Generierung von Trainingsdaten für die Handschrifterkennung aus TEI annotierten Dokumenten – Ein Erfahrungsbericht aus dem EU-Projekt READ

Maximilian Bryan[1], Tobias Hodel[2], Nathanael Philipp[1]

Automatische Sprachverarbeitung, Universität Leipzig[1]
Staatsarchiv des Kantons Zürich[2]

Zusammenfassung

Zum Trainieren maschineller Lernverfahren zur Erkennung von Handschriften werden Textdaten mit korrespondierenden Bildern benötigt. Die Textdaten liegen häufig im TEI-Format das diverse Möglichkeiten eröffnet, um textuelle und semantische Phänomene auszuzeichnen, weiter können gar eigene Tags oder Auszeichnungsarten eingeführt werden. In diesem Beitrag wird ein im EU-Projekt READ entwickeltes parametrisierbares Tool beschrieben, das mit unterschiedlichen Auszeichnungsstilen in TEI umgehen kann und Textdateien auf Seitenbasis liefert, die zur Zuordnung von Text zu Bilddaten (text-to-image) genutzt werden können und somit zur Aufbereitung von Trainingsdaten für Modelle der Handschriftenerkennung dienen. Die gezeigten Beispiele und Anwendungen stammen alle aus Projekten, die ihre Daten für READ zur Verfügung stellten.

1 Einleitung

Im Rahmen des EU-Projekts READ[1] werden Werkzeuge entwickelt, die es ermöglichen, historische handschriftliche Dokumente, wie sie in Archiven und anderen Erinnerungsinstitutionen vorliegen, les- und suchbar zu machen. Die Dokumente, die bearbeitet werden sollen, liegen dabei in unterschiedlichen Bildformaten vor. Nach diversen Verarbeitungsschritten, die technisch unterschiedlich unterstützt werden oder manuelles Eingreifen erfordern, ist das Endresultat ein transkribierter Text, der eng mit dem Digitalisat verknüpft ist und systemintern als PAGE XML (Pletschacher 2010) vorgehalten wird. Mögliche Prozessschritte können dabei - wie bei der OCR (*optical character recogniton*) – grafische Vorverarbeitung wie Kontrastierung oder das Erkennen von Zeilen und Tabellen sein. Der zentrale Schritt der Texterkennung wird in READ mit maschinellem Lernen durchgeführt. Dieses Verfahren profitiert von Trainingsdaten, welche in diesem Fall Bildausschnitte (auf Zeilenbasis) mit korrespondierendem Text sind. Für die Generierung von Ground-Truth Daten zum Training von Handschriftenmodellen können auch bereits bestehende Editionen und Transkriptionen genutzt werden. Als Format für die Umschriften der Texte (Transkriptionen) wird häufig XML nach TEI[2] verwendet.

[1] Recognition and Enrichment of Archival Documents, https://read.transkribus.eu/. Dieses Projekt wird von der Europäischen Union im Rahmen des Forschungs- und Investitionsprogramms *Horizion 2020* gefördert unter dem Grant Agreement Nr. 674943.

[2] Text Encoding Initiative, http://www.tei-c.org/index.xml

Veröffentlicht durch die Gesellschaft für Informatik e.V. 2018 in
M. Burghardt & C. Müller-Birn (Hrsg.): INF-DH 2018 – Workshopband, 25. Sept. 2018, Berlin
 https://doi.org/10.18420/infdh2018-11

Die Text Encoding Initiative (TEI) bemüht sich seit den späten 80er Jahren, um die Definition von Grundsätzen zur Aufbereitung von Texten, die unter geisteswissenschaftlichen Fragestellungen ausgewertet werden sollen. Erst in SGML heute in XML hat sich TEI als zentrale Anlaufstelle zur Codierung textueller und visueller Phänomene herausgebildet. Der Standard TEI wurde entsprechend geprägt durch Expertinnen und Experten aus geisteswissenschaftlichen Disziplinen, die sich mit der Beschreibung von Texten beschäftigen. Insbesondere aus den Editionswissenschaften, die vorwiegend durch die Geschichts-, Literatur- und Sprachwissenschaften geprägt wurden, sind Traditionen und Begrifflichkeiten übernommen worden.

Obwohl durch die Entmaterialisierung bzw. Digitalisierung von Editionen, Einschränkungen des Drucks überwunden wurden, wird die jahrhundertealte Tradition der Textedition durch die TEI weitergetragen.[3] Für die Generierung von Ground-Truth-Daten für die Handschrifterkennung aus TEI annotierten Dokumenten ist dies insofern problematisch, als dass Textannotationsprojekten mit TEI eine Vielzahl von Annotationsmerkmalen zur Verfügung stehen, die nicht immer konsistent verwendet werden.[4] Entsprechend unterschiedlich sind die bearbeiteten Texte in der Nutzung von Textauszeichnunen (*tags*), nicht zuletzt da die Perspektiven auf Texte abhängig von den geisteswissenschaftlichen Fragestellungen divergieren können.

Im Rahmen dieses Artikels soll zunächst das Forschungsprojekt READ vorgestellt werden. Anschließend wird der Fokus auf den Teilbereich der Aufbereitung der XML-Daten gelegt, da diese die Grundlage darstellen, um die Modelle der Texterkennung mit rekurrenten neuronalen Netzen zu trainieren. Der Fokus wird dabei auf die bereits angedeutete Problemstellung der inkonsistenten Auszeichnungsformen gelegt.

2 Transkribus

Innerhalb des EU-Projektes READ und dem Vorgängerprojekt *tranScriptorium* wurde die Plattform *Transkribus* entwickelt (Mühlberger, 2015). Ziel der Plattform ist das automatische Erkennen von Texten in historischen, handgeschriebenen Dokumenten. Um dies zu erreichen, werden verschiedene Werkzeuge bereitgestellt, welche unterschiedliche Stadien der Texterkennung abdecken:

Zur **Layoutanalyse** gehören Werkzeuge, die grafische Eigenschaften der Dokumente berücksichtigen. Hierzu gehört das Erkennen von Textregionen, Tabellen und Zeilen. Vor allem Letzteres ist für die darauffolgende Texterkennung wichtig ist.

Die **Handschrifterkennung** (HTR, *handwritten text recogniton*) basiert auf maschinellem Lernen mit neuronalen Netzen (Leifert et al., 2016) und funktioniert unabhängig von Schriftsystem oder Sprache. Einzige Voraussetzung ist, dass genügend Beispiele (Trainingsmaterial als Ground-Truth) vorhanden sind, aus denen das Modell lernen kann; als Richtwert wird eine Mindestanzahl dreißig (idealerweise hundert) bereits transkribierten Seiten vorgegeben. Der Unterschied zu OCR ist dabei, dass die OCR bei gedrucktem Text sich auf regelmäßigere Buchstaben verlassen kann und bereits mit sehr wenig Kontext eine gute Erkennung möglich ist. Handgeschriebener Text hingegen weist viel unregelmäßigere optische Merkmale auf. Hierbei ist es deswegen hilfreich, Text nicht Ausschnitts-, also Zeichenweise, sondern Zeilenweise zu erkennen, da so bei der Erkennung von Wörtern auch auf vorangehende oder nachfolgende Zeichenfolgen als zusätzliche Kontextualisierung zurückgegriffen werden kann. Neben großen Menge an Trainingsdaten profitieren die maschinellen Lernverfahren zusätzlich von Wörterbüchern. Bei Wörtern, bei denen sich das Verfahren unsicher ist, können erkannte Wörter mit einem Wörterbuch abgeglichen werden. Wenn nun beispielsweise ein Wort undeutlich geschrieben ist oder die Endung abgekürzt bzw. nur angedeutet worden ist, kann durch ein Wörterbuchabgleich die Erkennungsrate deutlich erhöht werden.

---

[3] Siehe dazu: Sahle, Patrick: Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels. Teil 3: Textbegriffe und Recodierung., Bd. 3 / 3, Norderstedt 2013 (Schriften des IDE 9). Online: <http://kups.ub.uni-koeln.de/5352/>, Stand: 25.07.2014, S. 111-170.

[4] Aus diesem Grund hat das Deutsche Textarchiv ein eigenes eingeschränkteres Tagset in TEI vorgegeben und als Schema publiziert.

Vollständig transkribierte Dokumente können anschließend (als PDF, Textdokument oder TEI XML) **exportiert** oder auf einer **Webplattform** veröffentlicht werden. Hier können nun Nutzer die Dokumente betrachten und durchsuchen. Zusätzlich kann die Transkription des Modells bewertet, indem Seiten korrekturgelesen und Fehler korrigiert werden können. Die dabei entstehenden Daten lassen sich in die bestehenden Trainingsdaten mit integrieren, um Modelle schrittweise zu verbessern.

Eine Mischung aus Layoutanalyse und Handschriftenerkennung ist das so genannte Text-to-Image. Ein Tool, das in Transkribus implementiert wurde, um die Zuordnung von bereits transkribiertem Text zu Bildausschnitten zu ermöglichen. Dabei wird erst jede einzelne Seite segmentiert und anschließend mit einem möglichst passenden Handschriftenmodell erkannt. Während der Erkennung wird abgeglichen, ob eine erkannte Textzeile mit einer definierten Wahrscheinlichkeit mit einer vorgegebenen Transkriptionszeile bzw. einem Teil des Seitentexts übereinstimmt. Im Falle einer Übereinstimmung wird die vorgegebenen Transkriptionszeile eingefügt. Somit lässt sich innert kurzer Zeit bereits transkribierter Text mit Bildern alignieren. Als Eingabe wird neben den Bilddateien eine Textdatei (TXT) pro Seite erwartet.[5] Das Text-to-Image ist parameterisierbar, d.h. es kann unter anderem festgelegt werden wie hoch die Konfidenz bei der Übereinstimmung ist, oder ob Zeilen übersprungen werden können (etwa bei Problemen bei der Layoutanalyse) und schliesslich, ob die Zeilen aus dem Ausgangstext übernommen werden soll.[6]

3 TEI

Die im vorigen Abschnitt vorgestellten HTR-Modelle in Form von neuronalen Netzen benötigen Trainingsdaten um eine robuste Erkennung zu ermöglichen. Diese Trainingsdaten bestehen aus Eingabedaten in Form von Bildern und textnah codierten Transkriptionen. Diverse bestehende Kollektionen mit Bilddateien historischer Handschriften wurden bereits transkribiert und liegen im XML-Format vor. Hierbei wird meistens auf den TEI-Standard zurückgegriffen.

Da die TEI die Bedürfnisse aller edierenden Wissenschaftszweige abdecken wollte und will, bestehen grundsätzlich mehrere Möglichkeiten, um Phänomene zu markieren und auszuzeichnen. Das Format teilt sich in zwei Teile, einen *header* und einen *body*. Der *header* hat zum Ziel Metadaten wie Titel, Autor oder Erscheinungsdatum aufzunehmen und zu strukturieren, wobei nicht nur die digitale Datei, sondern auch die physische Ausprägung des zu edierenden Werks darin mitgemeint wird. Im *body* wird der Text des gewählten Dokuments widergegeben, wobei eine Vielzahl von Strukturmerkmalen wie Seiten, Absätze oder Zeilen gekennzeichnet werden können. Dadurch, dass mehrere Dokumentseiten in einer XML-Datei vorkommen können, ist es nicht möglich lediglich den Textinhalt der Datei zu extrahieren. Außerdem würden hierbei Annotationen und Zeilenumbrüche verloren gehen. Der Text selbst kann im Falle von Korrekturen oder Anpassungen markiert werden; durchgestrichene oder ungebräuchliche Wörter lassen sich ebenso markieren. Weiter ist es auch möglich semantische Merkmale zu annotieren: Namen können Personen, Orten oder Organisationen zugewiesen werden, Abkürzungen lassen sich kennzeichnen und mit Auflösungen versehen, unklare Wörter mitsamt Varianten aufnehmen. Die Art und Weise wie dies geschieht basiert zwar auf einem definierten Tag-Set. Aufgrund der Breite des Sets und der unendlichen Vielfalt an Textphänomenen, die potentiell durch TEI Tags abgedeckt werden, kann dasselbe Phänomen auf mehrere Arten markiert werden kann, ein Problem auf das bereits mehrfach hingewiesen wurde:

> TEI's strength is also its weakness. The Guidelines try to offer suggested markup for every conceivable task of scholarly editing. For any one task the Guidelines sometimes offer multiple possibilities, among which editors may choose for the purposes of a particular project. They really are, in other words, guidelines, and not a strict technical standard. Be very cautious if someone naively asserts that documents from two different editors can

[5] T2I wurde durch Gundram Leifert (Universität Rostock, CITlab) entwickelt. Seine Dissertation zu diesem Thema ist in Vorbereitung.

[6] Zu den Parametern des T2I-Tools siehe im Transkribus Wiki: https://transkribus.eu/wiki/index.php/Text2ImageParameters.

> easily be used together because both comply with the TEI Guidelines: there is plenty of room within the TEI for TEI-compliant texts to be incompatible in various ways.[7]

Die Problematik wird noch virulenter, da aufgrund der XML-Struktur keine oder nur sehr unbefriedigend, überschneidende Hierarchien abgebildet werden können.[8] Ein weiteres Problem ist die Tatsache, dass im Rahmen von Retrodigitalisierung bzw. *double-keying* Vorgängen, die TEI Struktur zwar übernommen, die Textauszeichnung jedoch nicht bis ins Details fortgeführt wird. Es werden entsprechend Konventionen aus dem Druckzeitalter, bspw. Abkürzungsauflösungen in runden Klammern, übernommen und nicht in entsprechende Tags verpackt. Parallel dazu werden in größeren Transkriptionsunternehmen vereinfachte Regeln eingeführt, die sich etwa durch Freiwillige rasch tippen lassen, ohne daß eine Einführung in TEI oder ähnliches notwendig wird.

4 Ein Werkzeug für die Extraktion von Text aus TEI

Wegen der Vielfalt von TEI-Annotationen ist die Extraktion von TEI-XML-Dateien problematisch, sobald die Extraktionsroutine auf TEI-XML aus unterschiedlichen Projekten angewandt werden soll.[9] Im Rahmen von READ wurde aus diesem Grund ein Tool entwickelt, das verschiedenste Annotationsstile und Dokumentenstrukturen verarbeiten kann und insofern parametrisierbar ist, als dass Eigenheiten eines Dokumentes mitgeteilt werden können.

Das Tool wurde mit Java 8 entwickelt. Es können entweder einzelne Dateien oder ganze Ordner verarbeitet werden. Im letzteren Fall werden die Ordner nach zu verarbeitenden Dateien durchsucht und dann einzeln verarbeitet. Falls eine ZIP-Datei vorliegt, wird diese zunächst entpackt und dann nach Dateien durchsucht. Alle Dateien werden in einem Ausgabeordner gespeichert. Entweder wird für jede Eingabedatei eine korrespondierende Ausgabedatei erstellt, wobei die zugrundeliegende Ordnerstruktur beibehalten wird, oder, falls entsprechend vorhanden, werden Dateiinformationen aus der XML-Struktur entnommen und die Dateien dementsprechend danach erstellt.

Bei der eigentlichen Extraktion werden die Dateien nach Textknoten durchsucht. Über die entsprechende Liste wird anschließend iteriert, wobei von jedem Knoten der Textinhalt extrahiert und weiterverarbeitet wird. Innerhalb des Textes vorkommende XML-Tags werden abhängig davon verarbeitet, ob diese bekannt sind oder nicht. Bekannte Tags können Abkürzungen sein. Hier wird dann entweder die Abkürzung oder die expandierte Version ausgewählt und die XML-Struktur verworfen. Bei unbekannten Tags wird der Text extrahiert und die Markierung verworfen, ein Beispiel hierfür wäre: <persName>Name</persName>. In diesem Fall wird das Tag verworfen und lediglich das Wort „Name“ in der Textdatei zu finden sein.

5 Parametrisierungen

Im Folgenden sollen Parametrisierungen des entwickelten Tools beschrieben werden, welche gesetzt werden können, um den verschiedenen Eigenheiten der Formate gerecht zu werden. Dabei soll berücksichtigt werden, dass diese Eigenheiten nicht immer nur verschiedene TEI-Varianten beschreiben, sondern auch Auszeichnungen, die innerhalb eines TEI-Dokumentes verwendet werden, jedoch einem alten Standard

---

[7] Christopher Blackwell, Neel Smith: XML Markup, URL: http://www.homermultitext.org/summer2014/reading/xmlmarkup.html.

[8] Siehe dazu: Bruder, Daniel; Teufel, Simone: Data Models for Digital Editions: Complex XML versus Graph Structures, in: Vogeler, Georg (Hg.): Konferenzabstracts. 5. Tagung des Verbands Digital Humanities im deutschsprachigen Raum e.V., DHd 2018: Kritik der digitalen Vernunft Universität zu Köln, 26. Februar bis 2. März 2018, Köln 2018.

[9] Aus diesem Grund hat das Deutsche Textarchiv ein eigenes eingeschränkteres Tagset in TEI vorgegeben und als Schema publiziert.

entsprechen. Das Tool mit den Parametrisierungen lässt sich entweder von der Kommandozeile aufrufen. Es kann jedoch auch von anderen Programmen als Bilbiothek eingebunden werden.

Abkürzungen werden innerhalb von TEI durch ein choice-Tag umschlossen und bestehen aus Abkürzung und Auflösung der Abkürzung, beispielsweise:

```
<choice><abbr>It</abbr><expan>Item</expan></choice>
```

Andere meist ältere Standards versehen Abkürzungen mit runden Klammern, bspw. It(em). Parametrisierbar ist nun, welche Form der Abkürzungsauszeichnung im vorliegenden Dokument verwendet wird und wie mit Abkürzungen umgegangen werden soll, konkret ob die abgekürzte oder expandierte Form extrahiert werden soll. Die Frage ob aufgelöste oder gekürzte Form verwendet werden soll, ist für die spätere Anwendung zum Training von Modellen zur Handschriftenerkennung von eminenter Bedeutung. Da Modelle möglichst konsistent trainiert werden müsse und stillschweigende Auflösung von Abkürzungen meist schlechtere Resultate beim Training ergeben.

Dateinamen von Bilddateien werden benötigt, um die Textabschnitte eines Dokumentes mit der optischen Darstellung zu verbinden. Häufig werden Dateinamen im <pb>-Tag (steht für *page break*) angegeben. Dies enthält dann das Attribut *facs* mit dem Hinweis auf die Datei sowie oft noch das Attribut *n*, welches die Seitenzahl enthält. Beispielhaft könnte dies so aussehen:

```
<pb n="17" facs="Z148514201\00000017.jpg"/>
```

Innerhalb des <pb>-Tags ist nun häufig der entsprechende Tag enthalten. Es gibt jedoch häufig auch Dokumente, die den Text nicht im <pb>-Tag enthalten sondern dies leerlassen und danach noch ein <p>-Tag einfügen, welches dann den Text enthält. Der Unterschied liegt daran, das in ersterer Variante das <pb>-Tag eigenständig Text einer Seite inklusive der Metainformationen enthält, sodass beim Auslesen die Reihenfolge keine Rolle spielt. Bei letzterer Variante ist beim Extrahieren des Textes zwingend darauf zu achten die Reihenfolge einzuhalten, da <pb> lediglich einen neue Seite ankündigt (page break), der Text jedoch im Nachfolgenden <p>-Tag enthalten ist.

Zeilenumbrüche sind in den Editionsdaten teilweise explizit ausgezeichnet (durch <lb />), bei Übernahme aus alten Editionen oder Transkriptionen jedoch manchmal als Doppelslash (//) gekennzeichnet, in einem dritten Fall aber auch ignoriert. Insgesamt eröffnen sich unterschiedliche Herangehensweisen, da die Text-Bild-Alignierung auch selbständig Zeilenumbrüche generieren kann.

Durchstreichungen können gemäß TEI folgendermaßen gekennzeichnet werden:

```
<del rend="strikethrough">TEXT</del>
```

Manche Dokumentsammlungen benutzen jedoch ein nicht-XML-Auszeichnungsformat, indem die durchgestrichenen Wörter mit jeweils zwei Gleichheitszeichen umrandet werden (==TEXT==). Parametrisierbar ist zum einen, welche Form vorliegt und wie mit ihr umgegangen werden soll, konkret ob der durchgestrichene Text extrahiert werden soll oder nicht. Da Durchstreichungen häufig auf unsicheren Lesungen basieren, können diese Teile auch ausgeschlossen werden, sodass sie nicht in Trainingssets aufgenommen werden, die entsprechenden Zeilen werden dann nicht zugeordnet und verfälschen das Training nicht.

Personennamen lassen sich beispielhaft folgendermaßen kennzeichnen:

```
<persName>Prick</persName>
```

Editionen, die sich nur lose an die TEI anlehnen, benutzen teilweise folgendes Format:

```
<name type=”person”>Prick</name>
```

Wenn nun Wörterbücher erstellt werden sollen von Namensangaben, muss geprüft werden, welches Annotation im vorliegenden Dokument verwendet wurde.

6 Ausblick

Für die getesteten TEI-XML-Dateien funktioniert das Tool bereits stabil. Als nächstes wird das Tool um die Funktion zur Extraktion von Worddateien erweitert. Als DOCX (Microsoft Word XML-Format) vorliegende Dateien sind ebenfalls in Form von XML-Dateien auslesbar und können ähnlich behandelt werden. Informationen zum Textsatz (wie Fettschreibung oder Kursivierung) werden ignoriert. Markierungen von Seitenumbrüchen können als Parameter eingefügt werden. Da eine Vielzahl an Editionen in den letzten zwei Jahrzehnten zwar digital erarbeitet wurden (meistens in Microsoft Word oder einem anderen Wordprozessor), eröffnet dieser Extraktionsschritt das Training einer Vielzahl weiterer Dokumente und Textserien.

Die in den Dokumenten annotierten Abschnitte wie Personen- und Ortsnamen werden bisher ignoriert. Durch die Umstellung des Workflows lassen sich unter Berücksichtigung aller Annotationen automatisiert verschiedene Register für Abkürzungen, Personen, Orte oder ähnliches erstellen, diese wiederum könnten als spezialisierte Wörterbücher eingesetzt werden, die die Erkennung problematischer Strings (häufig Eigennamen und Abkürzungen) substantiell verbessern werden. Das Tool befindet sich zudem in laufender Entwicklung, da sich in jeder weiteren Dokumentsammlung weitere Eigenheiten entdeckt werden.

7 Zusammenfassung

Im Rahmen dieses Artikels wurde die Problemstellung des Extrahierens von Text aus TEI-Dateien vorgestellt, die daraus resultieren, dass das Format mehrere Varianten der Kennzeichnung derselben Phänomene zulässt oder dass eine nicht standardkonforme eigene Auszeichnungsvariante eingeführt wurde. Der Extraktionsalgorithmus erlaubt es im großen Stil, bereits vorliegende Transkriptionen zu extrahieren und in einem nächsten Schritt mit Bilddaten zu alignieren. Damit wird es möglich, effizient qualitativ hochwertige Handschriftenmodelle zu erzeugen.

Literaturverzeichnis

Pletschacher, Stefan; Antonacopoulos, Apostolos: The PAGE (Page Analysis and Ground-Truth Elements) Format Framework, in, 2010, S. 257–260. Online: <https://doi.org/10.1109/ICPR.2010.72>.

Blackwell, Christopher; Smith, Neel: XML Markup, Online: <http://www.homermultitext.org/summer2014/reading/xmlmarkup.html>.

Sahle, Patrick: Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels. Teil 3: Textbegriffe und Recodierung., Bd. 3 / 3, Norderstedt 2013 (Schriften des IDE 9). Online: <http://kups.ub.uni-koeln.de/5352/>.

Leifert, G., Strauß, T., Labahn, R. (2016). Cells in Multidimensional Recurrent Neural Networks, Online: <https://arXiv.org/abs/1412.2620v02>

Mühlberger, G. (2015). Die automatisierte Volltexterkennung historischer Handschriften als gemeinsame Aufgabe von Archiven, Geistes- und Computerwissenschaftlern. Das Modell einer zentralen Transkriptionsplattform als virtuelle Forschungsumgebung. In: Becker, Irmgard; Oertel, Stephanie: Digitalisierung im Archiv. Marburg 2015, S. 87–116 (= Buchveröffentlichungen der Archivschule Marburg 60).