Opaque – digitale Arbeitsumgebung für die Humanities

Schlicht, Helene
helene.schlicht@uni-bielefeld.de
Univerität Bielefeld - SFB 1288 Praktiken des Vergleichens, Deutschland

Jentsch, Patrick
p.jentsch@uni-bielefeld.de
Univerität Bielefeld - SFB 1288 Praktiken des Vergleichens, Deutschland

Porada, Stephan
sporada@uni-bielefeld.de
Univerität Bielefeld - SFB 1288 Praktiken des Vergleichens, Deutschland

Opaque – digitale Arbeitsumgebung für die Humanities

Im Rahmen der DHd 2020 Spielräume möchten wir unsere in aktiver Entwicklung befindliche Webanwendung Opaque vorstellen. Anspruch ist es, Opaque als Arbeitsumgebung für DH-Projekte zu etablieren. Die Entwicklung der Webanwendung, deren Funktionen sukzessive erweitert werden sollen, wird im Rahmen des DFG-geförderten Sonderforschungsbereichs (SFB) 1288 "Praktiken des Vergleichens" im Teilprojekt (TP) INF "Dateninfrastruktur und Digital Humanities" durchgeführt.

Das TP INF betreut das Forschungsdatenmanagement des SFB und unterstützt dessen Wissenschaftler*innen darüber hinaus bei der Planung, Konzeptionierung und Durchführung von Forschungsprojekten unter Zuhilfenahme digitaler Methoden. Diese beiden Bereiche sollen in Opaque synthetisiert werden. Aufbauend auf den Erfahrungen der Kooperationen entwickeln wir Opaque zur Bündelung und Automatisierung der erprobten Workflows und Best Practices. Eine besondere Schwierigkeit ist hierbei die Heterogenität und Komplexität von Forschungsdaten in den Geisteswissenschaften. Um dieser Schwierigkeit zu begegnen orientiert sich unsere Etablierung von Best Practices an den verschiedenen Stadien des Data Life Cycle, bestehend aus Planung/ Beratung, Sammlung, Datenorganisation, Datenanalyse, Dissemination und Nachnutzung, und hat zum Ziel, für alle diese Stadien Best Practices zu entwickeln oder implementieren und so den Forscher*innen verfügbar zu machen. Die einzelnen in Opaque verfügbaren Funktionen werden durch etablierte Open Source-Lösungen realisiert, die durch die modulare Konstruktion der Webanwendung nicht nur gut erweitert sondern auch beständig auf dem neuesten Stand gehalten werden können, sowie reproduzierbare Routinen gewährleisten. Der Fokus auf Nachnutzung bestehender Software ermöglicht es uns, ein breites Spektrum an Funktionalitäten in Opaque zu integrieren.

Opaque: Die Webanwendung

Opaque bündelt verschiedene Werkzeuge und Services, die Geisteswissenschaftler*innen Methoden der DH an die Hand geben und somit deren verschiedene individuelle Forschungsprozesse unterstützen können. Mittels Opaque können Forschende digitalisiert vorliegende Quellen einer *Optical Character Recognition* (OCR) unterziehen. Die daraus resultierenden Textdateien können anschließend als Datengrundlage zum *Natural Language Processing* (NLP) weiterverwendet werden. Die Texte werden hierbei automatisiert verschiedenen linguistischen Annotationen unterzogen. Die via NLP prozessierten Daten können in der Webanwendung anschließend als Corpora zusammengefasst und mittels eines *Information Retrieval System* durch komplexe Suchanfragen analysiert werden. Der Funktionsumfang der Webanwendung wird zudem anhand der Bedarfe der Forschenden sukzessive erweitert.

Die Funktionsschwerpunkte von Opaque unterscheiden sich von anderen deutschen DH-Softwareentwicklungen. Hervorzuheben sind *TextGrid*, *FuD* und *CQPweb*, die einen ähnlichen Anspruch als virtuelle Forschungsumgebung verfolgen. Im Unterschied zu Opaque legen *TextGrid* und *FuD* ihre Schwerpunkte auf händische Datenaufbereitung und nachhaltige Speicherung via integrierter Publikationsplattformen, wohingegen *CQPweb* ein Werkzeug zur Korpusanalyse darstellt, dessen Query Processor in Opaque übernommen wurde. Opaque soll demgegenüber keine Publikationsplattform integrieren, sondern eine automatisierte Aufbereitung und Informationsanreicherung von Forschungsdaten mit anschließender Analyse ermöglichen. Die aufbereiteten Daten und Analyseergebnisse können mittels Exportfunktionen anhand gängiger Standards in offene Dateiformate exportiert und anschließend auf eigens gewählten Publikationsplattformen veröffentlicht werden. Die bereits in Opaque integrierten und beständig auf dem neuesten Stand gehaltenen Funktionen im Bereich des NLP und der OCR grenzen die Plattform von den genannten bestehenden Lösungen ab.[1, 2]

Da Opaque plattformunabhängig konzipiert ist, können die verschiedenen Funktionen von den Wissenschaftler*innen auf beliebigen Endgeräten ohne vorangehende Einrichtung genutzt werden. Alle Funktionen wie z.B. OCR werden innerhalb der Cloud-Infrastruktur ausgeführt, so dass Nutzer*innen selbst keine leistungsfähigen Endgeräte benötigen.

Nutzerorientiertes Design

Die in Opaque implementierten Funktionen und Workflows orientieren sich an den aus unserer Zusammenarbeit im SFB hervorgegangenen Erfahrungen, etablierter Best Practices sowie Vorgaben und Standards des Forschungsdatenmanagements.

Dies führt nicht nur zu besseren Ergebnissen für die Forscher*innen, sondern auch zu einer besseren Datenorganisation mittels anerkannter Standards.

Durch eine Gegenüberstellung soll auf dem Poster anhand der verschiedenen Stadien des Data Life Cycle veranschaulicht werden, wie sich Arbeitsprozesse und -schritte durch die Einführung von Opaque verändert haben. Prägnante Beispiele für diese Gegenüberstellung sind Datensammlung und Datenanalyse. Mit Hilfe der Webanwendung können Forscher*innen eigene Quellen und Texte einem OCR-Prozess unterziehen und die Ergebnisse zeitnah selbstständig hinsichtlich der Güte der Texterkennung evaluieren. Diese Automatisierung der Prozesse in Verbindung mit der intuitiven Bedienoberfläche tragen zu einer erhöhten Autonomie der Forschenden bei. Gleichzeitig macht die Echtzeitverfolgung der Jobstatus die Prozessabläufe transparent und nachvollziehbar. Gespräche, die vorher technischer und organisatorischer Natur waren, können nun gezielter für inhaltliche Diskussionen und Planung der Forschung genutzt werden.

Bezüglich der Qualität der Eingabedateien (z.B. Scans) offerieren wir Hinweise zur bestmöglichen Digitalisierung von Ausgangsmaterialien und orientieren uns an gängigen Standards zur Speicherung und Veröffentlichung von Forschungsdaten (z.B. FAIR), um deren Nachnutzung zu gewährleisten. Dies schließt neben den Forschungsdaten auch die Nachhaltung und Bereitstellung von für den Forschungsprozess genutzter Software in den jeweils genutzten Versionen mit ein, um die Reproduzierbarkeit von Forschungsergebnissen sicherzustellen.

Implementierung

Die Umsetzung beruht auf *Free Open Source Software* und Python. Auf dem Poster werden die Vorteile von Linux Containern in einem skalierbaren Docker-Rechencluster, wie z.B. eine einfache Verwaltung verschiedener Softwareversionen – insbesondere wichtig um Forschungsdaten reproduzieren zu können –, vorgestellt und die einzelnen im Folgenden aufgeführten Module der Plattform näher beleuchtet.

- **Webanwendung**: Die Webanwendung dient als Schnittstelle zwischen Nutzer*innen und Recheninfrastruktur. Hier können Datenaufbereitungen in Form von Jobs gestartet und in Echtzeit verfolgt werden, dabei werden die Jobs automatisch auf das zugrundeliegende Rechencluster verteilt. Das Webinterface bietet außerdem die Möglichkeit über ein *Information Retrieval System* Auswertungen durchzuführen.
- **Daemon**: Agiert im Hintergrund, um die von den Nutzer*innen durch die Webanwendung abgesetzten Befehle und Services umzusetzen bzw. zu verwalten.
- **Datenbank**: Die Datenbank speichert alle Metadaten, die während der Nutzung der Webanwendung anfallen. Als Datenbanksystem wird *PostgreSQL* benutzt.
- **Netzwerkspeicher**: Speichert die von den Nutzer*innen hochgeladenen Dateien sowie die daraus generierten Resultate. Die Netzwerkspeicherlösung garantiert den Servern des Cloud-Rechenclusters gleichermaßen Zugriff auf die zu bearbeitenden Dateien.
- **Services**: OCR und NLP-Dienste werden mittels der state of the art Software *Tesseract OCR* und *spaCy* realisiert. Die Korpusanalyse erfolgt durch eine Anbindung an den *CQP query processor* der IMS Open Corpus Workbench. Jede Ausführung eines Dienstes ist mit einem Job assoziiert, der in einem eigens dafür erstellten Container bearbeitet wird.

Ein zusätzliches Hands-On von Opaque soll zu einem Erfahrungsaustausch einladen.

Fußnoten

1. Eintrag des offiziellen DARIAH-Wikis schildert, dass gängige Funktionen wie ein Lemmatisierer nicht mehr nachinstalliert werden können.
2. Der Abschlussbericht des Projekts TextGrid aus dem Jahr 2012 schildert die Implementierung einer OCR Funktion mittels OCRopus, welche in den aktuellen Versionen nicht mehr zu finden ist.