Programmable Corpora – Die digitale Literaturwissenschaft zwischen Forschung und Infrastruktur am Beispiel von DraCor

Fischer, Frank
frafis@gmail.com
Higher School of Economics, Moskau, Russland

Börner, Ingo
ingo.boerner@univie.ac.at
Universität Wien

Göbel, Mathias
goebel@sub.uni-goettingen.de
WU Wien

Hechtl, Angelika
angelika.hechtl@wu.ac.at
Niedersächsische Staats- und Universitätsbibliothek Göttingen

Kittel, Christopher
contact@christopherkittel.eu
Karl-Franzens-Universität Graz

Milling, Carsten
cmil@hashtable.de
Berlin

Trilcke, Peer
trilcke@uni-potsdam.de
Universität Potsdam

Einleitung

Obwohl sich infrastrukturell einiges getan hat, sieht ein typischer Operationsmodus der digitalen Literaturwissenschaft immer noch so aus, dass eine bestimmte Forschungsmethode auf ein oft nur ephemeres Korpus angewandt wird. Im besten Fall ist das Ergebnis *irgendwie* reproduzierbar, im schlechtesten Fall gar nicht. Im besten Fall gibt es ein offen zugängliches Korpus in einem Standardformat wie TEI, einer anderen Markup-Sprache oder zumindest als txt-Datei. Im schlechtesten Fall ist das Korpus gar nicht zugänglich, d. h., die Forschungsergebnisse müssen einfach hingenommen werden.

Doch seit kurzem gibt es Anzeichen, dass sich dies ändert. Einige Digital-Humanities-Projekte stellen Schnittstellen zu stabilen Korpora zur Verfügung, über die man mannigfaltige Zugriffsmöglichkeiten bekommt und reproduzierbar arbeiten kann. Eines dieser Projekte ist DraCor, eine offene Plattform zur Dramenforschung, die in diesem Vortrag vorgestellt werden soll (zugänglich unter https://dracor.org/ bzw. über die Repos und verschiedene Schnittstellen). DraCor transformiert vorliegende Textsammlungen zu ›Programmable Corpora‹ – ein neuer Begriff, den wir mit diesem Vortrag ins Spiel bringen möchten.

Die Bausteine

Vanillekorpora

Ähnlich wie die COST Action zu europäischen Romanen (Schöch et al. 2018), versucht das DraCor-Projekt als Basis für eine digitale Komparatistik einen Stamm an multilingualen Dramenkorpora aufzubauen, die in basalem TEI kodiert sind. Ein selbst betriebenes russischsprachiges (https://dracor.org/rus) und ein deutschsprachiges Korpus (https://dracor.org/ger) dienen dabei als Einstieg. Diese Korpora sind, ähnlich wie die Sammlung »Théâtre classique« von Paul Fièvre, im weitesten Sinne als Vanillekorpora angelegt, die über das notwendige Markup hinaus zunächst kaum weitere spezielle Auszeichnungen enthalten, allerdings frei zur Verfügung stehen und damit fork- und erweiterbar sind. Zur Demonstration, dass auch andere, reicher kodierte Korpora dazugebunden und sofort alle bereits bestehenden Extraktions- und Visualisierungsmethoden der Plattform angeboten werden können, wurden das Shakespeare Folger Corpus sowie das schwedische Dramawebben-Korpus geforkt und angedockt (https://dracor.org/shake bzw. https://dracor.org/swe). Dramenkorpora in weiteren Sprachen sollen folgen; einzige Voraussetzung dabei ist jeweils, dass diese in TEI vorliegen.

Die Vorteile von frei auf GitHub gehosteten Korpora liegen auf der Hand. Unabhängig von den letztlich durch die Plattform zur Verfügung gestellten Schnittstellen können die Korpora alternativ durch Klonen oder andere Downloadmethoden, etwa über den SVN-Wrapper von GitHub, direkt bezogen und individuell weiterverarbeitet werden. Ein offen zugängliches GitHub-Repositorium heißt auch, dass Pull Requests zur Fehlerkorrektur und Forks für Erweiterungen möglich und erwünscht sind.

XML-Datenbank (eXist-db) und Frontend

DraCor als Plattform setzt auf die eXist-Datenbank, um die TEI-Dateien zu verarbeiten und Funktionen zur Beforschung der Korpora zur Verfügung zu stellen. Das Frontend wurde mit ReactJS gebaut, ist responsiv und einfach erweiterbar. Der Schwerpunkt liegt aber nicht auf der GUI, sondern auf der API (vgl. generell zur Unterscheidung zwischen beiden Schnittstellenansätzen Bleier/Klug 2018).

API und Entwicklungsumgebung

Um dem Ideal und der Möglichkeit nahe zu kommen, auf einfache Weise »alle Methoden auf alle Texte« anwenden zu können (Frank/Ivanovic 2018), braucht es mehr als offene Korpora. Der zitierte Text von Frank/Ivanovic macht sich hinsichtlich dessen für SPARQL-Endpunkte stark; auch DraCor bietet einen solchen an, besitzt darüber hinaus aber eine reiche API, die über Swagger dokumentiert und erläutert wird (https://dracor.org/documentation/api/). In einem Teilbereich der Korpusphilologie, den Digital Scholarly Editions, hat die Diskussion um eine proaktivere Nutzung von APIs bereits begonnen (zur Vorgeschichte vgl. wiederum Bleier/Klug 2018), als Beispiel hierfür diene die Folger Digital Texts API (https://www.folgerdigitaltexts.org/api), über die man sich spezifische Querys zusammenbauen kann. Der Vorteil einer moderneren Lösung wie Swagger besteht darin, dass API-Querys live und direkt ausgeführt und die Outputs genauer kontrolliert und gesteuert werden können.

Ein einfaches Use-Case-Szenario sieht dann so aus, dass man etwa im RStudio mit zwei, drei Zeilen Code einen Blick in ein Korpus werfen kann, etwa über die zeitliche Entwicklung der Anzahl der Charaktere im russischen Drama zwischen 1740 und 1940, die in der Metadatentabelle festgehalten sind (https://dracor.org/api/corpora/rus/metadata). Diese Datei, beziehbar im JSON- oder CSV-Format, wird in eine Data.Table eingelesen, woraufhin die Werte zweier Spalten (Erscheinungsjahre und Number of Speakers) einfach über ggplot visualisiert werden können (Abb. 1).

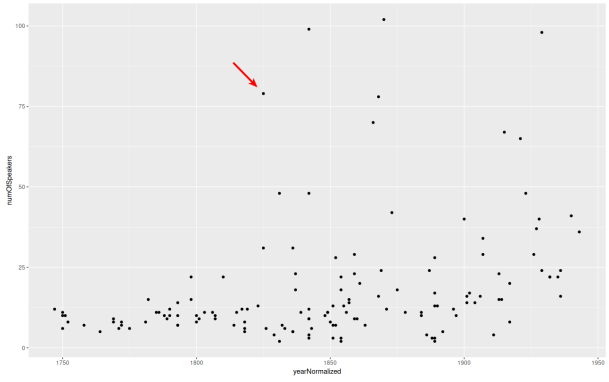

Abbildung 1: Anzahl der Charaktere pro Drama in chronologischer Ordnung (Quelle: RusDraCor).

Anhand dieses sehr simplen Beispiels zeigt sich dann recht deutlich, dass sich mit Puschkins an Shakespeare angelehntem historischen Drama »Boris Godunow« (1825), in dem Sprechakte von 79 Charakteren vorkommen, eine strukturelle Diversifizierung der russischen Dramenlandschaft Bahn bricht.

Die Möglichkeiten beschränken sich aber nicht darauf, vorgefertigte API-Funktionen zu benutzen. Neue Forschungsideen zeitigen immer auch neue Bedarfe an einfach bezieh- und reproduzierbaren Daten und Metriken; die API kann dementsprechend erweitert werden. Dies wird dadurch erleichtert, dass über Apache Ant die gesamte Entwicklungsumgebung auf dem eigenen System nachgebaut werden kann.

Durch bereits implementierte Funktionen können neben Struktur- und Metadaten etwa auch Volltexte ohne Markup bezogen werden (auch Untermengen von Volltexten wie Regieanweisungen), etwa wenn Methoden wie die Stilometrie oder das Topic Modeling der Endzweck sind, also Methoden, die nach dem »bag of words«-Prinzip arbeiten, für das kein Markup vonnöten ist.

Insgesamt wird durch den Aufbau und die Dokumentation offener APIs die bisher oft aufwendige Reproduzierbarkeit von Forschungsergebnissen erheblich erleichtert.

Shiny App

Ein Beispiel für die vielseitigen Nutzungsmöglichkeiten der DraCor-API ist die Shiny App, die Ivan Pozdniakov aufgesetzt hat (https://shiny.dracor.org/). Shiny ist ein auf R basierendes Framework, das es ermöglicht, interaktive Visualisierungen im Browser darzustellen. Die DraCor-Shiny-App tut genau dies und setzt dabei vollkommen auf die DraCor-API für den Datenbezug. So kann zu Lehr- und Forschungszwecken, aber auch zur einfacheren Datenkorrektur, auf Visualisierungen des aktuellen Datenbestandes zugegriffen werden.

Didaxe

Das Markup oder andere Formalisierungen literarischer Texte sind nicht selbsterklärend. Zwar gibt es einige Standards, aber die jeweilige Operationalisierungslösung hängt von der Forschungsfrage ab. Allein das Extrahieren von Figurennetzwerkdaten ist auf viele Arten und Weisen möglich, was dazu führt, dass etwa alle von verschiedenen Forschungsgruppen extrahierten Netzwerke aus Shakespeares »Hamlet« zu leicht verschiedenen Ergebnissen kommen. Selbst für Dramen ist dies also schon ein nicht-trivialer Akt, von Romanen dann ganz zu schweigen (beispielhaft seien Grayson et al. 2016 genannt, die verschiedene Extraktionsmethoden für Romane durchtesten und die Ergebnisse vergleichen). Um diese Erkenntnis schon in der Lehre zu fördern, wurde das Tool »Easy Linavis« (https://ezlinavis.dracor.org/) entwickelt und in die DraCor-Toolchain integriert. Per

Hand können Netzwerkdaten aus Texten extrahiert und dabei das Bewusstsein für die Kontingenz dieses Vorgangs geschärft werden, eine wichtige Vorstufe zur Operationalisierung.

Neben einem Ansatz zur Gamifizierung des TEI-Korrekturvorgangs (Göbel/Meiners 2016) haben wir für Lehrzwecke auch ein Dramenquartett entwickelt, um spielerisch das Verständnis von Netzwerkwerten zu trainieren (Fischer at al. 2018).

Die aufgezählten, um die Plattform herumgruppierten didaktischen Mittel sind integraler Bestandteil des ganzen Projekts, da sie auf dessen Daten und Operationalisierungen aufsetzen. Wichtig dabei war die Erkenntnis, dass Daten mehrere Gestalten annehmen und für Forschung und Lehre gleichermaßen von Bedeutung sein können.

Linked Open Data (LOD)

Im TEI-Code sind PND- bzw. Wikidata-Identifier sowohl für Autor*innen als auch für die Werke hinterlegt. Auf diese Weise lassen sich verschiedene Realien, die außerhalb der eigenen Korpusarbeit liegen, hinzufügen. Eine automatisch erstellte Autor*innengalerie hat dabei noch eher illustrativen Charakter (de la Iglesia/Fischer 2016).

Darüber hinaus kann man aber zum Beispiel feststellen, ob es nicht einen unbewussten regionalen Bias im Korpus gibt. Dafür lässt man sich über die Wikidata-Identifier die Verteilung der Geburts- und Sterbeorte der Autor*innen auf einer Karte anzeigen. So konnte dann für das deutschsprachige Korpus GerDraCor ausgeschlossen werden, dass es einen solchen Bias gibt, da sich die Orte relativ gleichmäßig über die (historisch) deutschsprachigen Gebiete verteilen (Göbel/Fischer 2015).

Ebenso lässt sich über die Wikidata-ID der Stücke herausfinden, wo diese uraufgeführt worden sind (Beispiel-Query: http://tinyurl.com/y9vga68j), d. h., Aspekte der Aufführungsgeschichte lassen sich zuschalten, obwohl diese gar nicht im Fokus des Kernprojekts liegen. Programmable Corpora verbinden sich also auch mit der Welt um sie herum, was sie u. a. von den nach innen gerichteten Workbenches der Korpuslinguistik unterscheidet.

Infrastruktur statt Rapid Prototyping

Projekte wie DraCor versuchen nichts anderes als den digitalen Literaturwissenschaften eine verlässliche und ausbaufähige Infrastruktur zu geben, damit sie sich stärker auf eigentliche Forschungsfragen konzentrieren und reproduzierbare Ergebnisse hervorbringen können.

Eine wichtige Folgerung für uns war, dass wir die Weiterentwicklung unserer seit vier Jahren entwickelten all-in-one Python-Skriptsammlung *dramavis* aufgeben und uns lieber der Arbeit an der API widmen. *Dramavis* (Kittel/Fischer 2014–2018 sowie Fischer et al. 2017) folgte dem in den Digital Humanities nicht untypischen Rapid Prototyping mit direkter Verarbeitung literarischer XML-Daten (Trilcke/Fischer 2018) und einer mittlerweile stark gewachsenen Codebasis, die alles auf einmal kann, deren Maintenance aber immer schwieriger geworden ist und oft genug von den eigentlichen Forschungsfragen weggeführt hat.

Fazit

In Anlehnung an das Projekt »ProgrammableWeb« – das eine Datenbank von offenen APIs unterhält und dessen Slogan lautet: »APIs, Mashups and the Web as Platform« (zugänglich unter https://www.programmableweb.com/) – schlagen wir für infrastrukturell-forschungsorientierte, offene, erweiterbare und LOD-freundliche Korpora den Begriff ›Programmable Corpora‹ vor.

Programmable Corpora erleichtern es, Forschungsfragen auf viele Arten und Weisen um Korpora herum programmieren zu können. Es steht zu erwarten, dass sich infrastrukturelle Anstrengungen dieser Art für die gesamte Community auszahlen mit Effekten, wie sie John Womersley in seiner Präsentation auf der ICRI2018 in Wien aufgezählt hat: a) dramatically increase scientific reach; b) address research questions of long duration requiring pooled effort; c) promote collaboration, interdisciplinarity, interaction.

Der Anschlussmöglichkeiten sind viele, egal ob man gar nicht programmieren möchte, sondern nur eine GEXF-Datei für Gephi benötigt, ob ein Korpus über seine Verbindungen zur Linked Open Data Cloud beforscht oder einfach aus R oder Python heraus bestimmte Daten bezogen werden sollen, ohne dass man sich mit dem Korpus und dessen Maintenance und Reproduzierbarkeit selbst kümmern muss (all dies bleibt natürlich aber eine Option). Programmable Corpora erleichtern die Entscheidung, auf welcher Ebene der eigene Forschungsprozess einsetzt.

Bibliographie

Bleier, Roman / Klug, Helmut W. (2018): *Discussing Interfaces in Digital Scholarly Editing*. In: Digital Scholarly Editions as Interfaces. BoD, Norderstedt, S. V–XV. URL: https://kups.ub.uni-koeln.de/9094/

de la Iglesia, Martin / Fischer, Frank (2016): *The Facebook of German Playwrights*. URL: https://dlina.github.io/The-Facebook-of-German-Playwrights/

Fischer, Frank / Dazord, Gilles / Göbel, Mathias / Kittel, Christopher / Trilcke, Peer (2017): *Le drame comme réseau de relations. Une application de l'analyse automatisée pour l'histoire littéraire du théâtre*. In: Revue d'historiographie du théâtre. # 4. URL: https://hal.archives-ouvertes.fr/hal-01811799

Fischer, Frank / Kittel, Christopher / Milling, Carsten / Schultz, Anika / Trilcke, Peer / Wolf, Jana (2018): *Dramenquartett – Eine didaktische Intervention.* In: Konferenzabstracts zur DHd2018, Universität zu Köln. S. 397 f. DOI: https://doi.org/10.6084/m9.figshare.5926363.v1

Göbel, Mathias / Fischer, Frank (2015): *The Birth and Death of German Playwrights*. URL: https://dlina.github.io/The-Birth-and-Death-of-German-Playwrights/

Göbel, Mathias / Meiners, Hanna-Lena (2016): *Play(s): Crowdbasierte Anreicherung eines literarischen Volltext-Korpus*. In: Konferenzabstracts zur DHd2016, Bern/CH. S. 140–143. URL: http://www.dhd2016.de/abstracts/vortr%C3%A4ge-007.html

Grayson, Siobhán / Wade, Karen / Meaney, Gerardine / Greene, Derek (2016): *The Sense and Sensibility of Different Sliding Windows in Constructing Co-Occurrence Networks from Literature*. In: 2nd IFIP International Workshop on Computational History and Data-Driven Humanities. Trinity College Dublin 2016. PDF: http://derekgreene.com/papers/grayson16chdh.pdf

Kittel, Christopher / Fischer, Frank (2014–2018): *dramavis. Python-Skriptsammlung*. Repo: https://github.com/lehkost/dramavis

Schöch, Christoph et al. (2018): *Distant Reading for European Literary History. A COST Action [Poster]*. In: DH2018: Book of Abstracts / Libro de resúmenes. Mexico: Red de Humanidades Digitales A. C. URL:

Trilcke, Peer / Fischer, Frank (2018): *Literaturwissenschaft als Hackathon. Zur Praxeologie der Digital Literary Studies und ihren epistemischen Dingen*. In: Wie Digitalität die Geisteswissenschaften verändert: Neue Forschungsgegenstände und Methoden. Hrsg. von Martin Huber und Sybille Krämer (= Sonderband der Zeitschrift für digitale Geisteswissenschaften, 3). DOI: http://dx.doi.org/10.17175/sb003_003