Metadaten von Primärdigitalisaten, zu der Abbildung der Textstruktur (Seitenumbrüche, Überschriften, Absätze, Sätze) und zu der inhaltlichen Annotation (Personen-, Orts-, Körperschaftsnamen, Datumsangaben).

Zu Beginn wurden die Primärdigitalisate analysiert, um festzustellen, welche Informationen in den Annotationen erforderlich oder optional sind. Dabei wurden strukturelle Elemente, ihre Relevanz ermittelt und in welcher Form sie erfasst werden können. Im Fokus stand die Sicherung der Metadaten und des Layouts jedes Digitalisats. Darüber hinaus wurde geprüft, in welchem Umfang die Entitäten erfasst werden sollten.

Im nächsten Schritt wurde das Vokabular des XML-Schemas definiert. Dazu dienten die Richtlinien der TEI, insbesondere [TEI-Header](https://tei-c.org/release/doc/tei-p5-doc/en/html/HD.html) und [Namen, Daten, Personen und Orten](https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html). Um eine standardisierte und einheitliche Darstellung aller Daten in den XML-Dokumenten sicherzustellen, wurden die Datentypen festgelegt und sämtliche Elemente und Attribute definiert, die in den XML-Dateien verwendet werden. Hierbei war entscheidend, nicht nur die ausgewählten Elemente zu berücksichtigen, sondern auch die exakte Reihenfolge der Elemente genau einzuhalten. Dabei wurden die Elemente mit festgelegter Reihenfolge und ohne unterschieden. Genauso wurde mit den Attributen vorgegangen. Im Schema wurde festgelegt, ob ein Attribut obligatorisch ist und ob ein Attribut ein festgelegter Wert hat. Schließlich wurde das Schema (rng-Datei) in *Oxygen* Editor auf Basis einer vormodellierten XML-Datei erstellt.

Jedes TEI-Dokument folgt einer bestimmten Grundstruktur: <teiHeader> und <text>. Im ersten Teil befindet sich die Information über die physische Transkription, die über das <teiHeader>-Element kodiert wurde. Das Element fasst Metadaten zur elektronischen Ressource um, wie die Information über die Verfasser:innen der Primärquelle, die Sprache der Ressource, verantwortliche Personen usw. Der zweite Teil der XML-Datei fokussiert sich auf den Textbereich und nimmt Bezug auf die digitale Repräsentation. Dabei orientiert sich das <text>-Element an der Inhaltslogik des Textes.

Das oberste Element <teiHeader> enthält die Metadaten eines Digitalisats in einem Hauptbereich: <fileDesc> (Dateibeschreibung) (Abbildung 5.3). Das <fileDesc>-Element umfasst drei obligatorische Elemente: <titleStmt>, <publicationStmt> und <sourceDesc>, in denen die Angaben zum Titel, zur Publikation und Quelle kodiert wurden.

Ein Bild, das Text, Diagramm, Screenshot, Quittung enthält.

KI-generierte Inhalte können fehlerhaft sein.

Abb. 5.3

Das oberste <text>-Element umschließt den gesamten Text eines Digitalisats und enthält eine Hierarchie von weiteren Elementen (Abbildung 5.4). Dadurch wird die Struktur und Semantik eines Textes definiert. Das untergeordnete <body>-Element enthält die eigentlichen Textabschnitte. Der Textteil einer XML-Datei ist relativ überschaubar, wobei das <div>-Element den gesamten Text umfasst.

Ein Bild, das Text, Screenshot enthält.

KI-generierte Inhalte können fehlerhaft sein.

Abb. 5.4

Seiten wurden durch das <p>-Element kodiert. Seitennummern wurden als n-Attribut zu dem <p>-Element zugewiesen. Der Titel jeder Lebensbeschreibung wurde im <head>-Element eingeschlossen. Die Kustode wurde den TEI-Richtlinien als patextuelle Formelemente im Element <fw> ausgezeichnet. Für die Erfassung von Personen-, Orts-, Körperschaftsnamen wurde über das <persName>-, <placeName>, <orgName>- und <date>-Element kodiert.

## Informationsextraktion aus Texten

Die Informationsextraktion besteht aus mehrere Teilschritten: Named Entity Recognition (NER) und Linking (NEL), Kohärenzauflösung, Extraktion zeitlicher Informationen, Relationsextraktion (RE). Die Umsetzung der Implementierung erfolgte in Python unter Verwendung der Standardbibliotheken SpaCy, ….

In Bezug auf der Informationsextraktion aus unstrukturierten Daten können unterschiedliche Kategorie von Datentypen identifiziert und extrahiert werden. Man spricht dabei über Entitäten, Beziehungen, Adjektiven und ergänzende Informationen (vgl. Lanquillon/Schacht 2023: 163).

### 5.4.1 Named Entity Recognition (SpaCy, ML, generative LLM)

Entitäten sind hauptsächlich Subjekte und Objekte in einem Text. Die meinst bekannteste sind Named Entities wie Personen, Orte, Firmen usw. Je nach Forschungsfrage können auch andere Subjekte wie Krankheiten, Medikamente usw. als Entitäten bezeichnet werden (vgl. Lanquillon/Schacht 2023: 163).

Eine wichtige Komponente bei der Wissensextraktion im Bereich von Text-Mining ist Named Entity Recognition (NER) (vg. Hamann Hinkelmann/ Hoppe/Humm 2025: 81).

Durch ein NER-Tagger werden Entitäten in einem Text identifiziert und markiert. Er verwendet dabei generische Typen wie *Person* oder *Firma* (vg. Hamann Hinkelmann/Hoppe/Humm 2025: 81).

Bei NER treten verschiedene Schwierigkeiten. Ein Problem besteht darin, ob die einzelnen Wörter im Text zusammen oder getrennt betrachtet werden sollen (*New Yourk*). Ein weiteres Problem ist die Mehrdeutigkeit von Wörtern (*Washington*). Bei solchen Wörtern kann schwierig sein, die richtige Entitätsklasse zu bestimmen (vgl. Lanquillon/Schacht 2023: 172).

Der erste Schritt betrifft die Rohdaten. Das Ziel der Vorerarbeitung (ML-Training) ist es, …

## Relationsextraktion (Ollama)

Relationsextraktion verfolgt das Ziel, die erkannte Entitäten im Text miteinander zu verknüpfen (vg. Hamann Hinkelmann/ Hoppe/Humm 2025: 84). Beziehungen treten als Verben auf und bilden die syntaktische und semantische Verbindung zwischen Entitäten. Sie können mithilfe regelbasierter Verfahren identifiziert werden. Normalerweise werden die Beziehungen allgemein geschrieben: *ist Chef von, ist angestellt bei* (vgl. Lanquillon/ Schacht 2023: 165).

Dadurch werden Tripeln direkt gebildet (vg. Hamann Hinkelmann/ Hoppe/Humm 2025: 84)

Damit Relationsklassifikation effektiv funktioniert, müssen die Entitäten in kurzen Textabschnitten gemeinsam erwähnt werden (vg. Hamann Hinkelmann/ Hoppe/Humm 2025: 85).

## 5.3 Datenabgleich

Kohärenzauflösung

Während der Datenvorverarbeiten wurden die Texte in Sätze untergeteilt.

Pronomen oder Umschreibungen von Entitäten können Informationen im Text verschleiern, wenn der Kontext nicht berücksichtigt wird.

Die automatische Zuordnung von Entitäten zu einem Graph erfolgt durch Named Entity Linking (NEL) (vg. Hamann Hinkelmann/Hoppe/Humm 2025: 83)

Da die Maschinen keine indirekten Bedeutungen erkennen können, daher müssen Relationen zwischen Informationen eindeutig sein. Normdaten wie [GND-Nummer](https://www.dnb.de/DE/Professionell/Standardisierung/GND/gnd_node.html) (Gemeinsame Normdaten) ermöglichen diese Eindeutigkeit. Wird eine Person über GND-Nummer identifiziert, können Inhalte automatisch mit bspw. [Wikipedia](https://www.wikipedia.de/)-Eintrag oder [Deutschen Biographie](https://www.deutsche-biographie.de/) verknüpft werden. Trotzt der Bemühungen der Deutschen Nationalbibliothek kann die Eindeutigkeit von Normdaten nicht in allen Fällen gewährgeleistet werden. Regelmäßig auftretende Probleme, wie etwa fehlerhafte oder doppelt vorhandene GND-Einträge, lassen sich häufig auf die dezentrale Struktur der Datenerfassung zurückführen, die im Rahmen kooperativer Arbeitsprozesse. In der Praxis wird aus Effizienzgründen oder unter Zeitdruck nicht selten darauf verzichtet, bereits bestehende Datensätze auf ihre Richtigkeit und Vollständigkeit hin zu prüfen. Stattdessen wird ein neuer Eintrag erzeugt, was wiederum zu Redundanzen sowie zu einer Verringerung der Datenkonsistenz und -qualität im Gesamtsystem führen kann. Die Erstellung und Pflege von Normdaten stellt auch die Herausforderung im Bereich von Körperschaften, geographische Orte, Werte, Periodika usw. Ihre eindeutige Identifikation erweist sich oftmals als deutlich komplexer. Insbesondere bei Orten treten spezifische Schwierigkeiten auf: Während georeferenzierende Systeme (GIS) in der Regel lediglich punktuelle Koordinaten erfassen, bleiben flächenhafte Ausdehnungen sowie historische Veränderungen unberücksichtigt. Die historische Dimension eines Ortes – etwa sich verändernde politische Zugehörigkeiten, Ortsnamen oder Grenzverläufe – kann daher nur unzureichend über bestehende textbasierte Normdaten erfasst werden und bedarf ergänzender, oft nicht-textueller Quellen (vgl. Kamzelak 2018: 426-427).

## 5.5 RDF

## 5.6 OWL (Lanquillon/Schacht 2023, S. 187)

## 5.7 Turtle

## 5.8 Wikidata-Datenmodell

Laube-Artikel

Zur Auffindbarkeit und technischen Interoperabilität von Projektdaten ist es sinnvoll, etablierte Identifikatoren und Normdaten zu verwenden. Im Rahmen dieser Arbeit wurde Wikidata genutzt, um externe Konzepte und Realitäten referenziert und gemeinsamen Bezugsrahmen zu schaffen.

* 1. Zusammenfassung

## SPARQL

Laube-Artikel

## Visualisierung

Netzwerkanalyse biete neue Möglichkeit zur Visualisierung und Analyse von Sammlungswissen. Ihre zentralen Ziele sind die Identifikation wichtiger Verbindungen und Strukturen. (vgl. Werner 2020: 248–249).

Für geistes- und sozialwissenschaftliche Forschung ist die Verbindung zwischen Netzwerken und Visualisierung von großer Bedeutung. Die graphische Visualisierung von abstrakten oder komplexen Strukturen macht die Zusammenhänge, die sich auf rein sprachlichem Wege nur schwer oder unzureichend vermitteln lassen, greifbar. Digitale Werkzeuge bieten hier neue Möglichkeiten, große Datenbestände nicht nur zu verarbeiten, sondern auch visuell aussagekräftige Formen überführen können. Daraus ergibt sich einen Anknüpfungspunk für die Verwendung netzwerkbasierter Modelle. Für die biographische Forschung eröffnet der Netzwerkansatz die Perspektive, individuelle Lebensläufe nicht isoliert zu betrachten, sondern als Teil größer Beziehungsgeflechte (vgl. Dörpinghaus 2022).

## Auswertung / Perspektive

## Zusammenfassung

## Literaturverzeichnis

## Anhang

1. Anhang 2 [↑](#footnote-ref-1)
2. Kommerzielle OCR-Software wie *ABBYY* setzt auf einen Wörterbuchabgleich, während Open-Source-Programme wie *Tesseract* und *Ocropus* auf unterschiedliche Sprachmodelle oder neuronale Netzen zurückzugreifen (vgl. Baierer/Zumstein 2016: 73). [↑](#footnote-ref-2)