

# Introduction to Digital History

Ina Serif



# Table of contents

<b>Welcome</b>	<b>1</b>
<b>1 Was ist Digital History?</b>	<b>3</b>
<b>2 Forschung und Lehre</b>	<b>5</b>
2.1 Digitale Tools zur Analyse . . . . .	7
2.2 Digitale Tools zur Kommunikation . . . . .	9
2.3 Digitale Tools in der Hochschullehre . . . . .	9
2.4 Projekte und Ressourcen . . . . .	9
2.4.1 Alte Geschichte . . . . .	9
2.4.2 Mittelalter und Frühe Neuzeit . . . . .	10
2.4.3 Moderne und Zeitgeschichte . . . . .	10
2.4.4 Jüdische Geschichte . . . . .	11
2.4.5 Geschichte Afrikas . . . . .	11
2.4.6 Osteuropäische Geschichte . . . . .	11
2.4.7 Epochen-/Areaübergreifend: . . . . .	12
<b>3 Digital Literacy, Digital Criticism</b>	<b>13</b>
3.1 Digital Criticism, Data Criticism . . . . .	14
<b>4 Datenerhebung, -aufbereitung und -analyse</b>	<b>19</b>
4.1 Datenaufbereitung . . . . .	24
4.2 Datenanalyse . . . . .	27
4.3 Datensicherung . . . . .	28
<b>5 FAIR und CARE</b>	<b>29</b>
<b>I Praxis</b>	<b>31</b>
<b>6 Briefedition ‘Der Sturm’</b>	<b>35</b>
<b>7 Durch den Vordereingang</b>	<b>39</b>

<b>8</b>	<b>Durch die Hintertür</b>	<b>41</b>
8.1	Strg-F 2.0 . . . . .	43
8.1.1	Erste Analysen . . . . .	46
<b>9</b>	<b>Ausblick</b>	<b>53</b>
	 <b>Appendices</b>	 <b>56</b>
<b>A</b>	<b>Glossar</b>	<b>57</b>
<b>B</b>	<b>Literatur, Tools, Tutorials</b>	<b>59</b>
B.1	Einführungen und Guides . . . . .	59
B.2	Tools für digital history (free/open source) . . . . .	60
B.2.1	Allgemein . . . . .	60
B.2.2	Text-/Korpusanalyse . . . . .	60
B.2.3	Visualisierung . . . . .	60
B.3	Digital Literacy, Digital Criticism . . . . .	61
B.4	Terminal/Command Line/Shell . . . . .	61
B.5	Regular Expressions . . . . .	61
B.6	XML . . . . .	61

# Welcome

Der vorliegende Guide, erstellt im Herbstsemester 2022, begleitet die Einführungskurse im Fach Geschichte an der Universität Basel und soll einen ersten Einblick in den Bereich Digital History geben. Es ist ein *living document*, das regelmäßig aktualisiert wird und dabei auch die epochen- und areaspezifischen Inhalte der verschiedenen Einführungskurse berücksichtigt, indem Verweise auf verschiedene Digital-History-Projekte aus unterschiedlichen Bereichen mit der Zeit in den Guide einfließen. Für die Teilnehmer:innen der Einführungskurse wird der Guide von einer Präsenzsitzung begleitet, bietet aber hoffentlich auch unabhängig davon einen Mehrwert. Für Kommentare, Anregungen oder Beschwerden freue ich mich über eine Nachricht.

Der Guide ist in zwei Teile gegliedert: Die Kapitel 1–5 sollen eine erste Übersicht über Digital History bieten und den Blick auf Neuerungen und Veränderungen richten, die sich in den Geschichtswissenschaften aus der Nutzung digitaler Methoden ergeben. Der anschließende praktische Teil zeigt an einem konkreten Beispiel die Anwendung verschiedener Techniken auf, die sich (nicht nur) für Historiker:innen bei der Arbeit mit Quellenmaterial anbieten. Der Praxisteil verfolgt dabei zwei Ziele: Zum einen sollen Hemmungen bei der Arbeit mit dem Computer, die über die Nutzung als elektronische Schreibmaschine hinausgeht, abgebaut werden. Zum anderen soll ein grundlegendes Verständnis dafür hergestellt werden, welche Möglichkeiten computergestützte Analysen bieten und wie diese in der historischen Arbeit eingesetzt werden können.

Die Übersicht soll möglichst knapp gehalten werden – es gibt zahlreiche ausführliche Grundlagenwerke, weswegen viele Themen nur kurz angeschnitten, dafür aber mit weiterführenden Verweisen versehen werden. Dasselbe gilt für den Praxisteil: Weiterreichende Anleitungen, Tutorials oder Onlinekurse werden an entsprechender Stelle verlinkt. Vollständigkeit wird an keiner Stelle beansprucht; Hinweise auf weitere Online-Angebote nehme ich gerne auf.



# Chapter 1

## Was ist Digital History?

Über die Antwort zur Frage, was Digital History ist oder umfasst, kann man ausgiebig diskutieren. Als Teilgebiet der Digital Humanities, der digitalen Geisteswissenschaften, kann die folgende aktuelle und pragmatische Definition von Blaney et al. (2021) hilfreich sein:

Digital humanities, in our view, is a question of approach: if you are actively and critically using digital tools to aid your work in researching, teaching or learning, you are probably doing digital humanities. We would encourage anyone to learn to program if they are interested in doing so, but we do not see it as a defining characteristic of work in digital humanities.<sup>1</sup>

Dabei umfassen “digital tools” eine große Bandbreite – und es wird sich kaum eine:r finden, der:die Studium, Forschung oder Lehre völlig ohne die Nutzung digitaler Techniken betreibt. Wir sind alle Historiker:innen im digitalen Zeitalter, und als solche müssen wir ohnehin neue Kompetenzen entwickeln. Wir können uns aber zudem dafür entscheiden, für ein Forschungsprojekt Methoden und Techniken einzusetzen, die über die traditionellen Werkzeuge der Geschichtswissenschaften hinausgehen – Analyse und Interpretation von Quellen durch deren genaue Lektüre, sogenanntes *close reading* –, und uns durch den Computer unterstützen lassen. Ob wir hierbei auf vorhandene Software zurückgreifen oder selbst Programme schreiben, um uns nicht nur als Historiker:innen im digitalen Zeitalter, sondern auch als digitale Historiker:innen zu verstehen, mögen manche als Glaubensfrage auffassen; eine inkludierende Haltung zu dieser Frage scheint mir dabei nur Vorteile zu haben.<sup>2</sup>

---

<sup>1</sup>Blaney, Jonathan; Winters, Jane; Milligan, Sarah u. a.: Doing digital history: a beginner’s guide to working with text as data, Manchester 2021 (IHR research guides), S. 6.

<sup>2</sup>Entgegen einer häufig zitierten Aussage von Emmanuel Le Roy Ladurie (\*1929), der Historiker von morgen werde Programmierer sein, oder er werde nicht sein: “L’historien de demain sera programmeur ou il ne sera pas.” Le Roy Ladurie, Emmanuel: La fin des érudits, in: Le Nouvel Observateur, 08.1968.

Für eine erste Idee dafür, wie man historische Fragestellungen mithilfe digitaler Methoden beantworten kann und wie unterschiedlich digital unterstützte Forschungsprojekte aussehen können, bietet sich unter anderem der Übersichtsartikel “State of the Field: Digital History” von Romein et al. (2020) an.<sup>3</sup> Eine anwachsende Liste an Beispielprojekten aus unterschiedlichen Epochen bzw. Themenbereichen findet sich unter Projekte und Ressourcen in Kapitel 2.

Um eine Annäherung an die aktive, kritische und reflektierte Nutzung digitaler Methoden in Forschung und Lehre mit einem Fokus auf deren Anwendung in den Geschichtswissenschaften geht es im vorliegenden Guide. Weiterführende Texte zur Frage, was Digital History ist bzw. umfasst, finden sich unter Literatur, Tools, Tutorials

---

<sup>3</sup>Romein, C. Annemieke; Kemman, Max; Birkholz, Julie M. u. a.: State of the Field: Digital History, in: History 105 (365), 04.2020, S. 291–312. Online: <<https://doi.org/10.1111/1468-229X.12969>>, Stand: 15.09.2022.



## Chapter 2

# Forschung und Lehre

Die fortschreitende Digitalisierung in ganz unterschiedlichen Lebensbereichen zieht Veränderungen und Entwicklungen auch für die historische Arbeit nach sich, und dies auf mehreren Ebenen: in Bezug auf die Arbeit bzw. den Umgang mit Quellen, hinsichtlich des Einsatzes digitaler Methoden nicht nur zur Analyse von Forschungsergebnissen, sondern auch für deren Kommunikation, und schließlich für die Hochschullehre.

Als Historiker:innen steht die Arbeit mit Quellen im Mittelpunkt unserer Analysen. Das bedeutet gleichzeitig, dass der Zugang bzw. die Verfügbarkeit von Dokumenten einen Einfluss darauf hat, welche Fragen wir beantworten oder welche Analysen wir vornehmen können. Zugangsbeschränkungen, die die Größe und Zusammensetzung unseres Untersuchungskorpus beeinflussen, können dabei von Gedächtnisinstitutionen – also Museen, Archiven, Bibliotheken – ausgehen, beispielsweise wenn bei zeitgenössischen Akten eine Schutzfrist festgesetzt wird oder wenn ein Objekt zu fragil für die Benutzung ist. Auch kann es aus finanziellen und/oder organisatorischen Gründen schwierig sein, bestimmte Archive an weiter entfernten Orten aufzusuchen, um weitere Dokumente für die Untersuchung zu berücksichtigen. Groß angelegte Digitalisierungsprojekte in Bibliotheken und Archiven bergen damit die Möglichkeit, zusätzliche Quellen nicht nur über einen Eintrag im Bibliothekskatalog zu finden, sondern die entsprechenden Dokumente in digitaler Form auf den eigenen Rechner zu laden. Gerade auch für wertvolle historische Bestände – antike Papyri, Handschriften aus dem Frühmittelalter, einzelüberlieferte Frühdrucke usw. – entsteht hier die Möglichkeit, diese einem größeren Kreis verfügbar zu machen, ohne das Objekt zu großer Belastung durch häufige Benutzung auszusetzen, und ohne dass die Benutzer:innen lange Reisen auf sich nehmen müssten. Für mittelalterliche und frühneuzeitliche Handschriften und Drucke beispielsweise existieren mittlerweile mehrere (meist nationale) Portale, die eine zentrale Suche über alle Bestände ermöglichen; eine Auswahl findet sich unter Section 2.4.2.



Figure 2.1: Randall Munroe, History Department, xkcd.com (17.12.2018).

Neben der Digitalisierung vorhandener Quellen (Retrodigitalisierung) steht die unaufhörliche Entstehung neuer Quellen in rein digitaler Form (*born digital data*). Der relativen Knappheit von Quellen – und damit Daten –, die Vormodernehistoriker:innen oftmals zu beklagen haben, steht eine Überfülle an zeitgenössischem Material gegenüber, und beide Situationen – zu wenig/zu unvollständige und zu viele/zu unübersichtliche Datenmengen – bergen methodische Probleme: Wie stellt man ein Korpus, also eine Sammlung von Quellen zusammen, das ausreichend Dokumente beinhaltet, um Fragestellungen zu beantworten, Thesen zu stützen, neue Erkenntnisse zu erhalten, das aber gleichzeitig in einem Forscher:innenleben bewältigbar bleibt? Historiker:innen müssen neue Kompetenzen erwerben, um mit solchen Fragen reflektiert umzugehen. Zur klassischen Quellenkritik kommt die digitale Quellenkritik, zur Fähigkeit, analoge Quellen zu lesen und zu verstehen, ein Äquivalent für den digitalen Bereich. Etwas ausführlicher geht es in Kapitel 3 um Digital Literacy und Digital Criticism.

## 2.1 Digitale Tools zur Analyse

Die hier bereits zitierte Definition, die aktive und kritische Nutzung digitaler Werkzeuge in Forschung, Lehre oder Studium sei es, was Digital Humanities ausmachen, wirft die Frage auf, was genau unter digitalen Werkzeugen, unter *digital tools* zu verstehen ist, und zu welchem Zweck man sie einsetzt. Allein schon das Lesen dieses Guides ist ohne digitale Hilfsmittel nicht möglich – es existiert kein gedrucktes Exemplar davon. Lesen am Bildschirm allein macht noch keinen *digital humanist*, aber man muss nicht erst eine Programmiersprache lernen, um den Computer für die eigene Arbeit zu nutzen und zu Ergebnissen zu kommen, die mit klassischen Methoden – im Bereich der Geschichtswissenschaften etwa papierbasiertes *close reading* von Quellen und Forschungsliteratur – nicht im selben Ausmaß erzielt werden könnten.

Untersuchungen, die digitale Methoden einsetzen, sind im Normalfall skalierbar – wenn man eine Software benutzt, die die Häufigkeit von Begriffen in einem Dokument zählt, sollte es keinen Unterschied in der Anwendung machen, ob man eines oder einhundert Dokumente auswerten will. Würde man dasselbe per Hand tun, wäre man analog zum Anwachsen der Dokumente mit der Auszählung beschäftigt. Digitale Werkzeuge ermöglichen es also unter anderem, Untersuchungen auf größere Mengen von Dokumenten auszuweiten. Sie ermöglichen es auch, an ein so erweitertes Korpus andere Fragen zu stellen, als dies mit einer kleineren Quellen-/Datengrundlage möglich wäre. Die vorherrschende Überlieferung historischer Quellen besteht aus Text, handgeschrieben, gemeißelt oder gedruckt – und durch die Möglichkeit, diesen mittels Texterkennung in computerlesbare Daten umzuwandeln, ergeben sich neue Perspektiven für die Arbeit von Historiker:innen: Wenn Texte als Daten verstanden werden, lassen sich aus Textquellen Datenbestände erstellen, die mithilfe quantitativer Methoden untersucht und ausgewertet werden können.<sup>1</sup>

---

<sup>1</sup>Kommandozeile/Command Line, Bash, Shell oder Prompt finden sich oft als syn-

Für die Literaturwissenschaften beispielsweise ist ein wichtiges Anwendungsfeld die Überprüfung von Autor:innenschaft: Ob ein anonym überliefertes Werk einem: r namentlich bekannten Autor:in zugeschrieben werden kann, lässt sich entweder durch *close reading* von Literaturwissenschaftler:innen überprüfen, oder durch die Suche nach *patterns*, Mustern, nach quantifizierbare Eigenschaften eines Textes, wie beispielsweise die Häufigkeit von Funktionswörtern, Partikeln, Satzzeichen usw. Der unter dem Pseudonym Robert Galbraith veröffentlichte Kriminalroman *The Cuckoo's Calling* konnte mit entsprechender Software Joanne K. Rowling zugeschrieben werden – damit dauerte die Untersuchung dreißig Minuten, was etwa dem Lesen von zwanzig Romanseiten entspricht. Zu einem Artikel, der diesen Fall thematisiert und in das Feld der linguistischen Forensik einbettet, die Straftäter:innen mithilfe quantitativer Textanalyse ermittelt, geht es hier. Ein Video zur Entwicklung und Anwendung von Software zur Zuschreibung von Autor:innenschaft finden Sie hier. Die genutzte Software, JGGAP,<sup>2</sup> lässt sich offensichtlich auch für historische Analysen nutzen – man denke nur an Herrschaftssysteme, in denen strenge Zensur geübt wird/wurde und viele Autor:innen daher nicht unter ihrem Klarnamen publizier(t)en. Durch eine Identifikation anonymer Schreiber:innen lassen sich weitere Aspekte rund um die Thematik Zensur untersuchen – welche Akteur:innen waren öffentlich bekannt, wer publizierte gleichzeitig anonym und unter Klarnamen, welche Autor:innen schrieben aus dem Exil, welche Netzwerke lassen sich rekonstruieren usw. Dadurch, dass ein Programm durch quantitative Auswertungen die Kärnerarbeit der Identifikation abnehmen *kann* – um einen reflektierten Umgang mit Daten und Algorithmen geht es in Kapitel 3 –, bleibt mehr Zeit für die qualitative Arbeit; gleichzeitig fußt die Analyse auf einem aussagekräftigen Datensatz, anstatt nur Einzelbeispiele beleuchten zu können.

Quantitative und qualitative Methoden sollen hier keinesfalls gegeneinander ausgespielt werden; vielmehr soll verdeutlicht werden, dass beide Herangehensweisen Vor- und Nachteile haben, und dass sie im besten Fall gewinnbringend miteinander kombiniert werden können – quantitative Auswertungen nur um ihrer selbst willen und ohne eine spezifische historische Fragestellung generieren kaum je einen Mehrwert.

Je nach Datengrundlage, Analysezweck und Forschungsfrage bieten sich unterschiedliche Tools zur Nutzung an; für die meisten Forschungsvorhaben bis zum Ende des Studiums dürfte existierende Software ausreichen, sei es für die Akquise und Aufbereitung von Daten(-sätzen), für verschiedene Arten von Textanalysen, statistische Auswertungen, Netzwerkanalysen, Geomapping oder

---

onym genutzte Begriffe für Command Line Interfaces. Auf UNIX-basierten Betriebssystemen wie Mac OS und Linux ist das Terminal als Interface weit verbreitet; für Details: [https://en.wikipedia.org/wiki/Command-line\\_interface#History](https://en.wikipedia.org/wiki/Command-line_interface#History). Windows-nutzer:innen kommen mit der PowerShell ganz gut zurecht, es empfiehlt sich eventuell die Installation von Cygwin oder MinGW, um mit einem UNIX-basierten Interface arbeiten zu können.

<sup>2</sup>Unter [openglam.ch](http://openglam.ch) finden sich Informationen zu Schweizer GLAM-Einrichtungen, die offene Daten anbieten.

Visualisierungen. Eine Auswahl an Tools – alle kostenfrei/*open source* – für spezifische Analysen findet sich unter Literatur, Tools, Tutorials. Für gewisse Analysen bietet es sich an, Programmierkenntnisse zu erwerben – das Erstellen eigener Skripts, also kleiner Programme, beinhaltet die umfassende Kontrolle darüber, wie Daten eingelesen, aufbereitet, angereichert, analysiert und visualisiert werden; bei wiederkehrenden Prozessen, die händisch einige Arbeitszeit in Anspruch nehmen würden, lässt sich so zusätzlich Zeit sparen.

Für geisteswissenschaftliche Projekte werden zurzeit vor allem zwei Programmiersprachen genutzt, R und Python. Da sich beide großer Beliebtheit in den Humanities erfreuen, existieren mittlerweile zahlreiche Packages, die Data und Text Mining, also groß angelegte Daten- und Textanalysen, sehr einfach machen. Solche Packages für Programmiersprachen kann man sich wie Plug-Ins für Programme vorstellen, beispielsweise ein AdBlocker für den Browser. So etwas war von den Entwickler:innen ursprünglich nicht vorgesehen, aber jemand hatte Bedarf, Werbeanzeigen zu blockieren, hat hierzu ein Programm geschrieben und es der Allgemeinheit zur Verfügung gestellt. Der Unterschied zu einem Package ist, dass dieses verschiedene Funktionen zur Verfügung stellt – auswählen und ausführen müssen die Anwender:innen. Wer in Schule und Studium keine Berührungspunkte mit Programmieren hatte, wird zu Beginn vielleicht größere Berührungängste haben – aber noch einmal: Sie müssen nicht programmieren können, um quantitativ zu arbeiten. Speziell an Historiker:innen ohne Programmier-Vorkenntnisse richtet sich das Projekt “The Programming Historian”, das seit 2008 zahlreiche Tutorials veröffentlicht, um verschiedene Tools, Techniken und Workflows für die geschichtswissenschaftliche Forschung und Lehre vorzustellen.

## 2.2 Digitale Tools zur Kommunikation

tbd

## 2.3 Digitale Tools in der Hochschullehre

tbd

## 2.4 Projekte und Ressourcen

### 2.4.1 Alte Geschichte

Projekte:

Ressourcen/Portale:

### 2.4.2 Mittelalter und Frühe Neuzeit

Projekte:

- Repertorium Academicum: Projekt zur Erfassung europäischer Gelehrter zwischen 1250 und 1550

Ressourcen/Portale:

- dMGH: Monumenta Germaniae Historica online (Beta-Version)
- e-codices: Virtuelle Handschriftenbibliothek der Schweiz
- Fragmentarium: Laboratory for Medieval Manuscript Fragments
- Handschriftenportal: Zentraler nationaler Nachweis für Buchhandschriften in deutschen Bibliotheken und in deutscher Sprache (Entwicklungsstadium)
- e-manuscripta: Digitalisierte handschriftliche Quellen aus Schweizer Bibliotheken und Archiven
- e-rara: Plattform für digitalisierte Drucke aus Schweizer Institutionen
- Gallica: Digitalisierte Quellen aus französischen Bibliotheken
- swisscollections: Suchplattform für historische Schweizer Bestände
- transcriptiones: Plattform zum Erstellen, Teilen und Nutzen von Transkriptionen historischer Manuskripte

### 2.4.3 Moderne und Zeitgeschichte

Projekte:

- Refugee History: Wissenschaftliches Blog und interaktives Netzwerk zu aktuellen Debatten um das Thema “Flüchtlinge”

Ressourcen/Portale:

- Datenbank Bild + Ton zur Geschichte (Schweizer) sozialer Bewegungen
- Dodis: Wissenschaftliche Edition von Dokumenten zur Schweizer Außenpolitik
- e-newspaperarchives.ch: Schweizer Zeitungen online
- e-periodica: Schweizer Zeitschriften online
- Historische Statistik der Schweiz (HSSO)
- histat: Zeitreihen zur Historischen Statistik

### 2.4.4 Jüdische Geschichte

Projekte:

- Digital Jewish Studies Online, Stroum Center for Jewish Studies, University of Washington

Ressourcen/Portale:

- Blavatnik Archive: Archiv zur Erhaltung und Verfügbarmachung von Material zur (jüdischen) Geschichte des 20. Jahrhunderts mit Fokus auf die zwei Weltkriege und Sowietrussland.
- Menny, Anna; Rürup, Miriam; Siegel, Björn: Jüdische Geschichte im deutschsprachigen Raum, in: Busse, Laura u. a. (Hg.): Clio-Guide. Ein Handbuch zu digitalen Ressourcen für die Geschichtswissenschaften, Berlin 2018, S. E.2-1–E.2-56. Online: <https://doi.org/10.18452/19244>.

### 2.4.5 Geschichte Afrikas

Projekte:

- Emandulo: Digitales Archiv, das archivalische/museale Sammlungen und Präsentationen über präkoloniale südafrikanische Geschichte zusammenführt und neu zusammenstellt.
- Legacies of British Slavery: Forschungsprojekt zum britischen Sklavenhandel und -besitz

Ressourcen/Portale:

- FHYA: Experimentelle digitale Forschungsplattform über präkoloniale südafrikanische Geschichte
- Legacies of British Slavery: Datenbank zum britischen Sklavenhandel und -besitz
- Slave Voyages: Datenbanken zum transatlantischen und interamerikanischen Sklavenhandel und Personendatenbank

### 2.4.6 Osteuropäische Geschichte

Projekte:

- Gulag: Many Days, Many Lives: Archiv und Präsentationsplattform zu den sowjetischen Gulags
- Gulag Online: Virtuelles Museum mit Präsentationen und Quellen zum Leben im Gulag, zu Personen und Objekten
- Seventeen Moments in Soviet History: Multimediales Online-Archiv mit ausgewählten Quellen zu Ereignissen in der sowjetischen Geschichte anhand 17 verschiedener Jahre zwischen 1917 und 1991

Ressourcen/Portale:

- Blavatnik Archive: Archiv zur Erhaltung und Verfügbarmachung von Material zur (jüdischen) Geschichte des 20. Jahrhunderts mit Fokus auf die zwei Weltkriege und Sowietrussland.
- The Other Side: Webarchiv von Interviews ehemaliger *Ostarbeiter:innen*, Kriegsgefangener und Insassen deutscher Lager; Publikationsplattform

### 2.4.7 Epochen-/Areaübergreifend:

Projekte:

- Lord of the Rings Project: Interaktive Analyse der Werke J. R. R. Tolkiens

Ressourcen/Portale:

- Around DH in 80 days: Portal zur Vorstellung achtzig verschiedener Digital-Humanities-Projekte weltweit und aus verschiedenen Disziplinen
- Internet Archive: digitale Bibliothek zur Archivierung von Büchern, Bildern, Filmen, Software, Musik und Webseiten



## Chapter 3

# Digital Literacy, Digital Criticism

Unter Data Literacy wird die Kompetenz verstanden, Daten zu sammeln, zu managen, zu evaluieren und zu nutzen,<sup>1</sup> eine Kompetenz, die jeder für den mittlerweile unvermeidlichen Umgang mit Daten verschiedenster Art im eigenen Alltag entwickeln sollte. Je nach Forschungsdisziplin ergeben sich weitere gewisse Spezifika, wobei Studierenden der Geisteswissenschaften ein Thema wie Algorithmenkritik nicht als erstes in den Sinn kommt, wenn es um die im Studium zu erwerbenden Kompetenzen geht.<sup>2</sup> Aber auch ohne den Quellcode von *machine-learning*-Software im Detail zu verstehen, ermöglicht ein grundlegendes Verständnis von und ein Wissen über die Funktionsweisen solcher Anwendungen einen reflektierten Umgang mit diesen. Eine solche Art von Digital bzw. Data Literacy ist vor allem dann relevant, wenn es um die Interpretation von Ergebnissen geht, die scheinbar objektiv sind, bzw. scheinbar objektiv entstanden. Ein gutes Beispiel hierfür sind die Ergebnislisten bei Suchanfragen in einer Suchmaschine. Je nachdem, welchen Anbieter Sie nutzen, spielen verschiedene Umstände in die Generierung von Trefferlisten hinein, beispielsweise Ihre Suchhistorie, sodass *search neutrality* nicht mehr gewährleistet ist.<sup>3</sup>

---

<sup>1</sup>Ridsdale, Chantel; Rothwell, James; Smit, Mike u. a.: Strategies and Best Practices for Data Literacy Education Knowledge Synthesis Report, 2015, S. 8. Online: <<https://doi.org/10.13140/RG.2.1.1922.5044>>.

<sup>2</sup>Kommandozeile/Command Line, Bash, Shell oder Prompt finden sich oft als synonym genutzte Begriffe für Command Line Interfaces. Auf UNIX-basierten Betriebssystemen wie Mac OS und Linux ist das Terminal als Interface weit verbreitet; für Details: [https://en.wikipedia.org/wiki/Command-line\\_interface#History](https://en.wikipedia.org/wiki/Command-line_interface#History). Windows-nutzer:innen kommen mit der PowerShell ganz gut zurecht, es empfiehlt sich eventuell die Installation von Cygwin oder MinGW, um mit einem UNIX-basierten Interface arbeiten zu können.

<sup>3</sup>Unter [openglam.ch](http://openglam.ch) finden sich Informationen zu Schweizer GLAM-Einrichtungen, die offene Daten anbieten.

Gehen Sie auf die Bilder-Suche von Google und suchen Sie nach “historian”. Was sehen Sie?

Wüsste ich nichts über Geschichtswissenschaftler:innen, würde ich aufgrund der Ergebnisse meiner Suche davon ausgehen, “a historian” wäre meist ein alter, weißer Mann mit Brille, Bart und einem großen Bücherregal; wenn Sie sich am Departement Geschichte der Uni Basel umsehen, dürfte ein etwas anderer Eindruck entstehen. Die Ergebnisse von Suchmaschinen, die für ihr Funktionieren Algorithmen anwenden, sind *biased*, verzerrt: Sie beruhen auf vorangegangenen Suchen, Vorlieben, geographischem Standort – und auf von Menschen eingegebenen Metadaten, also Daten mit Informationen über andere Daten. Ein Bewusstsein hierfür und das Hinterfragen von Datensätzen gehören also mit zur Arbeit in einer digitalisierten Welt.

### 3.1 Digital Criticism, Data Criticism

Digitalisierte Quellen ebenso wie rein digitale erfordern eine erweiterte Art von Quellenkritik – im Einführungskurs an der Universität Basel lernen Sie die Grundlagen klassischer Quellenkritik:

Woher kommt eine Quelle, wer hat sie unter welchen Umständen und zu welchem Zweck erstellt? Welche Absichten können darin verborgen sein, und welche Verzerrungen können sich durch sie ergeben?

Welche Tendenzen könnten sich in hochmittelalterlichen Herrscherchroniken verstecken, wenn der Verfasser in direkter Abhängigkeit des Auftraggebers stand? Wie sind Zeug:innenaussagen in Hexenprozessen zu bewerten, die unter Anwendung körperlicher Strafe entstanden sind? Mit wie viel Vorsicht sind die Inhalte eines Tagebuchs zu bewerten, das allem Anschein nach mit Blick auf eine spätere Veröffentlichung verfasst wurde?

Neben der inneren Kritik geht es bei der Arbeit mit Quellen immer auch um Fragen der Korpusbildung: Wie kann eine Quellengrundlage erstellt werden, die für Beantwortung einer spezifischen historischen Fragestellung belastbar und aussagekräftig genug ist und gleichzeitig in angemessener Zeit bearbeitet werden kann? Hinzu kommen Spezifika bei der Arbeit mit unterschiedlichen Quellenformen bzw. -formaten: Bei analogen Quellen, die auch in digitaler Form zur Verfügung stehen, besteht die Gefahr, dass ein Thema, ein Bereich, ein Aspekt vernachlässigt wird, wenn nur die unmittelbar verfügbaren, digitalisierten Bestände zur Korpusbildung genutzt werden. Wenn Sie sich beispielsweise für die Schweizer Historikerin und Frauenrechtlerin Meta von Salis (1855–1929) und deren briefliche Korrespondenz – Friedrich Nietzsche war einer ihrer Brieffreunde – interessieren und über die Suchplattform für historische Schweizer Bestände, swisscollections, in nationalen Bibliotheken und Archiven nach entsprechenden Dokumenten suchen, erhalten Sie 361 Treffer:

Digital verfügbar waren hiervon im Oktober 2022 lediglich drei Einträge, wobei der erste ein Brief von Nietzsche an Meta von Salis ist, der zweite Eintrag

Figure 3.1: Erweiterte Suchmaske von swisscollections

Figure 3.2: Suchergebnisse für “Meta von Salis” + “Brief”

umfasst sieben Briefe von Caroline Farner, und der dritte Eintrag ist weder an noch von Meta von Salis, sondern hat sie nur zum Thema:

Ihnen würde bei einer Korpuserstellung vom Schreibtisch aus, also nur mit den angezeigten Digitalisaten, also der Großteil der Überlieferung fehlen, und Ihre Untersuchungsergebnisse wären wohl sehr verzerrt, würden Sie statistische Aussagen treffen wollen: Meta von Salis unterhielt brieflichen Kontakt zu einem Mann und einer Frau, das Geschlechterverhältnis wäre also ausgeglichen; und Frauen schreiben im Schnitt mehr Briefe an Meta von Salis als Männer. Beim Blick auf alle Suchergebnisse würden sich Ihre Aussagen aber sehr ändern, und es würde sich lohnen, diese Verzerrung, diesen Bias aus Ihrer Datengrundlage zu entfernen.

Hinzu kommt natürlich immer das grundlegende Problem bei der Suche nach Quellen: swisscollections und ähnliche Portale können nur anzeigen, was die Kooperationspartner:innen zur Verfügung stellen. Hat eine Bibliothek Briefe von Meta von Salis in ihrem Bestand, diese aber noch nicht als Datensatz erfasst, wissen Sie im Gegensatz zum obigen Beispiel nicht einmal, dass Ihnen etwas entgehen würde, dass in Ihrem Korpus überhaupt ein Bias vorhanden ist.

Ähnliche Vorsicht zur Vermeidung von Verzerrungen in der Datengrundlage gilt bei der Arbeit mit rein digitalen Daten, beispielsweise bei der Auswertung von Datensätzen aus Befragungen. Wenn Sie sich am 27.10.2022 vor die Universitätsbibliothek in Basel stellen und einen Tag lang mithilfe eines kurzen Fragebogens


Stichwortsuche Indexsuche Durchstöbern Archivtektonik Bibliographien

**swisscollections** [Erweiterte Suche bearbeiten](#) [Neue erweiterte Suche starten](#) [Neue einfache Suche starten](#)


Digitalisat verfügbar ☒ Format Brief ☒

**3 Suchergebnisse für: (Alle Felder:meta von salis)**


Seite: [←](#) 1 von 1 [→](#) Sortieren nach Relevanz



**Brief an Meta von Salis-Marschlins**  
**von Friedrich Nietzsche**  
 Nietzsche, Friedrich  
 1887.09.01-14  
 Basel, UB, UBH NL 61 : N. II 10 IV



**7 Briefe an Meta von Salis-Marschlins**  
**von Caroline Farner**  
 Farner, Karoline  
 1893.05.19-1895.04.07  
 Basel, UB, UBH NL 61 : S. II C. 2 1a-1b u.a.



**Brief an Prof. Dr. Richard Oehler**  
**von der Universitätsbibliothek Basel**  
 Universitätsbibliothek Basel  
 25. Mai 1937  
 Dokument=Item=Pièce  
 Basel, UB, UBH NL 53 : B III 1, Beil. 3

Seite: [←](#) 1 von 1 [→](#)

☒ Digitalisat verfügbar

**Ressourcentyp** ^

+ ☐ Archivmaterial (3) x

+ ☐ Handschrift (3) x

**Bibliothek** ^

☐ Basel, UB (3) x

**Person / Körperschaft** ^

+ ☐ Salis-Marschlins, Meta von (18... (2) x

+ ☐ Farner, Karoline (1842-1913) (1) x

+ ☐ Nietzsche, Friedrich (1844-1900) (1) x

+ ☐ Oehler, Richard (1878-1948) (1) x

+ ☐ Universitätsbibliothek Basel (1) x

**Sprache** ^

☐ Deutsch (3) x

**Ort** ^

☐ Basel (1) x

Figure 3.3: Suchergebnisse für “Meta von Salis” + “Brief” + “Digitalisat verfügbar”

und einer Tabellendatei erfassen, wie zufrieden die befragten Personen mit dem Essen in der Unimensa sind, werden Sie am Ende einen Datensatz erhalten, in dem sich vermutlich über 80% der Befragten für besseres und nahezu 100% für günstigeres Essen in der Mensa aussprechen – eine gute Schlagzeile für die BZ, die sich auf die neuesten Ergebnisse einer wissenschaftlichen Studie berufen kann. Führen Sie die gleiche Umfrage eine Woche später, mitten während der Herbstmesse durch, werden die Ergebnisse wohl erheblich anders aussehen. Die Wahrscheinlichkeit, dass die Mensa infolge der BZ-Schlagzeile innerhalb weniger Tage den Menüplan überarbeitet und die Preise herabgesetzt hat, ist dabei wohl geringer als diejenige, dass sich Ihr Sample, die Auswahl an Datenpunkten, also befragten Personen, durch die Messe stark verändert hat: Im Umkreis der Bibliothek treffen Sie nun nicht mehr vor allem Studierende und andere Uni-Angehörige an, sondern auch Messebesucher:innen vom Petersplatz. Auch hier sind Verzerrungen entstanden, ähnlich wie beim vorherigen Beispiel mit den Briefen: Wenn aus einer Gesamtheit nur eine spezifische Untermenge beobachtet wird, die sich durch ein gemeinsames Merkmal von der Gesamtheit unterscheidet – digitalisierte Quelle oder Besucher:in der Universitätsbibliothek –, ist die Datengrundlage und damit die Untersuchungsergebnisse biased. Um bei Daten, die Sie nachnutzen, eventuell vorhandene Verzerrungen nicht weiterzutransportieren, ist das Üben von Datenkritik eine essentielle Kompetenz.

Zur Tatsache, dass Daten eben nicht “gegeben” sind (lat. dare, datum: geben, gegeben), sondern gemacht, und daher entsprechend interpretiert werden müssen, finden Sie ein gutes Interview von Roopika Risam (2020);<sup>4</sup> zur Zementierung von Klischees durch Übersetzungsalgorithmen gibt es einen Artikel in der Republik von Marie-José Kolly und Simon Schmid (2021);<sup>5</sup> und über die Macht von Data Science und dem Änderungspotential von Data Feminism haben Catherine D’Ignazio und Lauren F. Klein 2020 ein ganzes Buch veröffentlicht.<sup>6</sup>

Zur Frage, wie sich die digitale Wende, der *digital turn*, auf die Quellenkritik auswirkt, sehen Sie sich dieses kurze Video des Projekts Ranke.2 – Quellenkritik im digitalen Zeitalter an:<sup>7</sup>

Eine Handreichung zum Umgang mit digitalisierten und digitalen Daten, das im selben Projekt erarbeitet wurde, finden Sie hier.

<sup>4</sup>Risam, Roopika: “It’s Data, Not Reality”: On Situated Data With Jill Walker Rettberg, 06.2020. Online: <<https://medium.com/nightingale/its-data-not-reality-on-situated-data-with-jill-walker-rettberg-d27c71b0b451>>, Stand: 16.08.2022.

<sup>5</sup>Kolly, Marie-José; Schmid, Simon: Sie ist hübsch. Er ist stark. Er ist Lehrer. Sie ist Kindergärtnerin, in: Republik, 04.2021. Online: <<https://www.republik.ch/2021/04/19/sie-ist-huebsch-er-ist-stark-er-ist-lehrer-sie-ist-kindergaertnerin>>, Stand: 23.08.2022.

<sup>6</sup>D’Ignazio, Catherine; Klein, Lauren F.: Data feminism, 2020. Online: <<https://direct.mit.edu/books/book/4660/Data-Feminism>>.

<sup>7</sup>comma separated value ist ein Format, in dem einzelne Werte, *values*, über spezifische Trenner, meist *commas*, eindeutig abgrenzbar sind und somit in einem Tabellenformat angezeigt werden können, wobei jeder Wert in einer separaten Zelle steht. Tabellensoftware wie Excel, Google Sheets oder Numbers kann csv-Dateien öffnen.



## Chapter 4

# Datenerhebung, -aufbereitung und -analyse

Jede Art von Forschung ist auf Daten angewiesen, seien sie mittels Personenbefragungen, medizinischer Messungen, Web Scraping oder interpretierender Analysen von Texten erhoben. Auf Grundlage von Daten können Forschungsfragen beantwortet, Thesen aufgestellt, Behauptungen widerlegt, Narrative untermauert werden. Analysen, die sich mit einem kleinen Set von Quellen bzw. Daten befassen, präsentieren Ergebnisse dabei oft in Form von Synthesen, die sich aus einer vorangehenden Interpretation der zugrundeliegenden Dokumente ergeben. Über das Quellenverzeichnis und entsprechende Anmerkungen im Text wird die Grundlage nachvollziehbar; dass ein bestimmter Abschnitt, ein Satz oder ein Wort auf eine gewisse Weise ausgelegt werden, wird aber auch durch die jeweiligen Forscher:innen selbst beeinflusst – eine Literaturwissenschaftlerin beispielsweise, die über Männerfiguren bei Joanne K. Rowling promoviert hat, wird bei der Diskussion um deren mögliche Autorschaft von *The Cuckoo's Calling* (siehe Section 2.1) diesen Text anders lesen und andere Argumente dafür oder dagegen aufwerfen als ein langjähriger Harry-Potter-Fan mit viel Leseerfahrung, aber anderer bzw. weniger formaler Ausbildung. Beide werden fundierte Aussagen treffen und Begründungen geben können, ob und wieso *The Cuckoo's Calling* von Rowling verfasst wurde oder nicht; beide werden auf ihre Erfahrung und gründliche Auseinandersetzung mit Rowlings Werk verweisen; und beide werden mit einzelnen Sätzen oder Passagen für eine Sichtweise argumentieren, die von einer dritten Person genau gegenteilig genutzt würde. Die Datengrundlage ist also dieselbe und nachvollziehbar, die Auswertung bzw. die Auswertungsstrategien hingegen sind es nicht mehr, und somit auch nicht die daraus gewonnenen Ergebnisse, die ja auch wieder Forschungsdaten darstellen.

Computergestützte Analysen haben den Anspruch, in allen Schritten nachvollziehbar zu sein und dadurch auch nachnutzbare Daten zu produzieren: Nicht

nur die Quellengrundlage, also die Erhebung von Daten und die Erstellung eines Datensatzes, sondern auch alle Schritte von der Datenanreicherung und -verfeinerung über die genutzten Methoden bzw. Programme für die Auswertung bis hin zur Sicherung und Aufbewahrung sollen transparent, gut dokumentiert und nachvollziehbar sein. Zum einen, um die Resultate und die darauf fußenden Aussagen belastbar zu machen; zum anderen, um die gewonnenen Daten zur weiteren Nutzung kostenfrei und offen verfügbar zu machen. Zu den Prinzipien, die bei der Arbeit mit Daten berücksichtigt werden sollten, geht es nochmals in Kapitel 5. An dieser Stelle stehen die konkreten Arbeitsschritte bei der Datenerhebung und -aufbereitung, der Datenanalyse und -sicherung im Zentrum, die in Digital-History-Projekten häufig vorkommen.

Es gibt verschiedene Möglichkeiten, Daten für die historische Forschung zu erheben bzw. zu erstellen, von denen einige im Folgenden kurz angesprochen werden.

Für Zeiträume, in denen Quellen vergleichsweise knapp sind und keine seriellen Daten existieren, bietet sich die **Digitalisierung von Texten** und deren anschließende Analyse an. Digitalisierung beinhaltet dabei nicht nur die Transformation von einer physischen Quelle in ein digitales Bild, sondern auch die Anreicherung des Bilds mit Layout und Text: Erst durch eine Markierung von Bereichen, in denen Text vorkommt, ist es in einem zweiten Schritt möglich, diesen als solchen zu erkennen und damit maschinenlesbar und auswertbar zu machen. Eine solche Umwandlung vom Bild zum Text ist dabei sowohl für moderne Texte, die als Typoskript vorliegen, als auch für vormoderne Handschriften und Drucke möglich, in lateinischer ebenso wie in arabischer, chinesischer oder japanischer Schrift. Es gibt kostenpflichtige Programme wie den Abbyy FineReader, aber auch Open-Source-Tools mit und ohne Graphical User Interface (GUI). Weit verbreitet ist Transkribus, das viele Funktionalitäten bündelt; die Texterkennung ist ab einer gewissen Menge Seiten allerdings kostenpflichtig, wobei studentische Projekte auf Anfrage unterstützt werden können. Programme, die über die Kommandozeile laufen, gänzlich kostenfrei sind und ebenfalls zahlreiche Funktionalitäten bieten, sind beispielsweise Kraken, OCR4all, OCRopus oder Calamari.

Zur **Extraktion von Daten** aus digitalen/digitalisierten Texten existieren verschiedene Möglichkeiten mithilfe kleiner Kommandozeilenprogramme (eher mühsam und schwierig zu lesen) oder mit Packages für Programmiersprachen, für die Geisteswissenschaften vor allem R oder Python (siehe dazu auch Section 2.1). So können beispielsweise aus digitalisierten Telefonbüchern Entitäten, also Einheiten, wie Personen, Straßennamen oder Berufe oder aus alten Theaterprogrammheften gespielte Stücke, beteiligte Schauspieler:innen und verantwortliche Regisseurinnen extrahiert und als Datensätze weitergenutzt werden.<sup>1</sup>

---

<sup>1</sup>Kommandozeile/Command Line, Bash, Shell oder Prompt finden sich oft als synonym genutzte Begriffe für Command Line Interfaces. Auf UNIX-basierten Betriebssystemen wie Mac OS und Linux ist das Terminal als Interface weit verbreitet; für Details: [https://en.wikipedia.org/wiki/Command-line\\_interface#History](https://en.wikipedia.org/wiki/Command-line_interface#History). Windows-



Der anfängliche Aufwand, der einer automatisierten Datenextraktion vorangeht und die steile Lernkurve bei der Bedienung mancher Programme können abschreckend wirken. Wenn Sie nur ein Theaterprogramm detaillierter auswerten wollen, sind Sie sicher schneller, wenn Sie die entsprechenden Daten in eine Tabellensoftware abtippen. Wenn Sie aber einen größeren Quellenbestand zur Verfügung haben, der in sich ähnlich strukturiert ist, wie das bei Telefonbüchern oder einer Serie von Theaterprogrammheften der Fall sein dürfte, macht es kaum einen Unterschied mehr, ob Sie zehn oder hundert Theaterprogramme analysieren möchten. Zudem können Sie Ihr erstelltes Skript, Ihr kleines Computerprogramm, anderen zur Verfügung stellen oder für ähnlich strukturierte Quellen in einem anderen Projekt nachnutzen.

Wenn Sie mit bereits digitalisierten Beständen aus öffentlichen Institutionen wie Galerien, Bibliotheken, Museen oder Archiven arbeiten wollen (sog. GLAMs: **G**alleries, **L**ibraries, **A**rchives, **M**useums), besteht oft die Möglichkeit, Daten über **Schnittstellen** herunterzuladen.<sup>2</sup> Solche Schnittstellen, engl. API (**A**pplication **P**rogramming **I**nterface), ermöglichen eine Kommunikation zwischen zwei Computern, ohne dass hierfür der Umweg über eine graphische Oberfläche nötig ist. Anstatt also beispielsweise über die Suchmaske der Staatlichen Museen zu Berlin nach Objekten oder Dokumenten mithilfe verschiedener Schlagwörter zu suchen und die Ergebnisse dann einzeln herunterzuladen, kann Ihr Computer mit der Schnittstelle des Museums direkt kommunizieren und mit einfachen Befehlen ganze Ergebnislisten zur Weiterarbeit herunterladen. Für solche Abfragen können ein Kommandozeilenprogramm oder Programmiersprachen genutzt werden, die Abfrage besteht dabei im Wesentlichen aus einer Zeile, wie hier in der Programmiersprache R:

```
library(jsonlite)

cats <- fromJSON("https://smb.museum-digital.de/json/objects?&s=katze")
```

Wenn Sie die Schritte nachvollziehen möchten, können Sie R hier herunterladen. Wenn Sie das Programm öffnen, müssen Sie zuerst das Paket `jsonlite` installieren: `install.packages("jsonlite")`. Mit “Enter” wird das Paket installiert. Dann können Sie die zwei Zeilen oben eintippen und ebenfalls mit “Enter” ausführen. Die Ergebnisse Ihrer Suche können Sie sich mit `cats` + “Enter” anzeigen lassen.

Das Ergebnis der Suchanfrage nach “katze” wird in der Variable `cats` gespeichert, und diese kann zur Weiterarbeit in ein Tabellenformat exportiert werden:

```
write.csv(cats, "docs/cats_smb.csv")
```

nutzer:innen kommen mit der PowerShell ganz gut zurecht, es empfiehlt sich eventuell die Installation von Cygwin oder MinGW, um mit einem UNIX-basierten Interface arbeiten zu können.

<sup>2</sup>Unter [openglam.ch](http://openglam.ch) finden sich Informationen zu Schweizer GLAM-Einrichtungen, die offene Daten anbieten.

## 22CHAPTER 4. DATENERHEBUNG, -AUFBEREITUNG UND -ANALYSE

Die Funktion `write.csv` speichert den Inhalt der Variable `cats` als csv-Datei<sup>3</sup> unter dem Dateipfad “docs/cats\_smb.csv” auf der Festplatte.

	A	B	C	D	E	F	G	H	I
1		objekt_id	objekt_name	objekt_inventarnr	objekt_erfasst_am	institution_id	institution_name	image	total
2	1	456	Statuette der Göttin Bastet in Gestalt einer sitzenden Katze	ÄM 2598	2021-11-02 21:15:59		Ägyptisches Museum und Papyrussammlung	data/lmb/resources/images/201806/200w_21081501496.jpg	134
3	2	52029	Shinô-Götterschein mit heiliger Katze	I D 17657 a,b	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06173101822.jpg	134
4	3	108895	Katze auf Rädern zum Ziehen	N (35 F) 917/1995,a	2021-11-02 21:15:59		Museum Europäischer Kulturen	data/lmb/resources/images/201808/200w_04160418311.jpg	134
5	4	256381	Einseitig bemaltes Ostrakon mit Darstellung einer Katze vor einem Opferisch mit Gans (Tefnu-Legende)	ÄM 3317	2021-11-02 21:15:59		Ägyptisches Museum und Papyrussammlung	data/lmb/resources/images/202009/200w_9f5f700958cb2.jpg	134
6	5	589	Figur der Göttin Bastet in Gestalt einer sitzenden Katze	ÄM 11385	2021-11-02 21:15:59		Ägyptisches Museum und Papyrussammlung	data/lmb/resources/images/201806/200w_21081927964.jpg	134
7	6	6962	„Häufte eines breiten Rings, darauf eine gelagerte säugende Hündin“ (tatsächlich Teil eines ägyptischen Sistrums mit säugender Katze)	Misc. 8482	2021-11-02 21:15:59		10 Antikensammlung	data/lmb/resources/images/201806/200w_27194138039.jpg	134
8	7	7012	Vierfüßiges Tier „Katze“? (es handelt sich um einen geometrischen Löwen)	Misc. 7899	2021-11-02 21:15:59		10 Antikensammlung	data/lmb/resources/images/201806/200w_27194155897.jpg	134
9	8	230034	Anonette mit Katze	1928107	2021-11-02 21:15:59		14 Kunstgewerbemuseum	data/lmb/resources/images/202009/200w_9f5ea0d3c91bf.jpg	134
10	9	63364	Mädchen mit Katze	60025-04.415	2021-11-02 21:15:59		Museum für Asiatische Kunst	data/lmb/resources/images/201807/200w_15173921440.jpg	134
11	10	106633	Geliebte Katze	N (47 B) 3/2017,35	2021-11-02 21:15:59		2 Kulturen	data/lmb/resources/images/201808/200w_04153954180.jpg	134
12	11	50739	Katze	I D 51881	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06165932553.jpg	134
13	12	51123	Katze mit Schellenbaum	I D 51948	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06171126839.jpg	134
14	13	51234	Katze 猫	I D 50252	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06171645114.jpg	134
15	14	51441	Ema: Katze	I D 52073	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06171906607.jpg	134
16	15	51467	Ema: Katze	I D 52049	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06171921233.jpg	134
17	16	51662	Ema: Katze	I D 52125	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06172105136.jpg	134
18	17	51736	Katzen	I D 52252 a,b	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06172711680.jpg	134
19	18	51794	Katzen	I D 52251 a-c	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06172700863.jpg	134
20	19	51890	Katze	I D 52290	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06172843317.jpg	134
21	20	51904	Nikko „Nemuri-no-neko“ „die schlafende Katze“ vor dem Eingang zur Cryptomerien Allee, die zur Gabelstraße des Iyeyasu führt. Nach Hidori Gingoro	VIII D 12502	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06172852239.jpg	134
22	21	50605	Kauende Katze 猫 Neko	I D 51748	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06165645371.jpg	134
23	22	50966	Katze	I D 51834	2021-11-02 21:15:59		11 Ethnologisches Museum	data/lmb/resources/images/201807/200w_06170166251.jpg	134
24	23	83570	Frñnzl mit Katze auf einer Decke legend	KdZ 26603	2021-11-02 21:15:59		15 Kupferstichkabinett	data/lmb/resources/images/201807/200w_28193302235.jpg	134
25	24	86574	St. Goar, Blick auf St. Goarshausen und Burg Katz	SM 9.13	2021-11-02 21:15:59		15 Kupferstichkabinett	data/lmb/resources/images/201807/200w_28195922385.jpg	134

Figure 4.1: Beginn der Trefferliste für “katze” über die API der Staatlichen Museen zu Berlin

Um Abfragen zu vermeiden, die die Server überlasten, haben die meisten APIs entweder eine Authentifizierung oder eine maximale Trefferanzahl pro Abfrage eingebaut. Beim obigen Beispiel erhalten Sie dadurch nicht die gesamte Trefferanzahl (134, aus der Spalte “total” ersichtlich), sondern nur die ersten 24 – diese Einstellungen haben die Entwickler:innen der Schnittstelle gemacht. Um dennoch alle Treffer mit einer Abfrage zu erhalten, müssten Sie die Dokumentation der API lesen und die Abfrage etwas modifizieren.

Wenn Sie das interessiert, finden Sie Details in der Fußnote.<sup>4</sup>

Wenn Webseiten keine Schnittstellen zur Verfügung stellen, besteht die Möglichkeit, mit **Web Scraping** an gewünschte Daten zu kommen. Je nach Webseite bzw. Inhalten ist die Rechtslage allerdings nicht ganz klar. Zum Download von Webseiten mit der Programmiersprache Python gibt es eine

<sup>3</sup>comma separated value ist ein Format, in dem einzelne Werte, *values*, über spezifische Trenner, meist *commas*, eindeutig abgrenzbar sind und somit in einem Tabellenformat angezeigt werden können, wobei jeder Wert in einer separaten Zelle steht. Tabellensoftware wie Excel, Google Sheets oder Numbers kann csv-Dateien öffnen.

<sup>4</sup>Die API aus dem Beispiel ist so konfiguriert, dass bei Abfragen mit Ergebnissen über 24 Treffern nur die ersten 24 ausgegeben werden; das ist etwas ungewöhnlich, aber wir können damit umgehen, indem wir die maximale Trefferausgabe pro Anfrage auf 10 setzen – diese Zahl ist nicht zu hoch, und wir können gut damit rechnen. Der Parameter für die maximale Trefferzahl kann mit `&breitenat=10` eingestellt werden. Den Startpunkt der Ausgabe kann man mit dem Parameter `&startwert=` ändern. Um also alle Treffer für eine Abfrage zu erhalten, können wir die Ergebnisse in 10er-Schritten abfragen und anschließend zusammenfügen. Damit das nicht zu einer copy-paste-Aktion wird, müssen wir etwas ausführlicher formulieren bzw. mehrere Variablen verwenden. Das hat aber den Vorteil, dass man auf diese Art dann nach jedem Begriff suchen kann.

Lektion im Programming Historian von William J. Turkel und Adam Crymble.  
Ein weiteres Tutorial zur Datenakquise, von Zach Coble, Liz Rodrigues, Erin Pappas, Chelcie Rowell, und Yasmeen Shorish, findet sich hier.

---

```
base_URL <- "https://smb.museum-digital.de/json/objects?&s=katze"
cats <- fromJSON(base_URL)
start <- 0
breite <- 10
iterations <- cats$total[1]%%10 + 1
endsize <- cats$total[1]-(iterations-1) * 10
cat_list <- data.frame()
for (i in 1:iterations){
  if(i < iterations){
    cat_list <- rbind(cat_list, fromJSON(paste(base_URL, "&gbreitenat=10&startwert=", start, sep="")))
  } else {
    cat_list <- rbind(cat_list, fromJSON(paste(base_URL, "&gbreitenat=",
                                              endsize, "&startwert=", start, sep="")))
  }
  start <- start + 10
  write.csv(cat_list, "Desktop/cat_list.csv")
}
```

## 4.1 Datenaufbereitung<sup>5</sup>

Bei der Arbeit mit Datensätzen, seien sie selbst erhoben oder von Dritten übernommen, ist es häufig der Fall, dass Informationen fehlen oder uneinheitlich erhoben wurden, was eine spätere Analyse erschwert.

Wenn in einer Umfrage unter Studierenden das Studienfach mit in eine Tabelle

```
}
```

Zuerst machen wir den Code übersichtlicher und speichern den Großteil der URL in `base_URL`:

```
base_URL <- "https://smb.museum-digital.de/json/objects?&s=katze"
```

Die Ergebnisse der Suchanfrage werden wieder im Objekt `cats` gespeichert:

```
cats <- fromJSON(base_URL)
```

Die Anzahl Durchgänge für eine Abfrage ergibt sich aus der **Anzahl der totalen Treffer**/10 + 1; die Anzahl der Treffer lässt sich aus der Spalte "total" im Objekt `cats` entnehmen. In R formuliert man das so:

```
cats$total[1]
```

Im Katzenbeispiel sind es 134 Gesamttreffer, also  $(134/10 \text{ ohne Rest})+1$ , also 14 Durchgänge:

```
iterations <- cats$total[1]/%10 + 1
```

Dann setzt man den Startwert auf 0:

```
start <- 0
```

Und die Maximaltreffer auf 10:

```
breite <- 10
```

Die letzte Iteration muss dabei nicht die nächsten 10 Treffer abfragen, sondern nur noch 4 (die letzten 4 nach 130):

```
endsize <- cats$total[1]-(iterations-1) * 10
```

Dann erstellen wir eine leere Tabelle, einen data frame, die wir mit unseren Anfragen nach und nach befüllen. (Bei kleinen Datenmengen kann die Funktion `rbind` zur Verbindung von Einzeltabellen genutzt werden; bei größeren Datenmengen ist das iterative Verlängern von data frames nicht empfohlen):

```
cat_list <- data.frame()
```

Wenn wir diese Variablen festgelegt haben, können wir einen Loop, eine Schleife bauen, die unter bestimmten Bedingungen verschiedene Aktionen ausführt:

```
for (i in 1:iterations){
```

Falls die letzte Iteration noch nicht erreicht ist, wird die Abfrage in 10er-Schritten durchgeführt, wobei der Startwert bei jedem Durchgang um 10 verschoben wird und die Ergebnisse hintereinander in `cat_list` geschrieben werden.

```
if(i < iterations){
  cat_list <- rbind(cat_list, fromJSON(paste(base_URL, "&gbreitenat=10&startwert=",
  start , sep="")))
} else {
```

Sobald die letzte Iteration erreicht ist, werden nicht mehr die nächsten 10, sondern so viel Treffer, wie in `endsize` gespeichert, abgefragt, in unserem Beispiel 4:

```
cat_list <- rbind(cat_list,fromJSON(paste(base_URL, "&gbreitenat=",      endsize,
"&startwert=", start, sep="")))
}
```

```
start <- start + 10
```

Zum Schluss, in diesem Fall nach 14 Iterationen, wird die Tabelle in eine Datei geschrieben:

```
write.csv(cat_list, "Desktop/cat_list.csv")
}
```

<sup>5</sup>Eine häufige Aussage ist, zur Datenvorbereitung/Preprocessing würde 80% der Arbeitszeit verwendet, zur Analyse und Interpretation blieben nur 20%. In einem Blogartikel von 2020 geht Leigh Dodds diesen Zahlen nach – ganz so dramatisch ist das Verhältnis in Wahrheit wohl nicht.

aufgenommen wurde, ohne zuvor Werte für diese Kategorie zu definieren, finden sich für “Geschichte” und “Deutsch” vielleicht auch folgende Varianten: “Gesch.”, “Geschichtswissenschaft”, “Geschichtswissenschaften”, “Geschichte”, “Germanistik”, “Dt.”, “Germ.”. Anstatt zwei Werten für zwei Studienfächer gibt es neun – ohne, dass sich das Fächerspektrum erweitert hätte. Im besten Fall werden solche Varianten schon bei der Erhebung der Daten vermieden, indem eine feste Liste an Werten erstellt wird. Erhält man jedoch einen Datensatz mit verschiedenen Varianten für ein und denselben Wert, muss man diese zusammenführen, um eine saubere Datengrundlage zu erhalten. Sie können entweder mit **Strg-R** versuchen, verschiedene Schreibweisen zu finden und zu ersetzen; in Tabellenprogrammen wie Excel, Open Office oder Google Sheets können Sie sich einzigartige Werte einzelner Spalten anzeigen lassen und zusammengehörende Varianten zu einem Grundwert zusammenführen; am hilfreichsten, recht voraussetzungslos zu bedienen und dabei auch für große Datensätze nutzbar ist die Software OpenRefine, mit der Sie Daten extrahieren,<sup>6</sup> säubern/vereinheitlichen<sup>7</sup> und anreichern<sup>8</sup> können, um eine für Ihre Forschungsfrage und dafür notwendige Analysen sinnvolle Datengrundlage zu erhalten.

Für Textdaten sind verschiedene Schritte zur Aufbereitung notwendig, je nachdem, welche Methode bzw. Software Sie nutzen möchten. Für die meisten Analysen ist es sinnvoll, mit sogenannten Stopword-Listen zu arbeiten. Stopwords sind Wörter, die vor einer Analyse aus einem Korpus entfernt werden, um aussagekräftigere Ergebnisse zu erhalten, gerade, wenn es um rein quantitative Methoden zur inhaltlichen Erschließung geht. Stopwords sind Wörter mit grammatikalischen Funktionen, die in großer Zahl in Dokumenten vorkommen, jedoch wenig Bedeutung tragen. Wenn man den unbearbeiteten Text dieses Guides nach Worthäufigkeiten auswertet, hier mit Voyant-Tools lässt sich nur schwerlich erraten, worum es geht – “digital” steht auf Platz 12, viel häufiger sind Artikel und Präpositionen. Mit Hilfe einer Stopword-Liste, die die häufigsten nicht-sinntragenden Wörter aus dem Text entfernt, wird der Inhalt klarer:

Weitere Schritte beinhalten oft eine Tokenisierung, also die Segmentierung in Einheiten der Wortebene, und eine Lemmatisierung, also die Rückführung von verschiedenen Formen eines Worts auf eine Grundform – aus “ist”, “war” und “sind” wird “sein”. Wie bei den Schreibvarianten der Studienfächer haben die verschiedenen Flexionsformen für die meisten Forschungsfragen keinen Mehrwert und können zur weiteren Analyse zusammengeführt werden. Für solche vorbereitenden Schritte gibt es existierende Software und Packages für Programmiersprachen, sodass hier das Rad nicht neu erfunden werden muss, vor allem für

<sup>6</sup>Evan Peter Williamson: Fetching and Parsing Data from the Web with OpenRefine, Programming Historian 6 (2017), <https://doi.org/10.46430/phen0065>.

<sup>7</sup>Seth van Hooland, Ruben Verborgh, Max De Wilde: Cleaning Data with OpenRefine, Programming Historian 2 (2013), <https://doi.org/10.46430/phen0023>.

<sup>8</sup>Karen Li-Lun Hwang: Enriching Reconciled Data with OpenRefine, The Bytegeist Blog 2018, <https://medium.com/the-bytegeist-blog/enriching-reconciled-data-with-openrefine-89b885dcadbb>

		Term	Count
<input type="checkbox"/>	1	und	191
<input type="checkbox"/>	2	die	170
<input type="checkbox"/>	3	https	150
<input type="checkbox"/>	4	in	119
<input type="checkbox"/>	5	der	101
<input type="checkbox"/>	6	sie	96
<input type="checkbox"/>	7	für	89
<input type="checkbox"/>	8	von	87
<input type="checkbox"/>	9	zu	83
<input type="checkbox"/>	10	mit	74
<input type="checkbox"/>	11	ist	72
<input type="checkbox"/>	12	digital	66
<input type="checkbox"/>	13	sich	64
<input type="checkbox"/>	14	data	61
<input type="checkbox"/>	15	oder	50
<input type="checkbox"/>	16	zur	49
<input type="checkbox"/>	17	eine	49
<input type="checkbox"/>	18	daten	49
<input type="checkbox"/>	19	ein	47
<input type="checkbox"/>	20	das	46
<input type="checkbox"/>	21	es	44
<input type="checkbox"/>	22	werden	42
<input type="checkbox"/>	23	den	37
<input type="checkbox"/>	24	auf	37
<input type="checkbox"/>	25	um	36

Figure 4.2: Worthäufigkeiten roher Text

		Term	Count
<input type="checkbox"/>	1	https	150
<input type="checkbox"/>	2	digital	66
<input type="checkbox"/>	3	data	61
<input type="checkbox"/>	4	daten	49
<input type="checkbox"/>	5	history	35
<input type="checkbox"/>	6	wiki	32
<input type="checkbox"/>	7	doi.org	29
<input type="checkbox"/>	8	tools	24
<input type="checkbox"/>	9	online	23
<input type="checkbox"/>	10	en.wikipedia.org	23
<input type="checkbox"/>	11	quellen	21
<input type="checkbox"/>	12	command	21
<input type="checkbox"/>	13	chapter	21
<input type="checkbox"/>	14	shell	18
<input type="checkbox"/>	15	digitale	18
<input type="checkbox"/>	16	text	17
<input type="checkbox"/>	17	arbeit	17
<input type="checkbox"/>	18	analyse	16
<input type="checkbox"/>	19	terminal	15
<input type="checkbox"/>	20	line	15
<input type="checkbox"/>	21	interface	14
<input type="checkbox"/>	22	geschichte	14
<input type="checkbox"/>	23	forschung	14
<input type="checkbox"/>	24	ressourcen	13
<input type="checkbox"/>	25	literacy	13

Figure 4.3: Worthäufigkeiten ohne Stopwords

moderne, weit verbreitete Sprachen, siehe auch Section B.2. Schwieriger wird es für nicht-standardisierte Sprachen bzw. Sprachformen, also dialektal geprägte oder vormoderne Texte. Zwar gibt es auch hierfür Programme, die tatsächlich erreichte Präzision muss dabei jedoch je nach Quelle beurteilt werden.

## 4.2 Datenanalyse

Wenn Sie einen Datensatz zur Analyse zur Verfügung haben, aus selbst erhobenen Daten oder durch Nachnutzung eines vorhandenen, und für Ihre Zwecke aufbereitet haben, folgt (endlich) auch die Analyse. Welche Software oder Methoden Sie verwenden, hängt dabei nicht nur von der Art und Menge der Daten, sondern auch dem Datenformat und vor allem auch Ihrer Forschungsfrage ab. Wenn Sie eine Personendatenbank haben, in der Briefschreiber:innen und Empfänger:innen aufgenommen sind und der Wohnort der Personen bekannt ist, Sie es jedoch versäumt haben, die Datierungen der Einzelbriefe zu verzeichnen, können Sie nur eine räumliche Verteilung, keine raum-zeitliche Entwicklung eines Briefschreiber:innennetzwerks darstellen.<sup>9</sup> Wenn Sie aber nur an der örtlichen Verteilung weiblicher und männlicher Verfasser:innen interessiert sind und die zeitliche Komponente für Sie keine Rolle spielt, erübrigt sich auch ein raum-zeitliche Analyse. Bevor Sie sich also für eine Methode entscheiden, sollten Sie sich fragen, zu welchem Zweck Sie Ihren Datensatz nutzen wollen und welche Frage(n) er beantworten soll.

In einem nächsten Schritt sollte über die konkrete Art der Analyse nachgedacht werden, die mit den vorhandenen Daten möglich ist. Unter den zahlreichen Möglichkeiten für die Arbeit mit **strukturellen Daten** sind für die Geschichtswissenschaften u.a. die Netzwerkanalyse oder die Regressionsanalyse häufig genutzte Methoden. Für **textuelle Daten** bieten sich ebenfalls verschiedene Arten der Analyse an, darunter beispielsweise Auszählungen von Worthäufigkeiten als Teil der Stylometrie/Zuschreibung von Autor:innenschaft (siehe Section 2.1), Topic Modelling als statistische Methode zur Identifizierung wiederkehrender Themen in größeren Textbeständen, oder Sentimentanalyse, um Stimmungen, Gefühle, Bewertungen aus Textpassagen zu extrahieren. Wenn Sie über **georeferenzierte Daten** verfügen, können Sie verschiedene Analysen mithilfe von GIS (Geographic Information System) durchführen und visualisieren.

Ob Sie für Topic Modelling ein eigenes Skript schreiben oder vorhandene Software nutzen, ob Sie Regressionsanalysen selbst durchführen oder auf Webseiten durchführen lassen, ist dabei Ihre Entscheidung; oftmals ist das Nutzen vorhandener Webangebote für erste kurze Analysen sinnvoll, um zu überlegen, ob die

<sup>9</sup>Ein Großprojekt an der Universität Stanford, “Mapping the Republic of Letters”, hat für das 18. Jahrhundert das Briefnetzwerk europäischer Gelehrter modelliert. Ein Fallbeispiel ist das Netzwerk Voltaire’s, in verschiedenen Visualisierungen: <http://republicofletters.stanford.edu/publications/voltaire/letters/>. Dan Edelstein. Interactive Visualization for Voltaire’s Correspondence Network. Letters in Voltaire’s Network [Created using Palladio], <http://hdlab.stanford.edu/palladio/>.

vorgesehene Methode überhaupt sinnvolle Ergebnisse liefern kann. Für größere Projekte, in denen komplexere Analysen über einen längeren Zeitraum durchgeführt werden sollen, bietet sich die Arbeit mit Programmiersprachen schon allein deswegen an, weil so ein sehr hohes Maß an Anpassungen von vorhandenen Funktionen für die eigenen Zwecke und die völlige Kontrolle über die eigenen Daten ermöglicht wird. Eine Auflistung häufig genutzter Tools für die historische Arbeit findet sich in Section B.2.

### 4.3 Datensicherung

In Kapitel 5 wird es um Fragen zur nachhaltigen Speicherung von Forschungsdaten gehen; an dieser Stelle sei darauf hingewiesen, dass die Sicherung von Daten am besten auch mit einer Versionierung und mit einer Dokumentation einhergeht. **Datenversionierung** hat den Vorteil, dass Schritte wieder rückgängig gemacht werden können, Datensätze in unterschiedlichen Stadien gespeichert und für eine spätere Weiterarbeit genutzt werden können und einzelne Schritte einzelnen Projektmitarbeiter:innen zugeschrieben werden können. Zusätzliche Versionierung geht dabei über die Funktionalitäten von Backup-Programmen oder Cloudspeichern wie Dropbox oder Switchdrive hinaus, und für Einzelprojekte wie auch für kollaboratives Arbeiten hat sich in der Wissenschaft wie in der Wirtschaft git etabliert, häufig in Kombination mit Daten-/Coderepositorien auf GitHub. Die meisten von Ihnen werden vermutlich keine eigenen GitHub-Repositorien anlegen, aber das System dennoch irgendwann nutzen, am ehesten durch den Download von dort zur Verfügung gestellten Daten – die Textdaten für diesen Guide liegen auch in einem GitHub-Repositorium. Die **Dokumentation** von gespeicherten Daten schließlich beinhaltet Informationen zur Entstehung des Datensatzes: Wie und von wem wurden die Daten erhoben? Wie wurden sie annotiert? In welchem Format sind die Daten vorhanden? Welche Software wurde an welcher Stelle benutzt? Was stellen die Daten dar? Die Sicherung von Daten an mehreren Orten, bspw. auf der lokalen Festplatte, in einem Cloudspeicher und auf einem USB-Stick, schützt sicher vor Datenverlust. Eine Dokumentation und die Sicherung in einem Repository, einem Langzeitspeicher für Daten, sorgt zusätzlich für Sichtbarkeit und die Möglichkeit zur Nachnutzung von Ergebnissen. Als Fachrepositorien für die Geisteswissenschaften existieren beispielsweise DARIAH-DE oder das DaSCH, es gibt spezialisiertere Repositorien wie AMAD (Mittelalter), oder für alle Disziplinen offene wie Zenodo (fächerübergreifend, betrieben durch das CERN). Sie können Ihre Forschungsdaten dort kostenfrei ablegen, Ihre Urheberschaft nachweisen und die Daten/Publication mit einem Digital Object Identifier (DOI), also einem eindeutigen und dauerhaften digitalen Identifikator, nachhaltig zitierbar machen.



## Chapter 5

# FAIR und CARE

Bereits beim Beginn eines Projekts, sei es eine Proseminararbeit oder ein kollaboratives Großprojekt, sollten Fragen nach Sicherung, Austauschbarkeit und Nachnutzbarkeit von Forschungsdaten gestellt werden. Denn oftmals enden Projekte, ohne dass erstellte Daten für anschließende Forschungen verfügbar gemacht werden, sei es, weil nicht rechtzeitig nach Lösungen zur langfristigen Speicherung gesucht wurde, sei es, weil Daten in einer Form erhoben und gespeichert wurden, die eine Nachnutzung erschwert oder auch unmöglich macht.

Zu Beginn des Studiums stehen solche Fragen weniger im Fokus; dennoch sollen diese hier kurz thematisiert werden, um dafür zu sensibilisieren; auch, weil sie den Prozess der Datenerhebung beeinflussen.

Die Prinzipien FAIRer Daten wurden 2016 von einem Konsortium aus Wissenschaftler:innen und Organisationen wie folgt definiert:<sup>1</sup> **F**indability, **A**ccessibility, **I**nteroperability, **R**euse of digital assets.

Daten sollen also auffindbar und zugänglich sein, zudem interoperabel, also mit verschiedenen Systemen nutzbar, und wiederverwendbar. Wenn Sie für eine Proseminararbeit zehn Testamente aus dem 18. Jahrhundert im Staatsarchiv Basel fotografieren, anschließend transkribieren, die vererbten Gegenstände identifizieren, zwischen den Erblasser:innen vergleichen und Ihre Ergebnisse ausgedruckt bei dem:r Dozierenden einreichen, sind Ihre Daten das genaue Gegenteil: Niemand weiß, dass Sie die Daten erhoben haben; und wenn Ihr:e Dozent:in Ihre Ergebnisse anderen Studierenden zur Verfügung stellen will, um weitere Forschung anzuregen, geht dies nur in Form von Kopien Ihrer gedruckten Arbeit. Wenn Sie Ihre transkribierten Texte und die identifizierten Objekte in Standardformaten und mit offener Lizenz auf einem Repositorium veröffentlichen,

---

<sup>1</sup>Wilkinson, Mark D.; Dumontier, Michel; Aalbersberg, IJsbrand Jan u. a.: The FAIR Guiding Principles for scientific data management and stewardship, in: Scientific Data 3 (1), 03.2016, S. 160018. Online: <<https://doi.org/10.1038/sdata.2016.18>>, Stand: 09.11.2022.

machen Sie nicht nur wichtige Teile Ihrer eigenen Arbeit sichtbar, sondern erleichtern so auch anschließende Forschungen. Zudem kann so vermieden werden, dass geleistete Arbeit wie beispielsweise Transkriptionen nicht doppelt gemacht wird.

**Part I**

**Praxis**



Im Praxisteil sollen verschiedene Schritte rund um die **Datenerhebung** – wie komme ich von Quellen zu (strukturierten) Daten, und was ist das überhaupt? –, die **Datenaufbereitung** – wie kann bzw. muss ich die Daten für meine Zwecke bearbeiten – und die **Datenanalyse** – wozu sind die strukturierten Daten da, und was mache ich damit? – an einem kleinen Beispiel durchgeführt werden. Dabei werden viele Praktiken und Konzepte nur angeschnitten, die bei Interesse mithilfe weiterführender Literatur und Tutorials vertieft werden können. Manch einer wird Inhalte vermissen – Lücken sind unvermeidbar, aber Anregungen sind herzlich willkommen.

Als Quellenbeispiel dient eine Briefsammlung, die im Rahmen der digitalen Edition “Der Sturm” an der Akademie der Wissenschaften in Mainz erstellt wurde.<sup>2</sup> Das Projekt, das Briefe von Personen der internationalen Avantgarde rund um die Zeitschrift “Der Sturm” aufbereitet, hatte dabei unterschiedliche Nutzer:innengruppen im Blick und stellt die Quellen bzw. die erstellten Daten auf verschiedene Weise zur Verfügung: Auf der Webseite des Projekts können die bisher edierten Briefe am Bildschirm gelesen werden, und es gibt ein Register der in den Texten genannten Entitäten, also Einheiten, hier der Personen, Orte und Werke; darüberhinaus gibt es die Möglichkeit, über eine Schnittstelle die Quellen oder die Registerdaten herunterzuladen. Wir können uns den Briefen also über den Vordereingang, die Webseite, oder durch die Hintertür, mittels Kommandozeile, nähern, und beide Herangehensweisen vergleichen. Grundlegende Konzepte für die Arbeit mit Daten, bzw. für die Schritte von der Quelle zum Datensatz, werden angesprochen.

Briefe sind eine gängige Quellengattung in den verschiedenen Epochen/Areas, und sie können sowohl für Textanalysen als auch zur Gewinnung von Strukturdaten genutzt werden. Das Beispiel zielt also auch auf eine größtmögliche Nähe zur tatsächlichen historischen Arbeit.

---

<sup>2</sup>Kommandozeile/Command Line, Bash, Shell oder Prompt finden sich oft als synonym genutzte Begriffe für Command Line Interfaces. Auf UNIX-basierten Betriebssystemen wie Mac OS und Linux ist das Terminal als Interface weit verbreitet; für Details: [https://en.wikipedia.org/wiki/Command-line\\_interface#History](https://en.wikipedia.org/wiki/Command-line_interface#History). Windows-nutzer:innen kommen mit der PowerShell ganz gut zurecht, es empfiehlt sich eventuell die Installation von Cygwin oder MinGW, um mit einem UNIX-basierten Interface arbeiten zu können.



## Chapter 6

# Briefedition ‘Der Sturm’

Die Webseite <https://sturm-edition.de/> dient als Portal für die Arbeit mit Quellen zum avantgardistischen STURM-Unternehmen, das mit der Gründung der gleichnamigen Zeitschrift in Berlin im Jahr 1910 durch Herwarth Walden begann und mit den nachfolgenden Gründungen einer Galerie, einer Bühne und eines Verlags internationale Bedeutung erlangte. Neben Walden waren weitere Akteur:innen an diesem Unternehmen beteiligt, und bisher (Stand November 2022) sind 179 Briefe von drei Künstler:innen an Walden im Portal verfügbar. Zudem wurden die Zeitschrift, Ausstellungskataloge, Jahrbücher, Verlagsschriften und weitere Materialien wie Plakate, Fotografien oder Einladungskarten digitalisiert.

Gehen Sie auf die Startseite und lesen Sie die Kurzbeschreibung zu den Briefen, Personen, Orten und Werken, um eine erste Idee vom Material zu bekommen. Klicken Sie dann die Briefabteilung an und schauen sich Brief Nummer 8 von Franz Marc an Herwarth Walden an.

Wie Sie sehen können, wurde der Brief nicht nur digitalisiert, also in ein digitales Bild umgewandelt, das Sie über einen externen Viewer betrachten können, sondern auch historisch-kritisch ediert und nach den Richtlinien der TEI P5 in XML codiert.

Was bedeutet das genau, und wieso ist es wichtig für unsere Arbeit als Historiker:innen?

Bei einer historisch-kritischen Edition wird keine reine Leseversion eines Textes erstellt, wie man es beispielsweise von der Textdarstellung in einem Roman kennt, sondern zusätzliches Material zum besseren Verständnis des Texts herangezogen und als Information in einem sog. kritischen Apparat zur Verfügung gestellt. Beispielsweise werden Quellen, die ein:e Verfasser:in für bestimmte Textpassagen als Vorlage genutzt hat, genannt, oder auf Ereignisse während der Entstehungszeit, die Einfluss auf den Text hatten, verwiesen.

Entitäten wie Personen, Orte, Werke usw. werden erklärt oder kommentiert. Der Text selbst wird so quellengetreu wie möglich dargestellt, orthographische oder grammatikalische Fehler werden nicht korrigiert, und meist werden auch extratextuelle Elemente wie Durchstreichungen oder Hervorhebungen im Lesetext dargestellt.

**[1914-04-14 / Sindelsdorf]**

↗ [185v]   
 Angabe der Folierung, d.h. der Seite im Quellenbestand in der Staatsbibliothek zu Berlin

↗ [185r] **L. W.,**

bestätige mit vielem Dank 60 Mk für 2 Holzschnitte. Auch diese kleinen Verkäufe nutzen mir momentan sehr. Ich habe wohl neun Holzschnitte in Arbeit, aber alle für die Illustration an der Genesis gedacht, die ich keinesfalls vorher veröffentlichen darf. Aber ich werde noch suchen, nebenbei für den Sturm ↗ (i) ein paar neue zu schneiden. Hat Arnold eigentlich die Bilder aus Breslau (3 kleine Kompositionen u. Wasserfall) Ihnen zugesandt? Ich habe von dort nichts gehört. Vollmacht hab ich geschickt. Ist Klugen ganz vom Erdboden verschwunden. Ich hegte noch immer stille Hoffnungen auf ihn. Auf die komische Sache bin ich neugierig. Daß Aug. Macke in der | Sezession ausgestellt haben soll, kann ich schwer glauben, wenigstens nicht von sich aus. C. wird die Sachen von anderer Seite haben. Oder ist es vielleicht Hellmuth Macke? August ist momentan mit Klee u. Moillet in Tunis! ↗

↗ [185v] Hrzl.  
Ihr FMarc.

[Empfänger]  
Herr  
Herw. Walden  
Verlag „Sturm“  
Berlin W. 9.  
Potsdamerstr. 134/a

**FAKSIMILES DIESER QUELLE**

185v  
185r

↗ DFG VIEWER

Bestandshaltende Institution: Staatsbibliothek zu Berlin - Preussischer Kulturbesitz. Lizenz: Public Domain.

Digitalisierter Brief  
Klick auf “DFG-Viewer” führt zu Darstellung in externem Viewer

Repräsentation von extratextuellen Elementen wie Unterstreichungen

Kritischer Apparat mit Erläuterung zur Tunisreise, die im Text genannt wird

Figure 6.1: Brief Nr. 8 von Franz Marc an Herwarth Walden.<sup>1</sup> Rote Einfärbung kennzeichnet eine Verlinkung, meist auf Entitäten wie Personen oder Orte.

Wie genau die Editor:innen jeweils vorgegangen sind – dies unterscheidet sich von Edition zu Edition –, wird in den jeweiligen Editionsrichtlinien vermerkt. Für die Quellen des STURM-Projekts gibt es verschiedene Richtlinien, je nach Quellenart. Die der Briefedition finden Sie hier.

<sup>1</sup>Kommandozeile/Command Line, Bash, Shell oder Prompt finden sich oft als synonym genutzte Begriffe für Command Line Interfaces. Auf UNIX-basierten Betriebssystemen wie Mac OS und Linux ist das Terminal als Interface weit verbreitet; für Details: [https://en.wikipedia.org/wiki/Command-line\\_interface#History](https://en.wikipedia.org/wiki/Command-line_interface#History). Windows-nutzer:innen kommen mit der PowerShell ganz gut zurecht, es empfiehlt sich eventuell die Installation von Cygwin oder MinGW, um mit einem UNIX-basierten Interface arbeiten zu können.



Eine Codierung in XML bedeutet, dass eine Textdatei mit der Extensible Markup Language ausgezeichnet wurde, d.h. Strukturen im Text werden durch festgelegte Zeichen so markiert, dass sie sowohl von Menschen als auch von Computern interpretiert werden können. Etwas geläufiger ist Ihnen vermutlich HTML, **H**ypertext **M**arkup **L**anguage, eine Sprache zur Auszeichnung elektronischer Dokumente. Die Prinzipien sind dabei ähnlich: Festgelegte Elemente werden durch einen öffnenden und einen schließenden Tag ausgezeichnet, z.B.

`<salute>Hrzl. Gruß</salute>`,

um die Grußformel in einem Brief als solche zu markieren. Das hat den Vorteil, dass bei der Suche nach Grußformeln nicht konkrete Begriffe formuliert werden müssen, sondern nach dem entsprechenden Element gesucht werden kann, und zwar sowohl vom Menschen als auch von der Maschine. Darauf kommen wir nochmals zurück. Zunächst betrachten wir die Briefe auf der Webseite.



## Chapter 7

# Durch den Vordereingang

Stellen Sie sich vor, Sie würden für ein Forschungsprojekt unter anderem die Briefe von Franz Marc an Herwarth Walden untersuchen wollen. Sie wären bei Ihrer Untersuchung daran interessiert, welche Grußformeln Marc in seinen Briefen an Walden benutzte, um aus eventuellen Änderungen Rückschlüsse auf das Verhältnis der beiden ziehen zu können. Wie würden Sie vorgehen, um sich eine Übersicht zu verschaffen? Würden Sie eine Downloadmöglichkeit der Webseite nutzen oder die Briefe online lesen? Welche Schritte würden Sie durchführen, um die Texte aller Briefe von Marc and Walden auf Ihrem Computer zu speichern? Notieren Sie Ihr Vorgehen zu den Fragen in einem Dokument.

Es gibt zahlreiche Wege, die zum Ziel führen, und keiner ist dabei besser oder schlechter; aber manche sind möglicherweise effizienter als andere – das heißt, Sie sparen Zeit, die Sie für andere Dinge verwenden können, sei es fürs Studium oder in der Badi.

Egal, wie Sie vorgegangen sind, erledigen Sie folgende Aufgaben:

1. Erstellen Sie eine Übersicht über alle Grußformeln in den 54 Briefen von Franz Marc an Herwarth Walden. Welches Format – analog oder digital – bzw. welchen Dateityp Sie wählen, bleibt Ihnen überlassen.
- 2.a) Wenn Ihr Nachname mit einem Buchstaben zwischen A und L beginnt: Nehmen Sie Brief Nr. 8 von Marc an Walden und unterteilen Sie ihn in strukturelle Elemente.
- 2.b) Wenn Ihr Nachname mit einem Buchstaben zwischen M und Z beginnt: Nehmen Sie Brief Nr. 9 von Marc an Walden und unterteilen Sie ihn in strukturelle Elemente. Ob Sie den Brief hierfür ausdrucken und Strukturelemente mit einem Stift markieren oder ob Sie am Computer arbeiten, können Sie wählen.
3. Nehmen Sie denselben Brief wie aus Aufgabe 2a bzw. 2b, je nach Nachname, und markieren Sie Entitäten, also Einheiten wie Personen, Orte, etc. Erstellen Sie hierfür ein Tabellendokument (mit Excel, Google Sheets, Open Office o.Ä.),

in das Sie die Entitäten aufnehmen.

Nachdem Sie die Aufgaben erledigt haben, lesen Sie “The Ten Commandments of Inputting Data” in Kapitel 3 im Buch “Quantitative Data in the Humanities”.<sup>1</sup> Das Kapitel finden Sie im ADAM-Workspace Ihres Einführungskurses – es lohnt sich als Ganzes, aber die “Ten Commandments” reichen auch. Eine eher praxisorientierte Onlinefassung finden Sie hier im das Buch begleitenden Blog.

Würden Sie nach der Lektüre bei Aufgabe 3 anders vorgehen? Notieren Sie sich etwaige Änderungen bzw. Erkenntnisse und bringen Sie Ihre Resultate zur begleitenden Sitzung mit, in analoger oder digitaler Form.

---

<sup>1</sup>Lemercier, Claire; Zalc, Claire: Quantitative Methods in the Humanities. An Introduction, Charlottesville 2019, S. 57–60.

## Chapter 8

# Durch die Hintertür

Bevor wir uns näher mit dem Beispielkorpus der Briefedition befassen, werfen wir einen kurzen Blick auf die Interaktionsmöglichkeiten mit dem Computer und wie wir diese für unsere Arbeit als Historiker:innen nutzen können, sei es für die Erhebung, die Aufbereitung oder die Analyse von Daten.

Es gibt zwei Arten, um mit einem Computer zu interagieren bzw. ihn zu nutzen: über ein **G**raphical **U**ser **I**nterface (GUI), also vor allem mit der Maus und durch das Anklicken von Objekten, oder, etwas direkter, über die Kommandozeile<sup>1</sup>. Um via GUI eine Datei “Brief1.txt” im Ordner “Briefe” zu löschen, öffnet man den Finder (Mac), den Explorer (Windows) oder den Filebrowser der Wahl (Linux), klickt sich zum Ordner “Briefe”, macht einen Rechtsklick auf die zu löschende Datei “Brief1.txt”, klickt “In den Papierkorb legen” oder zieht die Datei mit der Maus direkt dorthin. Dieselbe Aktion kann man als Kommando schreiben: Man öffnet das Terminal (Linux oder Mac; den Finder öffnen und im Suchfenster “Terminal” eingeben und Programm öffnen) oder eine PowerShell (Windows; mit der rechten Maustaste auf das Startsymbol klicken, dann “Windows PowerShell” auswählen), navigiert im sich öffnenden Fenster mit Texteingabe zum entsprechenden Ordner, bspw. `cd Documents/Briefe` + ‘Enter’ (Mac und Linux) bzw. `cd ./Documents/Briefe` (Windows) und gibt dort das Kommando `rm Brief1.txt` ein, das mit der Entertaste ausgeführt wird.

```
(base) serina00@dg-19-mac-02 ~ % cd Documents/Briefe
```

---

<sup>1</sup>Kommandozeile/Command Line, Bash, Shell oder Prompt finden sich oft als synonym genutzte Begriffe für Command Line Interfaces. Auf UNIX-basierten Betriebssystemen wie Mac OS und Linux ist das Terminal als Interface weit verbreitet; für Details: [https://en.wikipedia.org/wiki/Command-line\\_interface#History](https://en.wikipedia.org/wiki/Command-line_interface#History). Windows-nutzer:innen kommen mit der PowerShell ganz gut zurecht, es empfiehlt sich eventuell die Installation von Cygwin oder MinGW, um mit einem UNIX-basierten Interface arbeiten zu können.

```
(base) serina00@dg-19-mac-02 Bilder % rm Brief1.txt
```

Vorgehen in der Kommandozeile bzw. im Terminal auf MacOS

Die beiden Vorgehensweisen unterscheiden sich dabei in drei Punkten:

1. Das Kommando **rm** ist endgültig, die Datei ist ohne Übergangszeit im Papierkorb gelöscht.
2. Das Kommando lässt sich relativ simpel auf eine Vielzahl von Dokumenten anwenden, wobei ganz unterschiedliche Bedingungen beachtet werden können, und es lässt sich mit anderen Kommandos verbinden.
3. Terminal sieht k3wl aus.

Bevor wir den zweiten – und für unsere Arbeit hilfreichsten – Unterschied genauer anschauen, kurz zur Kommandozeile.

In einem Terminal/einer Shell – zur Unterscheidung siehe Fußnote 1 – können Kommandos bzw. Programme ausgeführt werden, die auf der Datenstrukturebene stattfinden – wie beispielsweise das Löschen einer Datei, **rm** *Dateiname.xyz* (**rm** für *remove*), oder das Erstellen eines Ordners, **mkdir** *NeuerOrdner* (**mkdir** für *make directory*). Ebenso möglich sind Operationen auf Dateninhaltsebene – wie beispielsweise das Suchen eines bestimmten Begriffs in einer Textdatei, **grep** 'Begriff' *Textdatei.txt* (Mac/Linux) bzw. **Select-String -Path Textdatei.txt -Pattern 'Begriff'** (Windows), oder das Auszählen mehrerer Begriffe und das Speichern des Ergebnisses in einer neuen Datei, **grep -Ec '(Begriff1|Begriff2)' Textdatei.txt | wc -l > Ergebnisse.txt** (Mac/Linux) bzw. **(Select-String -Path Textdatei.txt -Pattern '(Begriff1|Begriff2)'.Matches.Count > Ergebnisse.txt** (Windows) – die Kommandos werden weiter unten nochmals einzeln erklärt.

Woher weiss Ihre Shell aber, was sie ausführen soll, wenn Sie **rm** oder **grep/String-Select** eintippen? Es gibt zahlreiche Shell-Programme, die bereits auf Ihrem System vorinstalliert sind, und mit denen Sie vieles tun können – öffnen Sie Ihre Shell und tippen Sie **date** ein: Das aktuelle Datum mit Uhrzeit erscheint. (Ihre Shell sucht nach dem ersten Argument, dem Befehl **date**, im Filesystem des Computers, und wenn sie fündig wird, führt sie eine Aktion mit den entsprechenden Parametern aus.)

#### **i** Note

tmi: Wenn Sie **echo \$PATH** im Terminal (Mac/Linux) bzw. **\$env:PATH** (Windows) eingeben, sehen Sie eine Auflistung der Orte, an denen nach Befehlen gesucht wird. Tippen Sie **which date** ein und drücken Sie 'Enter', um zu sehen, wo das Programm "date" in Ihrem Computer liegt.

Falls Sie einen Befehl eintippen, den es nicht gibt bzw. für den kein installiertes

Programm auf Ihrem Computer existiert, bekommen Sie eine simple Fehlermeldung – kaputtgehen kann dabei nichts:

```
(base) serina00@dg-19-mac-02 ~ % nonsense

command not found: nonsense
```

Der Output auf Windows ist etwas ausführlicher:

```
nonsense: The term 'nonsense' is not recognized as a name of a cmdlet, function, script file,
Check the spelling of the name, or if a path was included, verify that the path is correct and
```

Das aktuelle Datum wird Ihnen wahrscheinlich auch in Ihrer Toolbar angezeigt, und einen neuen Ordner können Sie per Rechtsklick erstellen, dazu brauchen Sie das Terminal nicht unbedingt. Um einen Begriff in einem Textdokument zu finden und alle Vorkommen zu zählen, können Sie das Dokument öffnen, **Strg-F** drücken, den Begriff eingeben und das Ergebnis sehen. Wenn Sie nach mehreren Begriffen suchen wollen, müssen Sie dieselbe Aktion zweimal ausführen: **Strg-F**, Begriff 2. Und wenn Sie mehrere Dateien durchsuchen möchten, beispielsweise um herauszufinden, wie oft die Grußformel “Mit herzlichem Gruß” in einer Briefsammlung vorkommt, müssen Sie die Suche in jeder Datei einzeln ausführen. Wenn Sie dann noch nach der Variante “Mit herzlichen Grüßen” oder gar “Herzl. Gruß” suchen wollen, vervielfacht sich Ihre Arbeit.

Sie können dasselbe auch mit dem Terminal machen und einige der Built-in-Programme nutzen, um sich Zeit und Arbeit zu sparen.

## 8.1 Strg-F 2.0

Wir arbeiten wie im vorangegangenen Kapitel mit einem Teilkorpus der Quelenedition “Der Sturm”, nämlich mit allen Briefen, die von Franz Marc verfasst wurden. Um die folgenden Schritte nachzuvollziehen, laden Sie sich den Ordner “letters\_Der\_Sturm” herunter. Sie können dazu entweder das vollständige GitHub-Repository zu diesem Guide als Zip-Datei herunterladen und entpacken, und im Ordner “docs” befindet sich der Ordner “letters\_Der\_Sturm”.

Sie können das Repository auch über die Kommandozeile klonen

```
(base) serina00@dg-19-mac-02 ~ % git clone https://github.com/wissen-ist-acht/digitalhistory.i
```

oder als bequeme Variante diesen Direktlink nutzen.

Wenn Sie die Schnittstelle der Webseite ausprobieren wollen, kommen Sie mit wenigen Kommandos an die Dateien.

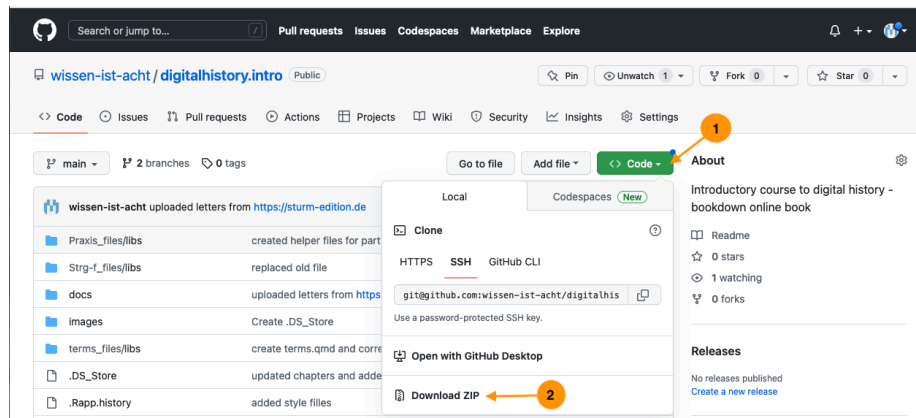


Figure 8.1: GitHub-Repositorium mit Quellcode, Download via “Code” und “Download ZIP”.

### 🔥 Download per Schnittstelle (für Mac/Linux)

Mit dem ersten Kommando erstellen wir eine Datei “briefe\_marc.xml” mit den Dateinamen aller Briefe, die von Franz Marc geschrieben wurden – über das Register auf der Webseite wissen wir, dass er die Personen-ID P.0000003 hat; die URL zur Abfrage der Schnittstelle können wir der Dokumentation entnehmen:

```
curl https://sturm-edition.de/api/persons/P.0000003 --output briefe_marc.xml
```

Wenn Sie die Datei mit einem Editor öffnen, der XML-Dateien lesen kann, sehen Sie, dass neben den Dateinamen, die nach “target=” stehen, noch viel Beifang ist, den wir loswerden möchten:



```
<person xmlns="http://www.tei-c.org/ns/1.0" source="http://d-nb.info/gnd/11857745X" xml:id="
  <persName type="pref">Marc, Franz</persName>
  <persName type="fn">Franz Marc</persName>
  <linkGrp type="files">
    <ptr n="Bl.375" target="Q.01.19191212.JVH.01.xml"/>
    <ptr n="Bl.377" target="Q.01.19200114.JVH.01.xml"/>
    <ptr n="Bl.219" target="Q.01.19160128.FMA.01.xml"/>
    <ptr n="Bl.222" target="Q.01.19160205.FMA.01.xml"/>
    <ptr n="Bl.223" target="Q.01.19160302.FMA.01.xml"/>
    <ptr n="Bl.218" target="Q.01.19160101.FMA.01.xml"/>
    <ptr n="Bl.221" target="Q.01.19160122.FMA.01.xml"/>
    <ptr n="Bl.220" target="Q.01.19160115.FMA.01.xml"/>
    <ptr n="Bl.207" target="Q.01.19150703.FMA.01.xml"/>
    ...
```

Denn eigentlich brauchen wir nur die Dateinamen, um die Dateien mit einem entsprechenden Befehl herunterladen zu können. Mit dem zweiten Kommando erstellen wir eine deswegen eine neue Datei, in der die einzelnen extrahierten Dateinamen mit dem Download-Kommando `curl` kombiniert und um die entsprechende URL zum Download ergänzt werden:

```
cat briefe_marc.xml | grep -o 'Q.*xml\b' | perl -nle 'print "curl -o $_ https://sturm-edition.de/api/files/"
```

Die Datei “dateinamen\_briefe\_marc.txt” sieht so aus:

```
curl -o Q.01.19191212.JVH.01.xml https://sturm-edition.de/api/files/Q.01.19191212.JVH.01.xml
curl -o Q.01.19200114.JVH.01.xml https://sturm-edition.de/api/files/Q.01.19200114.JVH.01.xml
curl -o Q.01.19160128.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19160128.FMA.01.xml
curl -o Q.01.19160205.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19160205.FMA.01.xml
curl -o Q.01.19160302.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19160302.FMA.01.xml
curl -o Q.01.19160101.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19160101.FMA.01.xml
curl -o Q.01.19160122.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19160122.FMA.01.xml
curl -o Q.01.19160115.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19160115.FMA.01.xml
curl -o Q.01.19150703.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19150703.FMA.01.xml
curl -o Q.01.19150417.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19150417.FMA.01.xml
curl -o Q.01.19151106.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19151106.FMA.01.xml
curl -o Q.01.19150918.FMA.01.xml https://sturm-edition.de/api/files/Q.01.19150918.FMA.01.xml
...
```

`cat briefe_marc.xml` gibt den Inhalt der Datei ins Terminal; `grep -o 'Q.*xml\b'` findet in diesem Inhalt alle Zeichenketten zwischen “Q” und “xml”, wobei nach “xml” durch das Hinzufügen von “\_” das Zeichenende angezeigt ist; die 54 gefundenen Zeichenketten werden in je eine neue Zeile geschrieben, wobei mit `curl -o $_` der Befehl “curl -o” und mit `$_` als Platzhalter die Zeichenkette (also der Dateiname) geschrieben wird,

gefolgt von “https://sturm-edition.de/api/files/*undmit`\_wieder* die Zeichenkette (also wieder der Dateiname). Mit einem dritten Kommando, `bash`, führen wir die erstellte Datei aus, d.h. die in ihr stehenden Kommandos werden ausgeführt -- also `viacurl` (Client **U**RL) die Briefe heruntergeladen.

```
bash dateinamen_briefe_marc.txt
```

Egal, wie Sie die Dateien heruntergeladen haben, sollten Sie 54 Briefe im xml-Format vorfinden. Öffnen Sie dann das Terminal (Mac/Linux) bzw. die PowerShell (Windows) und bewegen sich mit `cd`, also **change directory**, in den Ordner (directory), in dem Ihre Textdateien liegen.

In meinem Fall ist das unter `Documents/GitHub/digital_history_intro/docs/letters_Der_Sturm`.

```
(base) serina00@dg-19-mac-02 ~ % cd Documents/GitHub/digital_history_intro/docs/lett
```

Bei den meisten von Ihnen ist das vermutlich unter “Downloads” – probieren Sie es aus.

(Um zu prüfen, was in einem Ordner liegt, können Sie im Terminal `ls` (für **list**) eingeben, bzw. in der PowerShell `dir` (für **directory**):

```
% ls
```

```
Q.01.19140115.FMA.01.xml  Q.01.19150315.FMA.02.xml
Q.01.19140119.FMA.01.xml  Q.01.19150327.FMA.01.xml
Q.01.19140121.FMA.01.xml  Q.01.19150417.FMA.01.xml
Q.01.19140124.FMA.01.xml  Q.01.19150501.FMA.01.xml
Q.01.19140125.FMA.01.xml  Q.01.19150615.FMA.01.xml
Q.01.19140125.FMA.02.xml  Q.01.19150703.FMA.01.xml
Q.01.19140409.FMA.01.xml  Q.01.19150710.FMA.01.xml
Q.01.19140414.FMA.01.xml  Q.01.19150818.JVH.01.xml
Q.01.19140421.FMA.01.xml  Q.01.19150827.FMA.01.xml
Q.01.19140507.FMA.01.xml  Q.01.19150906.FMA.01.xml
Q.01.19140512.FMA.01.xml  Q.01.19150911.FMA.01.xml
...
```

### 8.1.1 Erste Analysen

Wenn Sie in den Ordner navigiert sind, in dem die Briefdateien liegen, können Sie mit einem einzeiligen Kommando die Suchvorgänge, die Sie nacheinander mit **Strg-F** mit jeder einzelnen Datei in einem Texteditor ausführen würden, mit dem Programm **grep** (Global Regular Expression Print, Mac/Linux) bzw.

`Select.String` (Windows) für alle Briefe in diesem Ordner vornehmen, indem Sie alle Dateien, die auf “.xml” enden, in die Suche aufnehmen. Die Ergebnisse – bei dieser Suche nach “Mit herzlichem Gruß” oder “Mit herzlichen Grüßen” ein Treffer in einem Brief – können Sie sich im Terminal anschauen:

Mac/Linux:

```
% grep -E -i '(Mit herzlichem Gruß|Mit herzlichen Grüßen)' *.xml
```

Windows:

```
% Select-String -Path *.xml -Pattern "(Mit herzlichem Gruß|Mit herzlichen Grüßen)"
```

Output:

```
Q.01.19160115.FMA.01.xml:          <salute>Mit herzlichen Grüßen für Sie beide</salute> <
```

Die Formulierung “Mit herzlichen Grüßen” kommt also einmal im Korpus vor, und zwar im Dokument Q.01.19160115.FMA.01.xml.

Sie können auch mit `wc -l` (Mac/Linux) den **W**ordcount, die Anzahl der gefundenen Treffer auf Zeilenebene, `-l` zählen, bzw. mit `Matches.Count` (Windows), und mit `>` in eine neue Datei schreiben (die während der Ausführung des Kommandos erstellt wird):

Mac/Linux:

```
% grep -E -i '(Mit herzlichem Gruß|Mit herzlichen Grüßen)' *.xml | wc -l > count_greetings.txt
```

Windows:

```
% (Select-String -Path *.xml -Pattern "(Mit herzlichem Gruß|Mit herzlichen Grüßen)").Matches.C
```

Wenn Sie die neu erstellte Datei `count_greetings.txt` öffnen, die sich im selben Ordner wie die Briefe befindet, sollte dort “1” stehen, weil unsere Suche einen Treffer ergeben hat.

Das Kommando `grep` (Mac/Linux) hat im obigen Befehl den Zusatzparameter `E` bekommen, das Kommando `Select.String` (Windows) den Parameter `-Pattern`, d.h. wir suchen nicht eine exakte Zeichenkette, sondern nutzen Möglichkeiten zur Mustersuche, zur Suche nach Patterns. Diese werden formuliert als sog. **E**xtended Regular Expressions (von hier kommt das `E`), als reguläre Ausdrücke. Wir haben in unserer Suchabfrage nämlich nicht nur nach “Mit herzlichem Gruß” gesucht, sondern auch nach “Mit herzlichen Grüßen”,

formuliert mit dem Zeichen “|”, hier als “oder” zu lesen. Mithilfe Regular Expressions können wir unsere Suche weiter ausbauen und nach verschiedenen Varianten/Schreibweisen auf einmal suchen.

### **i** Note

Regular Expressions haben verschiedene *flavours* – je nach Programmiersprache werden Dinge etwas anders formuliert, und manche Defaulteinstellungen unterscheiden sich. In unserem Fall benötigt **grep** noch den Parameter **-i**, um Groß- und Kleinschreibung zu ignorieren. **Select.String** ignoriert dies by default und braucht keinen zusätzlichen Parameter. Solche Feinheiten sind bei der Arbeit mit Regular Expressions wichtig zu wissen, aber das lernt man on the go.

Mac/Linux:

```
% grep -E -i '(Mit herzlichem Gru(ß|ss)|Mit herzlichen Grü(ß|ss)en|H(e|.?)rzt. Gru(f
```

Windows:

```
% (Select-String -Path *.xml -Pattern "(Mit herzlichem Gru(ß|ss)|Mit herzlichen Grü
```

So formuliert finden wir 17 Treffer für eine Grußformel, mit den möglichen Schreibweisen “Mit herzlichem Gruß”, “Mit herzlichem Gruss”, “Mit herzlichen Grüßen”, “Mit herzlichen Grüßen”, “Herzl. Gruß”, “Herzl. Gruss”, “Hrzt. Gruß”, “Hrzt. Gruss”.

Wenn wir herausfinden möchten, ob Grüße mal *herzlich*, mal *hrzt.* oder *freundlich* verschickt wurden, können wir die Suche und die Art der Ausgabe modifizieren:

Mac/Linux:

```
% grep -E -i 'Gr(u|ü)(ß|ss)' *.xml
```

Windows:

```
% Select-String -Path *.xml -Pattern "Gr(u|ü)(ß|ss)"
```

Output:

```
Q.01.19140115.FMA.01.xml:      stets sofort antworte; es muß verloren
Q.01.19140119.FMA.01.xml:      <salute>Hrzt. Gruß</salute> <signed>Ihr
Q.01.19140125.FMA.02.xml:      <salute>Hrzt. Gruß</salute>
```

```

Q.01.19140421.FMA.01.xml:      <closer>Gute Besserung <persName key="P.000000
Q.01.19140507.FMA.01.xml:      <salute>besten Gruß</salute>
Q.01.19140730.JVH.01.xml:      herzlichsten Grüßen für Sie beiden</salute> <signed>Ihre
Q.01.19140831.FMA.01.xml:      <salute>Hrzt. Gruß von Eurem Freund in Waffen</sa
Q.01.19140908.JVH.01.xml:      <salute>Viele herzlichsten Grüssen für Sie beiden
Q.01.19141113.FMA.01.xml:      <salute>Hrzt. Gruß 1 x 2</salute> <signed>Ihr <pe
Q.01.19141129.JVH.01.xml:      <salute>Viele herzliche Grüßen für Sie beiden</s
Q.01.19150112.FMA.01.xml:      <salute>Hrzt. Gruß Ihnen beiden</salute>
Q.01.19150116.FMA.01.xml:      <salute>Mit hrzt. Gruß Ihnen beiden</salute> <si
Q.01.19150121.FMA.01.xml:      <salute>Hrzt. Gruß</salute> <signed>Ihr <persNam
Q.01.19150131.JVH.01.xml:      <salute>Viele herzliche Grüßen für Sie beiden</s
...

```

Mit diesem Kommando durchsuchen wir also den Text nach dem Muster **Gr(u|ü)(ß|ss)**, also Beginn mit **Gr** oder **gr**, dann folgt entweder ein **u** oder ein **ü**, dann entweder ein **ß** oder **ss**. Weil wir kein Wortende markiert haben (das ginge mit `\b`), werden auch “Grüße” oder “Grüssen” gefunden.

Wenn Sie sich während der Lektüre des vorangegangenen Kapitels auf der Webseite durch die Briefe geklickt haben, werden Sie festgestellt haben, dass ein Brief nicht immer mit “Gruß” oder “Grüßen” endet. Beim Output der Suchanfragen im Terminal sehen Sie, dass alle Grußformeln von einem Tag-Paar umgeben sind: `<salute>` kennzeichnet den Beginn des Grußes, `</salute>` das Ende. Öffnen Sie eine der Briefdateien und suchen Sie nach “salute”. (Wenn Sie keinen XML-fähigen Editor auf dem Computer haben, öffnen Sie die Datei einfach mit einem Browser.)

Wie Sie sehen, gibt es das Tag-Paar `<salute>-</salute>` zweimal, einmal umrahmt vom Tag-Paar `<opener>-</opener>`, einmal von `<closer>-</closer>`. Die Anrede ist mit dem ersten, die Grußformel mit dem zweiten Tag-Paar markiert. Wir können also, wenn wir mit Dokumenten arbeiten, die nach festgelegten Richtlinien ausgezeichnet wurden, nach dem Element Grußformel suchen, ohne erst einen Blick in die Texte werfen zu müssen, um verschiedene Suchabfragen zu formulieren. Wir formulieren unsere Suchabfrage um:

Mac/Linux:

```
% grep -E -zo '<closer>[[:cntrl:]].*<salute>.*[[:cntrl:]].*<' *.xml
```

Output:

```

Q.01.19140115.FMA.01.xml:<closer>
                        <salute>Hrzt.<
Q.01.19140119.FMA.01.xml:<closer>
                        <salute>Hrzt. Gruß</salute> <signed>Ihr <persName key="P.0000003" ref="htt

```

```

112 ▼      <opener>
113          <salute>Lieber <persName key="P.0000001" ref="http://d-
            nb.info/gnd/118770950">Walden</persName>, </salute>
114      </opener>
115 ▼      <p>ich vermute, daß <persName key="P.0000058" ref="http://d-
            nb.info/gnd/118870645">Cassirer</persName> es auf den Proceß ankommen läßt; ich
            rate nicht dazu;
116          Sie wi<hi rend="underline">sse</hi>n ja meine Bedingung, daß ich auf gar
            keinen
117          Fall irgend welche Kosten von der Sache haben möchte. Das ist mir weder
118          <persName key="P.0000058" ref="http://d-
            nb.info/gnd/118870645">Cassirer</persName> noch <term type="journal"
            key="W.0000070" ref="http://d-nb.info/gnd/4127687-5">die
119          Aktion</term> wert.</p>
120      <p>Von <persName key="P.0000057" ref="http://d-
            nb.info/gnd/143669230">Reiche</persName> resp. <persName key="P.0000059"
            ref="http://d-nb.info/gnd/2097840-6">Arnold</persName> hab ich illustr. Katalog
            erhalten; Sie werden ihn wohl
121          auch inzwischen gesehen haben; sonst kann ich Ihnen den meinen schicken.</p>
122 ▼      <p>Von
123          <persName key="P.0000033" ref="http://d-
            nb.info/gnd/118969161">Filla</persName> hab ich die <term type="artwork"
            key="W.0000011">
124              <hi rend="underline">Häringe</hi>
125          </term> gewählt, von
126          <persName key="P.0000060" ref="P.0000060">Beneé</persName> ein größeres
            Aquarell, ich glaube mit blau (hell,
127          geometrisch) kein, hochformat.<pb xml:id="S.178v.02" n="178v"
            facs="http://resolver.staatsbibliothek-berlin.de/SBB0000DAA400000001"/> Auf
            beide Bilder haben sie
128          auf der Rückseite mit Blei „Marc“ geschrieben. Sie werden es schon finden.
            Sobald ich die Sachen habe, sende ich 2 Aquarelle; geben Sie mir bitte die
            Adressen.</p>
131 ▼      <closer>
132          <salute>Hrzt. Gruß</salute> <signed>Ihr <persName key="P.0000003"
            ref="http://d-nb.info/gnd/11857745X">F. Marc</persName>
133          </signed>
134      </closer>

```

Figure 8.2: Ausschnitt aus Brief Nr. 1 von Franz Marc an Herwarth Walden

```

Q.01.19140121.FMA.01.xml:<closer>
    <salute>Hrzl.</salute> <signed>Ihr <persName key="P.0000003" ref="http://d
Q.01.19140125.FMA.02.xml:<closer>
    <salute>Hrzl. Gruß<
Q.01.19140409.FMA.01.xml:<closer>
    <salute>Herzl.<
Q.01.19140414.FMA.01.xml:<closer>
    <salute>Hrzl.</salute> <signed>Ihr <persName key="P.0000003" ref="http://d
Q.01.19140507.FMA.01.xml:<closer>
    <salute>besten Gruß<
Q.01.19140512.FMA.01.xml:<closer>
    <salute>Hrzl.</salute> <signed>Ihr <persName key="P.0000003" ref="http://d
Q.01.19140606.FMA.01.xml:<closer>
    <salute>Hrzl.</salute> <signed>Ihr <persName key="P.0000003" ref="http://d
Q.01.19140608.FMA.01.xml:<closer>
    <salute>hrzl.</salute> <signed>Ihr <persName key="P.0000003" ref="http://d
...

```

Wenn wir die Ergebnisse direkt in eine Datei schreiben wollen, können wir das natürlich auch tun:

Mac/Linux:

```
% grep -E -zo '<closer>[[[:cntrl:]]].*<salute>.*[[[:cntrl:]]].*<' *.xml > Grussformeln.txt
```

Spätestens jetzt wäre es aber an der Zeit, das Instrumentarium zu wechseln: Mit dem Terminal bzw. der Shell kann man verschiedenste Operationen durchführen, und es gibt zahlreiche kleine Programme, die man zusätzlich installieren kann – zum Parsen, also Zerlegen von XML-Dateien, zur Bearbeitung von Bilddateien oder zum Download von YouTube-Videos. Die Übersichtlichkeit ist allerdings recht begrenzt, und gerade für die Analyse von Struktur- und Textdaten gibt es weitaus geeignetere Programmiersprachen wie R oder Python, wie in Section 2.1 bereits erwähnt.





## Chapter 9

# Ausblick

Wenn Sie die von Ihnen erstellten Strukturierungen von Brief Nr. 8 bzw. Nr. 9 von Franz Marc an Herwarth Walden und die identifizierten Entitäten mit denjenigen in den XML-Dateien der STURM-Editor:innen vergleichen, ergeben sich sicher einige Unterschiede. Ein zentraler ist sicher, dass sich die Herausgeber:innen bei ihrer Strukturierung an ein Schema gehalten haben, das im Bereich der Texteditionen Standard ist, TEI XML, und das kann zahlreiche Vorteile haben. So können Sie beispielsweise die im vorherigen Kapitel durchgeführten Abfragen nach der Grußformel durchführen, ohne sich Gedanken darüber machen zu müssen, ob sich die Benennung des Tags auf halber Strecke ändert. Und wenn Sie sich während der Forschungsarbeit dafür entscheiden würden, anstatt Grußformeln besser die Anrede zu untersuchen, oder aber das Korpus auf die Briefe von Jacoba van Heemskerck auszuweiten, könnten wir dies mit wenigen Änderungen in unseren Abfragen machen, weil auch die Anrede mit einem einheitlichen Tag codiert ist; würden wir die Quellen nur über die Webseite lesen und unsere Auszählungen von Hand machen, würde unsere Arbeit mit dem Hinzufügen neuer Dokumente von vorne beginnen.

Das Erstellen von standardisierten Daten mithilfe eines Schemas bzw. bestimmter Richtlinien ermöglicht es auch, verschiedenen Datensätze miteinander zu kombinieren oder mit weiteren Daten anzureichern. Bei einem Blick in die Briefe, auf der Webseite oder in der XML-Datei, wird ersichtlich, dass Entitäten wie Personen oder Orte nicht nur als solche markiert und projektintern verlinkt, sondern auch mit weiteren Normdaten verbunden wurden, beispielsweise mit dem dazugehörigen Eintrag in der GND, der Gemeinsamen Normdatei der Deutschen Nationalbibliothek, oder in Geonames, einer Datenbank für geographische Daten.

Wenn Sie den Link zu Kandinski oder zu Berlin anklicken, erhalten Sie auf den Seiten der GND bzw. Geonames zahlreiche zusätzliche Informationen zur Person bzw. zum Ort, unter anderem Lebensdaten bzw. Geokoordinaten.

```

112 <opener>
113 <salute>Lieber <persName key="P.0000001" ref="http://d-
nb.info/gnd/118770950">Walden</persName>, </salute>
114 </opener>
115 <p>ich habe nicht das geringste vom Anwalt erhalten; Sie kennen mich ja, daß ich
116 stets sofort antworte; es muß verloren gegangen sein. Grüßen Sie bitte D<hi rend="super">
117 <hi rend="underline">r</hi>
118 </hi>
119 <persName key="P.0000056" ref="P.0000056">Feige</persName> und sagen Sie ihm
120 das.</p>
121 <p>In einer niederrheinischen Zeitung soll von einer bevorstehenden großen
122 Ausstellung in <placeName key="0.0000045"
ref="http://sws.geonames.org/2952539">Barmen</placeName> von <persName key="P.0000009"
ref="http://d-nb.info/gnd/118559737">Kandinsky</persName> mir u.s.w.
berichtet worden sein; ich kann mir nur denken, daß die <persName key="P.0000057"
123 ref="http://d-nb.info/gnd/143669230">Reiche</persName>-<placeName key="0.0000032"
ref="http://sws.geonames.org/2935022">Dresden</placeName> Collection dahin
124 kommt;<pb xml:id="S.177v.02" n="177v" facs="http://resolver.staatsbibliothek-
berlin.de/SBB0000DAA300000001"/> ich
125 bin natürlich einverstanden, nur soll die fre<hi rend="underline">e
126 Rück</hi>fracht <placeName key="0.0000045"
ref="http://sws.geonames.org/2952539">Barmen</placeName> - <placeName key="0.0000002"
ref="http://sws.geonames.org/2950159">Berlin</placeName> gesichert
127 sein.<note>Im Jahr 1913 fand in der Stadt Barmen eine STURM-Ausstellung mit
128 Werken der Künstlergruppe „Der Blaue Reiter“ statt. In der Literatur gilt
129 diese Ausstellung als nicht ausreichend nachgewiesen; dieser Brief gibt
130 endgültigen Aufschluss über das tatsächliche Stattfinden der Ausstellung.
131 Vgl. Enders, Rainer: Ausstellungen außerhalb der Berliner Galerie. URL: <ref
target="https://www.arthistoricum.net/themen/portale/sturm/ausstellungen/">https://www.ar
thistoricum.net/themen/portale/sturm/ausstellungen/</ref>
132 (Aufruf 06.04.2017).</note> Aber es ist natürlich nur Vermutung von mir. Von
133 <persName key="P.0000057" ref="http://d-nb.info/gnd/143669230">Reiche</persName> höre ich
gar nichts.</p>
134 <closer>
135 <salute>HrZl.</salute>
136 <signed>Ihr <persName key="P.0000003" ref="http://d-nb.info/gnd/11857745X">F.
137 Marc</persName>
138 </signed>
139 </closer>

```

Figure 9.1: Ausschnitt aus Brief Nr. 1 von Franz Marc an Herwarth Walden, Normdaten gelb hervorgehoben.

Wenn Sie nun beispielsweise wissen wollten, welche Orte in den Briefen Franz Marcs genannt werden, könnten Sie diese nicht nur mithilfe des Tag-Paars `<placeName>-</placeName>` extrahieren, sondern mit den dazugehörigen Geokoordinaten anreichern und sich auf einer Karte anzeigen lassen. Für solche Vorgänge reicht ein kurzes (aber nicht unbedingt schnell erstelltes ...) Skript, das sich auf weitere Dokumente ausweiten lässt – ob Sie nur die Orte in den Briefen Franz Marcs oder auch in denen Jacoba van Heemskercks extrahieren und visualisieren wollen, spielt mit Blick auf die Rechenzeit des Skripts keine Rolle.

Das Erstellen von Datensätzen nach bestimmten Richtlinien, einerseits formal, andererseits auch mit Blick auf die FAIR-Prinzipien, bietet also viele Vorteile für die eigene Arbeit – so müssen beispielsweise Schemata zur Klassifizierung nicht von Neuem erfunden werden – ebenso wie für die Arbeit anderer – grundlegende Informationen können übernommen werden und es bleibt mehr Zeit für die inhaltliche Forschung.

Ein anschauliches Beispiel für die Weiternutzung von Daten ist das von Studierenden erstellte Projekt *quoteSalute*, eine Webseite, auf der Sie historische Grußformeln generieren lassen können, falls Ihnen beim Briefeschreiben der Standardgruß zu langweilig geworden ist. Das Projekt hat hierfür mehrere XML-codierte Briefkorpora, die alle als offen nutzbare Daten online verfügbar

sind, genutzt, die Grußformeln extrahiert und angereichert. Die genaue Projektbeschreibung finden Sie hier.

Neuer Gruß quote me!

»Also, auf baldiges enormes Wiederseh'n. Herzlichst der Deinige.«

Hermann Bahr – Arthur Schnitzler: Briefwechsel, Aufzeichnungen, Dokumente 1891–1931  
Adele Sandrock an Hermann Bahr, 17. 8. 1894  
CC BY 4.0

**Grußformeln filtern**

<b>Von</b>	<b>An</b>	<b>Form</b>	<b>Sprache</b>
<input type="checkbox"/> ♀ Weiblich <input type="checkbox"/> ♂ Männlich <input type="checkbox"/> ○ Neutral	<input type="checkbox"/> ♀ Weiblich <input type="checkbox"/> ♂ Männlich <input type="checkbox"/> ○ Neutral	<input type="checkbox"/> 📧 Formal <input type="checkbox"/> ❤ Informal	<input type="checkbox"/> Deutsch <input type="checkbox"/> Englisch <input type="checkbox"/> Spanisch <input type="checkbox"/> Italienisch <input type="checkbox"/> Französisch <input type="checkbox"/> Griechisch <input type="checkbox"/> Latein

✓ Filter anwenden    🗑 Filter aufheben

**Schöner Grüßen**

Auf dieser Webseite können Sie nach historischen Grußformeln stöbern. Haben Sie die passende Grußformel gefunden, können Sie sie mit dem "quote me!"-Button ganz einfach kopieren. Öffnen Sie das E-Mail-Programm Ihrer Wahl und dann einfach in die nächste Mail einfügen und versenden! Wenn Ihnen die angezeigte Grußformel nicht gefällt, generieren Sie sich einfach mit Klick auf "Neuer Gruß" eine andere. Mit Filtern können Sie die Suchanfrage außerdem nach Ihren Wünschen einschränken.

**Mitmachen**

Die Grußformeln Ihrer digitalen Briefedition sollen bei quoteSalute dabei sein? Gerne! Mehr Informationen finden Sie in der [Dokumentation](#).

Figure 9.2: Startseite von quoteSalute

Wie Sie sehen, können Sie sich Grüße nicht nur generieren lassen, sondern auch filtern, beispielsweise nach Geschlecht der Absender:innen bzw. der Adressat:innen oder nach Sprache. Sie können dies deswegen, weil die vorhandenen Codierungen der Textdateien im Projekt ausgeweitet und zum Beispiel um das Geschlecht der beteiligten Personen, sofern ersichtlich, ergänzt wurden. Der Code für das Projekt ist dabei offen auf einem GitHub-Repository zugänglich, und neue Korpora werden gerne aufgenommen.

Computergestütztes und computerbasiertes Arbeiten, das sollte dieser Guide vermitteln, vereinfacht, beschleunigt oder ermöglicht gar Prozesse, die wir für unsere historische Forschung nutzen können. Dabei gibt es mittlerweile zahlreiche Programme, die über eine graphische Oberfläche bedient werden können und die meisten Anforderungen aus geisteswissenschaftlicher Sicht erfüllen. Manche Analysen hingegen erfordern sehr spezifische Schritte oder viel Rechenkapazitäten, für die sich das Erlernen einer Programmiersprache lohnen kann. Gemeinsam ist beidem, dass es sowohl für das Formulieren einer interessanten Fragestellung, für die Auswahl der Datenbasis als auch für die Interpretation der Ergebnisse, die die Maschine liefert, informierte Expertise braucht – ein Programm zur Bestimmung der Autor:innenschaft eines unbekannten Textes rechnet unendlich lange, wenn es alle existierenden Texte mit einem unbekannten vergleichen muss; hier braucht es eine Eingrenzung auf eine Auswahl durch Literaturwissenschaftler:innen. Historische Forschung ist immer auch Analyse des Einzelfalls, das Partikularen, eines *close readings*; die Möglichkeit, den Blick mithilfe des Computers auszuweiten, kann in den

meisten Fällen gewinnbringend in Betracht gezogen werden.

# Appendix A

## Glossar

---

API	<b>A</b> pplication <b>P</b> rogramming <b>I</b> nterface: a facility offered by a web resource which allows search queries independent of a <b>GUI</b> , often performed using scripts
bash	default program that runs in the <b>command line</b>
bias	systematic error that results from an unbalanced sample
big data	huge amount of data, identifiable through repeated freezing of your standard program when opening a file
born digital data	data which originated in a digital form
CLI	<b>C</b> ommand <b>L</b> ine <b>I</b> nterface, text interface that allows interaction with the computer; see also <b>bash</b>
close reading	careful and attentive interpretation of a text
CMS	<b>C</b> ontent <b>M</b> anagement <b>S</b> ystem
Console	See <b>CLI</b>
Crowdsourcing	projects that include the active participation of the public to generate content, transcribe sources etc.
csv	<b>c</b> omma <b>s</b> eparated <b>v</b> alues, a structured text format, using commas as separators between columns
distant reading	quantitative approach to huge amounts of texts, using computational methods to search for interpretable patterns
GUI	<b>G</b> raphical <b>U</b> ser <b>I</b> nterface
HTML	<b>H</b> ypertext <b>M</b> arkup <b>L</b> anguage, a structured text format, like the format this guide is written in, to render documents in a browser
Jupyter notebook	web application/interactive coding environment that runs in a browser; let's you create and share code ( <a href="https://jupyter.org">https://jupyter.org</a> )

---

machine learning	umbrella term for different methods that use data to do a task in a specific way, using data to learn and to improve the results
machine readable	transformation of, for example, text into a data format that is processable by a computer
OCR	<b>O</b> ptical <b>C</b> haracter <b>R</b> ecognition, process of transforming text on an image into a data format
OS	<b>O</b> perating <b>S</b> ystem
open source	freely available source code that can be used, modified and redistributed without limitations
OSS	<b>O</b> pen <b>S</b> ource <b>S</b> oftware
Regular Expression	syntax for search and replace text using patterns (instead of exact matches)
terminal	See <b>CLI</b>
web scraping	extracting data from websites

---

## Appendix B

# Literatur, Tools, Tutorials

- Brennan, Sheila A.: Digital History, in: The Inclusive Historian's Handbook, <https://inclusivehistorian.com/digital-history/>, 04.06.2019.
- Hohls, Rüdiger: Digital Humanities und digitale Geschichtswissenschaften, in: Busse, Laura u. a. (Hg.): Clio-Guide. Ein Handbuch zu digitalen Ressourcen für die Geschichtswissenschaften, Berlin 2018, S. A.1-1–B.1-34. Online: <https://doi.org/10.18452/19244>.
- Romein, C. Annemieke u. a.: State of the Field: Digital History, in: History 105 (365), 04.2020, S. 291—312. Online: <https://doi.org/10.1111/1468-229X.12969>.
- Winters, Jane: Digital History, in: Tamm, Marek; Burke, Peter (Hg.): Debating New Approaches to History, London 2019, S. 277–300.
- Art. “Digital history”, in: Wikipedia, 07.09.2022. Online: [https://en.wikipedia.org/w/index.php?title=Digital\\_history&oldid=1109027465](https://en.wikipedia.org/w/index.php?title=Digital_history&oldid=1109027465), Stand: 02.11.2022.

### B.1 Einführungen und Guides

- Battershill, Claire; Ross, Shawna: Using Digital Humanities in the Classroom. A Practical Introduction for Teachers, Lecturers, and Students, London u.a. 2022.
- Blaney, Jonathan u. a.: Doing Digital History. A Beginner's Guide to Working with Text as Data, Manchester 2021.
- Cohen, Daniel J.; Rosenzweig, Roy: Digital History. A Guide to Gathering, Preserving, and Presenting the Past on the Web, Philadelphia 2006. Online: <https://chnm.gmu.edu/digitalhistory/>.

- Döring, Karoline u. a. (Hg.): Digital History. Konzepte, Methoden und Kritiken Digitaler Geschichtswissenschaft, Boston 2022, Online: <https://doi.org/10.1515/9783110757101>.
- Dougherty, Jack; Nawrotzki, Kristen (Hg.): Writing History in the Digital Age, 2013. Online: <https://doi.org/10.3998/dh.12230987.0001.001>.
- Graham, Shawn u. a.: Exploring Big Historical Data. The Historian's Macroscope, 2022. Online: <https://doi.org/10.1142/12435>.
- Lemercier, Claire; Zalc, Claire: Quantitative Methods in the Humanities. An Introduction, Charlottesville 2019.

## B.2 Tools für digital history (free/open source)

### B.2.1 Allgemein

- Programming Historian: Tutorials zu verschiedenen Tools und Methoden für historische Forschung und Lehre

### B.2.2 Text-/Korpusanalyse

- AntConc: Korpusanalyse-Toolkit
- Lemmatisierung: Sammlung der FID Romanistik
- Natural Language Toolkit, Package für Python zur Tokenisierung, Lemmatisierung usw.: NLTK
- Tokenisierung: Tutorial von fortext zu NLTK
- Voyant-Tools: Sammlung von Tools zur Textanalyse, browserbasiert oder standalone

### B.2.3 Visualisierung

- Bostock, Michael; Heer, Jeffrey; Ogievetsky, Vadim: A Tour through the Visualization Zoo. A Survey of Powerful Visualization Techniques, from the Obvious to the Obscure, in: Queue 8, Nr. 5 (2010). Online: <https://queue.acm.org/detail.cfm?id=1805128>
- Data Visualisation Catalogue: Guide zur Auswahl von Visualisierungsformen
- FID Romanistik: Sammlung von Tools zur Datenvisualisierung
- RAWGraphs: Tool zur Datenvisualisierung von tabularen Daten (.tsv-, .csv-, .dsv- oder .json-Dateien)



## B.3 Digital Literacy, Digital Criticism

- Ekström, Andreas: The Moral Bias behind your Search Results, TED talk 7.12.2015 (9:18), Online: [https://www.youtube.com/watch?v=\\_vBggxCNNno](https://www.youtube.com/watch?v=_vBggxCNNno).
- Gibbs, Frederick W.: New Forms of History: Critiquing Data and Its Representations, in: The American Historian, February 2016. Online: <http://tah.oah.org/february-2016/new-forms-of-history-critiquing-data-and-its-representations/>.
- Tavani, Herman; Zimmer, Michael Zimmer: Search Engines and Ethics, in: Edward N. Zalta (Hg.): The Stanford Encyclopedia of Philosophy (Fall 2020 Edition), Online: <https://plato.stanford.edu/archives/fall2020/entries/ethics-search/>, Kap. 3.1.

## B.4 Terminal/Command Line/Shell

- Dawson, Ted: Introduction to the Windows Command Line with PowerShell, Programming Historian 5 (2016), <https://doi.org/10.46430/phen0054>. (self-learning lesson)
- MIT Computer Science Department: 1-hour-lecture on the Shell (video)
- Milligan, Ian; Baker, James: Introduction to the Bash Command Line, Programming Historian 3 (2014), <https://doi.org/10.46430/phen0037>. (self-learning lesson)
- datacamp course: Introduction to Shell (interactive self-learning lesson)
- Jeroen Janssens: Data Science at the command line (book)

## B.5 Regular Expressions

- Knox, Doug: Understanding Regular Expressions, Programming Historian 2 (2013), <https://doi.org/10.46430/phen0033>. (self-learning lesson)
- RegexOne: Learn Regular Expressions with simple, interactive exercises. (interactive self-learning tutorial)

## B.6 XML

- Latex Ninja Blog: A shamelessly short intro to XML for DH beginners (includes TEI) (blog post)

Blaney, Jonathan; Winters, Jane; Milligan, Sarah u. a.: Doing digital history: a beginner's guide to working with text as data, Manchester 2021 (IHR research guides).

- D'Ignazio, Catherine; Klein, Lauren F.: Data feminism, 2020. Online: <<https://direct.mit.edu/books/book/4660/Data-Feminism>>.
- Kolly, Marie-José; Schmid, Simon: Sie ist hübsch. Er ist stark. Er ist Lehrer. Sie ist Kindergärtnerin, in: Republik, 04.2021. Online: <<https://www.republik.ch/2021/04/19/sie-ist-huebsch-er-ist-stark-er-ist-lehrer-sie-ist-kindergaertnerin>>, Stand: 23.08.2022.
- Le Roy Ladurie, Emmanuel: La fin des érudits, in: Le Nouvel Observateur, 08.1968.
- Lemercier, Claire; Zalc, Claire: Quantitative Methods in the Humanities. An Introduction, Charlottesville 2019.
- Ridsdale, Chantel; Rothwell, James; Smit, Mike u. a.: Strategies and Best Practices for Data Literacy Education Knowledge Synthesis Report, 2015. Online: <<https://doi.org/10.13140/RG.2.1.1922.5044>>.
- Risam, Roopika: "It's Data, Not Reality": On Situated Data With Jill Walker Rettberg, 06.2020. Online: <<https://medium.com/nightingale/its-data-not-reality-on-situated-data-with-jill-walker-rettberg-d27c71b0b451>>, Stand: 16.08.2022.
- Romein, C. Annemieke; Kemman, Max; Birkholz, Julie M. u. a.: State of the Field: Digital History, in: History 105 (365), 04.2020, S. 291–312. Online: <<https://doi.org/10.1111/1468-229X.12969>>, Stand: 15.09.2022.
- Wilkinson, Mark D.; Dumontier, Michel; Aalbersberg, IJsbrand Jan u. a.: The FAIR Guiding Principles for scientific data management and stewardship, in: Scientific Data 3 (1), 03.2016, S. 160018. Online: <<https://doi.org/10.1038/sdata.2016.18>>, Stand: 09.11.2022.