Schlussbericht

Richtlinie zum „Software-Sprint“
DDD: Deutsche Dokumente Digitalisieren

Johannes Filter

30. September 2020

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S18 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

PDF zu Fließtext mit maschinellem Lernen

PDFs sind für Menschen gemacht und nicht für Maschinen (Computer). Das führt dazu, dass wir sie lesen können, aber Maschinen Probleme damit haben, Text zu extrahieren. Dies ist jedoch notwendig, um zum Beispiel große Mengen von PDFs im Rahmen journalistischer Recherchen auszuwerten. Auch Personen mit Seheinschränkungen sind darauf angewiesen, dass Computer ihnen Texte vorlesen. Im Rahmen der bereits erfolgten oder geplanten Digitalisierung deutscher Behörden müssen zudem große Aktenbestände digitalisiert werden.

Text extrahieren mit *pd3f*

Im Rahmen der Prototype-Fund-Förderung von „DDD: Deutsche Dokumente Digitalisieren“ ist die Software-Lösung *pd3f* entstanden, um „guten“ Text aus PDF zu rekonstruieren - „gut“ bedeutet hier, dass der ursprüngliche Text ohne unnötige Zeilenumbrüche wiederhergestellt werden kann. Aus dem zerstückelten Text im PDF wird somit wieder ein digitaler Fließtext. Im Deutschen gibt es viele lange Wörter und deswegen die Besonderheit, dass Wörter am Zeilenende getrennt werden. Bei einer üblichen Textextraktion werden getrennte Wörter jedoch nicht wieder zusammengefügt. Damit kann das ursprüngliche Wort zum Beispiel nicht mehr per Suche gefunden werden. Auch weiterführende Anwendungen, wie zum Beispiel die automatisierte Erkennung von Eigennamen (Named-Entity Recognition), werden erschwert.

Beitrag des Projektes zu den Zielen der Förderinitative „Software-Sprint“

Sämtliche erstellte Software steht unter einer Open-Source-Lizenz. Zudem wurden Kernfunktionalitäten ausgelagert, sodass neue Projekte einfach auf diese Software aufbauen können. Damit wird sichergestellt, dass diese Arbeit der Open-Source-Community zugutekommt. Inhaltlich ist durch diese Förderung die Anwendung von maschinellen Lernen auf deutschen Dokumenten bzw. Texten gefördert worden. Da viel Software nur auf die englische Sprache abzielt, wurde mit der Förderung insbesondere die deutsche Open-Source-Community gestärkt.

Ausführliche Darstellung der Ergebnisse

Automatisierte Texterkennung

Texterkennung auf gescannten Dokumenten (Optical Character Recognition/OCR) erfolgt schon heute zufriedenstellend mit Open-Source-Lösungen. Es ist aber aufgrund des veralteten Portable Document Format (PDF) weiterhin schwierig, die Wörter aus einem PDF zu einem Text zusammenfassen. Das Format folgt der Idee des Druckens, weshalb Fließtext darin nicht als Text dargestellt wird. Stattdessen wird teilweise jeder einzelne Buchstabe als Zeichen kodiert und ihm eine Position (x- und y-Wert für Höhe und Breite) auf dem (virtuellen) Blatt Papier zugewiesen.



Teilweise werden in PDF nur einzelne Buchstaben zusammen gespeichert. Wörter müssen erst aus einzelnen Buchstaben zusammengesetzt werden.

Um aus diesem Buchstabensalat nutzen bestehen Tools Heuristiken, um Buchstaben zu Wörter zusammen zu setzen. Aus Wörtern müssen wieder ganze Zeilen und diese Zeilen anschließend zu Paragraphen zusammengefasst werden. Das ist ohnehin eine schwierige Aufgabe, da es für den Satz eines Textes nahezu endlose Möglichkeiten gibt. Das Open-Source-Tool *Parsr*[1] des französischen Versicherungskonzerns Axa sorgt bereits für Besserung, denn es zerlegt relativ erfolgreich ein PDF in Zeilen und Absätze. Das Tool ist wenige Monate vor Start der Projektförderung erschienen und erwies sich in der Projektphase als nützlich. Unsere Software *pd3f* füttert zunächst das PDF in Parsr und nutzt die Ausgabe von Parsr, um darauf aufbauend guten Text wiederherzustellen.

Zeilenumbrüche entfernen

Worttrennung an Zeilenumbrüchen zu entfernen, ist eigentlich eine einfache Aufgabe: Alle Wörter mit einem „-“ am Zeilenende werden mit dem Wort auf der darauffolgenden Zeile zusammengefügt wie in diesem Beispiel.

> … die Bedeutung der finan-
> ziellen Interessen der Union …

[1] `https://github.com/axa-group/Parsr`

Das Wort „finanziellen“ entspricht dem ursprünglichen Text. Es gibt aber auch „-“ am Zeilenenden, das nicht entfernt werden darf, weil es Bestandteil des Worter ist:

> Auch andere EU-
> Staaten, wie bspw. Polen, …

Um an dieser Stelle weiterzukommen, braucht es mehr Verständnis über die deutsche Sprache. Hier kommt maschinelles Lernen in Form von Sprachmodellen zum Einsatz.

Was sind Sprachmodelle? (Language Models)

Bei Sprachmodellen geht es darum, dass ein Computerprogramm neue Wörter auf der Basis bereits genutzter Wörter lernt. Zum Einsatz kommen Sprachmodelle zum Beispiel auf Smartphones bei der Autovervollständigung. Sprachmodelle verinnerlichen die Charakteristiken der deutschen Sprache und können vorhersagen, welche Wörter oder Buchstaben wahrscheinlich als nächstes kommen. So kommt nach „Sehr geehrte“ wahrscheinlich „Frau“ als nächstes.

Solche Sprachmodelle operieren auf ganzen Wörtern oder auch nur auf Buchstaben (um nachfolgende Buchstaben zu erraten).

Texte reparieren mit *dehyphen*

Eine Unterkomponente von *pd3f* ist das Softwarepaket *dehyphen*, welches ebenfalls im Rahmen der Förderung entstand. Es benutzt Sprachmodelle, um zu entscheiden ob ein „-“ am Zeilenende entfernt werden sollte oder nicht. Die Grundidee ist dabei eine Berechnung darüber, welche die wahrscheinliche Möglichkeit ist, zwei Zeilen zu verbinden.

> Auch andere EU-
> Staaten, wie bspw. Polen, …

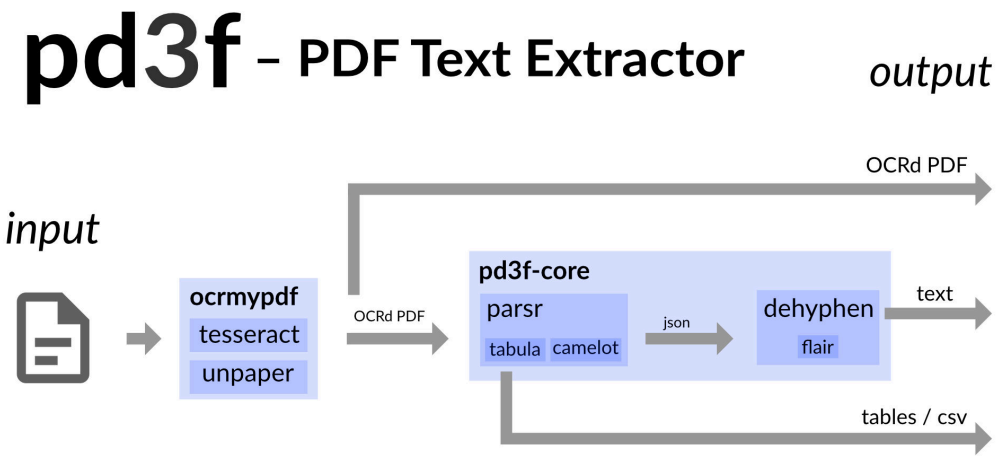
Bei diesem Beispiel kommt *dehyphen* zum richtigen Ergebnis: „EU-Staaten“ ist korrekt, nicht „EUStaaten“. *dehyphen* kann als Modul von anderen Software-Entwickler*innen einfach wiederverwendet werden.

Datenverarbeitungs-Pipeline *pd3f*

Das Hauptergebnis der Förderung ist *pd3f*: Eine komplette Anwendung und eine Datenverarbeitungs-Pipeline für PDFs. Mit ihrer Hilfe können (deutsche) Dokumente digitalisiert werden. Auf einem gescannten Dokument wird der Text automatisiert gescannt, dann mithilfe von Parsr der Text in Wörter, Zeilen und Absätze unterteilt. Anschließend wird mithilfe von *dehyphen* guter Text extrahiert. Anbei eine schematische Auflistung der benutzten Komponenten.

Die Demo von *pd3f*: `https://demo.pd3f.com`.

Der Fokus liegt auf der deutschen Sprache, deren besonderes Charakteristikum lange Wörter sind, *pd3f* kann aber auch für andere Sprachen angewandt werden. Es ist aktuell auch für Englisch, Spanisch und Französisch verfügbar.



Dieses Projekt baut auf einer großen Zahl von anderen Open-Source-Projekten auf.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Die Zielgruppe sind Civic-Tech-Entwickler*innen in Deutschland und andere Personen, die aus Dokumente Texte extrahieren möchte, zum Beispiel im Rahmen journalistischer Recherchen.

Da Dokumente in so vielen unterschiedlichen Formen vorkommen, funktioniert pd3f noch nicht für alle PDFs. Gerade bei schlecht gescannten PDFs ist der extrahierte Text zudem noch stark verbesserungsbedürftig. Es ist unwahrscheinlich, dass pd3f jemals für alle PDFs funktionieren wird, doch es wird weiterhin gearbeitet, die Resultate, zum Beispiel für mehrspaltige Dokumente, zu verbessern.

Was noch fehlt, ist eine systematische Evaluation der Textextrakte von *pd3f*. Diese wird voraussichtlich im April 2021 erfolgen.

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Ja, denn diese Art von experimenteller Produktentwicklung mittels maschinellem Lernen war auch für mich neu. Ich konnte in diesem Projekt eine Vielzahl meiner Fähigkeiten einbringen. So war es zum Beispiel zum Ende des Projekts wichtig die Ergebnisse auf einer eigens gebauten Webseite zu bewerben. Meine anderen Fähigkeiten rund um das maschinelle Lernen und der Python-Programmierung konnte ich ebenso einbringen.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Zunächst wurde mehr auf die Software *Parsr* aufgebaut. Es stellte sich jedoch heraus, dass viele Features von *Parsr* nicht zufriedenstellend. So gibt *Parsr* an, dass die Software auch Überschriften in einem Dokument erkennen kann. Die Genauigkeit dieser Identifikation zeigte sich jedoch als zu gering, um sie für dieses Projekt einsetzen zu können.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzerinnen

Auf der Webseite `https://pd3f.com` gibt es sowohl in Deutsch als auch Englisch eine Einführung in das Projekt. Auf der Webseite ist ebenso ein Video verlinkt, in dem innerhalb eines 20-minütigen Vortrags die Software präsentiert wird. Zudem gibt es auf `https://demo.pd3f.com` eine Demo der Software.

Der Code ist online auf GitHub[2] und wird dort weiter gepflegt. Die Hauptfunktionalitäten von *pd3f* sind zusätzlich in einem eigenen Python-Paket[3] gebündelt, sodass auch hier eine einfach weitere Verwendung als Bestandteil in einem anderen bzw. neuen Python-Projekt möglich ist.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Generell erfolgte die Arbeit dem Plan. Zum Ende des Projekts gab es noch eine Vielzahl von weiteren Möglichkeiten für zukünftige Arbeit. Durch eine Priorisierung von Arbeitspaketen blieben mehrere Pakete übrig (wie zum Beispiel eine systematische Evaluation der Ergebnisse).

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Einige Monate vor Förderbeginn hat der französische Versicherungskonzern Axa eine Open-Source-Bibliothek names *Parsr*[4] zur Dokumentverarbeitung veröffentlich. Diese Veröffentlichung änderte den Verlauf der Entwicklung, denn die Software erwies sich als nützlich und so baut *pd3f* in Teilen auf *Parsr* auf.

[2] `https://github.com/pd3f/pd3f`
[3] `https://github.com/pd3f/pd3f-core`
[4] `https://github.com/axa-group/Parsr`