Universität Basel
Projektbericht
Wanja Gerber, Davide Miotto, Phoebe Misiuk, Gianna Coletti

Übung: Grand Tour gone Digital
FS 2024
Dozentin: Dr. Ina Serif

Projektbericht

Digitale Anreicherung und Geovisualisierung eines frühneuzeitlichen Reiseberichts

Charles Burney, The present state of music in France and Italy (1771)

Inhaltsverzeichnis

1. Einleitung

Dieses Projekt befasst sich damit, Teile des Reiseberichts *The Present State of Music in France and Italy* von Charles Burney aus dem Jahre 1771 digital aufzubereiten, anzureichern, und zu visualisieren.

Da der Text nur als Digitalisat vorhanden ist, wurde in einem ersten Schritt mit *Transkribus* ein Volltext erzeugt. Anschliessend wurde ein Named Entity Recognition-Modell für die NLP-Bibliothek SpaCy trainiert und auf zwei Kapitel des Textes angewandt. Nach der Korrektur und Ergänzung der ausgezeichneten Entitäten konnte eine Geovisualisierung mithilfe von Nodegoat erstellt werden. Der dabei erarbeitete und dokumentierte Workflow kann auf den gesamten Reisebericht von Charles Burney, sowie auch auf andere Texte angewandt werden. In einem öffentlichen Repositorium sind neben der hier vorliegenden Dokumentation jegliche Skripte und Datensets, die im Laufe dieses Projekts verwendet wurden, abgelegt.[1]

[1] https://github.com/WanjaGe/Charles-Burney-digital

2. Transkribus

Der Drucktext des entsprechenden Kapitels, hier zum Kapitel “Venice”, wurde auf *Transkribus* hochgeladen. Mithilfe des öffentlichen Modells “Transkribus Print M1” wurde dann die Texterkennung durchgeführt. Nach der Texterkennung konnte der Text als Volltextdokument exportiert werden, um darauf aufbauend das Named Entity Recognition-Modell zu trainieren und schließlich das gesamte Kapitel so zu annotieren.

3. Named Entity Recognition mit SpaCy

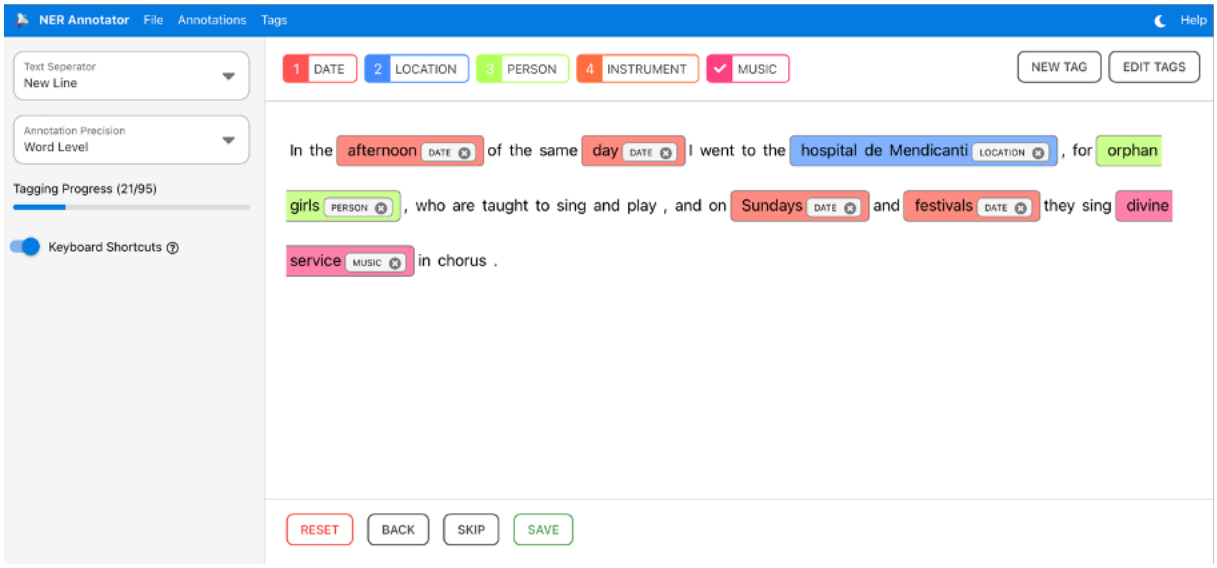
Mit dem Ziel, musikalische Ereignisse geografisch zu visualisieren, wurden fünf Enitäten zur Auszeichnung bestimmt. Diese definieren, welche Informationen schlussendlich quantitativ ausgewertet werden können und stellen die Grundlage das NER-Modell dar. Folgende Entitäten wurden ausgewählt: Date (<date>), Location (<placeName>), Person (<persName>), Instrument (<instrument>) und Music (<music>).

3.1 Preprocessing

Das für die folgenden Prozesse verwendete Python-Skript kann unter **01_preprocessing** eingesehen werden. In der Transkription von *Transkribus* kommt diesers Trennzeichen häufig vor: ¬. Durch das Skript wird das Zeichen im ganzen Text entfernt und die dadurch getrennten Worte werden verbunden. Dann werden in den folgenden Schritten alle Seitenzahlen entfernt und der Text wird so formatiert, dass nach jedem Satzende eine neue Zeile beginnt. Dieser wird schliesslich als neue Datei abgespeichert. Anschliessend werden Zeichen, Sätze und Tokens im Datensatz gezählt, sodass später eine Aufteilung der Daten in Trainings- und Validierungsdaten stattfinden kann.

3.2 Modell trainieren

Nun ist der Text vorbereitet und wird im NER Annotator ausgezeichnet.



Die Auszeichnungen können als json-Datei exportiert werden. Diese werden nun in Trainings- und Validierungsdaten aufgeteilt (80/20). Der Code dazu ist hier zu finden: **02_trainData-evaluationData**. Diese Dateien werden dann in das spacy-Format umgewandelt. Bei diesem Schritt sind anfangs einige Probleme entstanden. Das Skript konnte die Datei nicht umwandeln, da es in den Daten "null"-Werte gab. Nachträglich konnten diese darauf zurückgeführt werden, dass Zeilenumbrüche innerhalb von Sätzen vorhanden waren. Da der NER-Annotator die ausgezeichneten Daten anhand der Struktur von einzelnen Sätzen formatiert, haben diese Zeilenumbrüche zur Entstehung der «null»-Werte geführt. Bei der neusten Version des **01_preprocessing** ist daher ein Skript eingebaut, welches Zeilenumbrüche nur am Ende von Sätzen erlaubt.
Schliesslich kann mithilfe der ausgezeichneten Daten das Training in der Kommandozeile durchgeführt werden.

```
(base) wanjagerber@MBP-von-Wanja downloads % python -m spacy train config.cfg --output ./output --paths.train /Users/wanjagerber/Desktop/finalNER/trainData.spacy --paths.dev /Users/wanjagerber/Desktop/finalNER/valuationData.spacy
```

Der Output hat folgende Form:

```
✔ Created output directory: output
ℹ Saving to output directory: output
ℹ Using CPU

=========================== Initializing pipeline ===========================
✔ Initialized pipeline

============================= Training pipeline =============================
ℹ Pipeline: ['tok2vec', 'ner']
ℹ Initial learn rate: 0.001
E    #       LOSS TOK2VEC  LOSS NER  ENTS_F  ENTS_P  ENTS_R  SCORE
---  ------  ------------  --------  ------  ------  ------  ------
  0       0          0.00     15.64    0.00    0.00    0.00    0.00
  6     200        158.20   2506.56   74.96   71.02   79.37    0.75
 15     400         48.50    614.28   92.45   91.59   93.33    0.92
 25     600         29.84    145.35   92.96   91.67   94.29    0.93
 39     800         23.01     76.11   91.19   88.86   93.65    0.91
 55    1000         28.53     68.38   91.36   88.89   93.97    0.91
 76    1200         28.90     49.65   92.86   90.88   94.92    0.93
101    1400         99.82    114.13   91.69   92.28   91.11    0.92
132    1600         36.08     33.55   90.85   90.28   91.43    0.91
171    1800          9.92      8.11   91.71   90.43   93.02    0.92
217    2000          0.00      0.00   91.71   90.43   93.02    0.92
275    2200          0.00      0.00   91.48   90.91   92.06    0.91
✔ Saved pipeline to output directory
output/model-last
```

3.3 Evaluation

Mithilfe der folgenden Eingabe wird das Modell evaluiert:

```
(base) wanjagerber@MBP-von-Wanja downloads % python -m spacy
package          /users/wanjagerber/downloads/output/model-best
/users/wanjagerber/downloads/package --build wheel
```

Unser Modell schneidet so ab:

```
================================== Results ==================================

TOK      100.00
NER P    90.73
NER R    93.35
NER F    92.02
SPEED    3668


=============================== NER (per type) ===============================

                P        R        F
PERSON      92.13    92.86    92.49
LOCATION    91.14    96.00    93.51
MUSIC       96.97    86.49    91.43
DATE        92.86   100.00    96.30
INSTRUMENT  78.57    97.78    87.13
```

Das Modell weist einen F-Score von 0.92 auf, ein gutes Resultat.

3.4 Anwendung

Nach der positiven Evaluation kann das Modell angewandt werden. In der Datei **03_Anwendung-NER** ist der entsprechende Code für die Anwendung auf ein Kapitel aus Bourneys Reisebericht einzusehen. Dabei wird auch hier im ersten Schritt der Text bereinigt: Entfernung der Striche und Zusammenführung der Wörter, sowie Aufteilung in Absätze. Anschliessend muss die Text-Datei in eine xml-Datei umgewandelt werden, sodass diese wiederum als Grundlage für die NER dienen kann. Nach der erfolgreichen Anwendung können die Auszeichnungen in einer neuen xml-Datei in den Text eingefügt werden.

4. Anreicherung und Korrektur der TEI-XML-Datei

Um alle Elemente eines musikalischen Ereignisses bei der Visualisierung zusammen darstellen zu können, müssen örtlich und zeitlich zusammengehörige Entitäten in „Event“-Elementen zusammengefasst werden. Die Definition einer Textpassage als Ereignis erfolgt durch ein „close reading“ des Textes und erfordert eine qualitative Einschätzung der beschriebenen Situation sowie der dabei involvierten Akteur*innen. Da diese Kategorisierung keinem klaren Muster folgt und nicht regelbasiert ist, kann der Prozess nicht automatisiert werden. Da der Textabschnitt zu Venedig mit 54 Seiten sehr umfangreich ist, beschränkten wir uns hier auf die ersten 17 Events. Zum Auszeichnen der Events wird am Anfang und am Ende des jeweiligen Ereignisses im Text der Tag <event> mit der jeweiligen Nummer als id-Attribut <event id=“event1“> platziert. So kann man Ereignisse spezifisch mithilfe eines Xpather suchen und alle annotierten Elemente innerhalb dieses Events nach Entität geordnet als CSV-Datei exportieren. Das id-Attribut dient auch zur Zuordnung innerhalb der CSV-Tabelle, die mithilfe eines Python-scripts (siehe **04_export-csv**) erstellt wird. Nach dem Export wurden nicht zugehörige Entitäten entfernt. Dabei handelt es sich um rechtmässig erkannte Entitäten, die jedoch keine Relevanz für das darzustellende Ereignis haben.

5. Export und Visualisierung

5.1. Export

Um die ausgezeichneten Elemente der XML-Datei in Nodegoat importieren zu können, müssen diese im csv-Format vorliegen. Zunächst waren einzelne Xpather-Anfragen zur Extraktion der Daten angedacht. Dies würde jedoch beinhalten, die Entitäten jedes Event-Elementes einzeln in eine Tabelle zu übertragen. Schliesslich konnte ein Python-Skript diese Abfragen automatisieren und direkt in eine Datei überschreiben. Die Datei ist hier zu finden: **04_export-csv**.[2]

5.2. Koordinaten

Um die von Burney beschriebenen Orte auf einer Karte darstellen zu könnten, mussten sie vor dem Exportieren der CSV-Datei auf die Plattform Nodegoat mit Koordinaten versehen werden. Dies wurde manuell durchgeführt, da die Verwendung eines Anreicherungsprogrammes wie OpenRefine aufgrund der kleinen Datenmenge nicht sinnvoll war. Falls bei einem zukünftigen Projekt der ganze Reisebericht verarbeitet würde, wäre eine solche Automatisierung jedoch zu empfehlen. In fast allen Fällen konnte der im Text genannte Ort mittels moderner Kartendienste wie Google Maps lokalisiert werden, teils waren auch kurze Recherchen notwendig, wenn Burney abgekürzte Ortsbezeichnungen verwendete. Da es sich bei den Orten, an denen musiziert wurde, meist um Kirchen handelt, die heute noch existieren, stellte die Lokalisation dieser keine allzu grosse Herausforderung dar. Google Maps ermöglichte es, die Koordinaten im Format longitude / latitude zu bestimmen und so die Ortsangaben in der CVS-Datei zu vervollständigen.

[2] Dieses Skript wurde grosszügigerweise von Dr. Elena Spadini zur Verfügung gestellt, wofür ihr die Autor*innen an dieser Stelle danken. Das Skript wurde für die Verwendung im Projekt angepasst.

5.3. Schritte zur Visualisierung auf Nodegoat

5.3.1. Objekt erstellen

Ein Objekt namens „Event“ wurde erstellt. Darunter fünf Sub-Objekte:

- "People" – Beschreibt, welche Personen beim Event anwesend waren.
- "Place" - Name des Ortes, wo das Event stattfand, sowie auch seine Koordinaten (wenn möglich).
- "Date" – Wortlaut des Datums im Text, sowie in einem standardisierten Format (tt/mm/jjjj)
- "Instrument" – Beschreibt, welche Instrumente am Event gespielt wurden.
- "Music" - Beschreibt musikalische Kategorie der gespielten Musik und, wenn möglich, das Musikstück

5.3.2. Projekt erstellen

Das Projekt wurde "The present state of music in France and Italy - Charles Burney" genannt. Das Objekt "Event" wurde zum Projekt beigefügt.

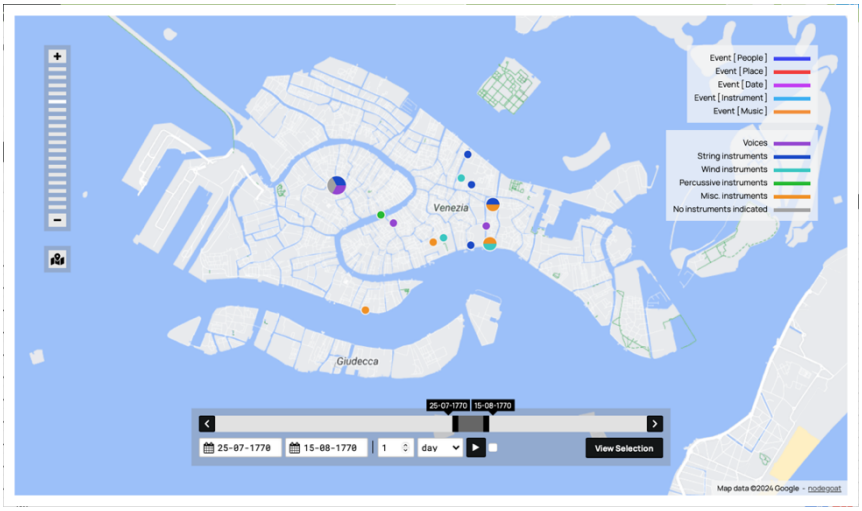
5.3.3. CSV-Datei importieren

Erst wurde die CSV-Datei auf Nodegoat hochgeladen. Daraufhin wurde ein Import Template erstellt und die Spalten in der CSV-Datei mit dem Objekt verknüpft.

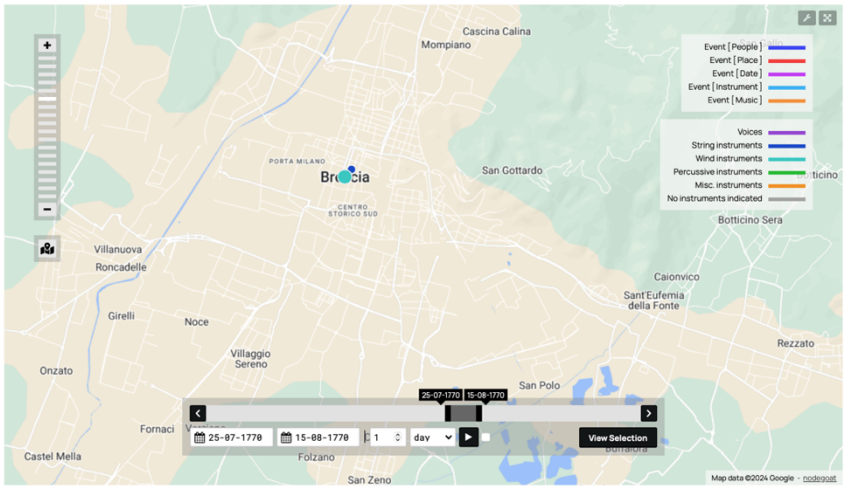
5.3.4. Visualisieren

Die verschiedenen Ereignisse wurden mit Nodegoat geografisch visualisiert.

Erster Versuch einer Visualisierung der Events in Venedig.



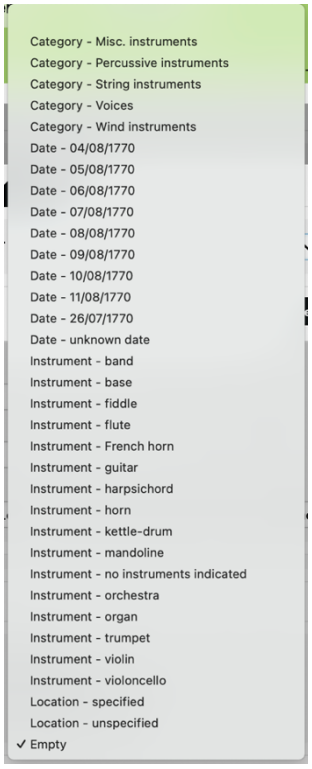
Fertige Visualisierung von Venedig.



Fertige Visualisierung von Brescia.

5.3.5. Filter erstellen

Wir haben für unser Projekt vier verschiedene Kategorien von Filtern erstellt: Instrumente, Instrument-Kategorien, Daten und Orte.



Mögliche Filter.

6. Probleme, Fragestellungen, Ausblick

6.1. Quellensuche

Diese Arbeit ist primär durch das Interesse am Bericht *The Present State of Music in France and Italy* von Charles Burney entstanden und darauf aufbauend entwickelt worden. Im Bericht beschreibt Burney Momente, in denen er während einer Reise durch Frankreich und Italien erlebt hat, wie Musik gespielt wurde. Der Bericht ist nach den Städten gegliedert, die er besucht hat. Die Quelle bot ideale Voraussetzungen zum Erlernen und Anwenden von geographischen Visualisierungen, da im Bericht oft der Ort in Verbindung mit einem jeweiligen Ereignis, an dem Musik gespielt wird, anschaulich erwähnt wird.

6.2. Aufbereitung der Quelle

Zunächst konzentrierten wir uns in allen Arbeitsschritten auf Burneys Berichte während seiner Aufenthalte in Venedig und Brescia. Dabei wurde ein Workflow erarbeitet, der grundsätzlich auf den ganzen Reisebericht, sowie auch auf einen grösseren Quellenkorpus angewandt werden könnte. Die Kategorisierung der ausgezeichneten Entitäten in einzelne Ereignisse geschieht jedoch manuell und ist zeitaufwändig. Daher beschränkt sich dieses Projekt auf die obengenannten Ausschnitte.

Alle 54 Seiten von Burney Schilderungen zu Venedig konnten mit einer niedrigen Fehlerrate transkribiert werden. Als Entitäten, die später mittels Named Entity Recognition ausgezeichnet werden sollten, haben wir *Daten, Personen, Orte, Instrumente* und *Musik* festgelegt. Zudem haben wir Möglichkeiten zum Export der Daten aus dem XML-Dokument besprochen, zunächst schien uns Xpather eine gute Möglichkeit zu sein. Ebenfalls wurde diskutiert, welche Programme zur Visualisierung sinnvoll sein könnten. Storymaps Knightlab wäre eine Option gewesen, erfordert jedoch sehr viel händische Arbeit, weshalb wir uns trotz langer Einarbeitungszeit für Nodegoat entschieden haben.

Ein Problem, was in verschiedenen Arbeitsschritten wiederkehrend auftrat, war die fehlerhafte Auszeichnung von Entitäten durch das NER-Modell. Nebst offensichtlichen Fehlauszeichnungen wurden auch Entitäten ausgezeichnet, die nicht für die Visualisierung benötigt wurden, wie beispielsweise “me” als Person. Dies und Wiederholungen von Entitäten innerhalb eines Events mussten von Hand in der CSV-Datei bereinigt werden. Zudem wurden auch Entitäten ausgezeichnet, die nicht direkt mit dem Event in Verbindung standen, sondern vom Autor als Referenz erwähnt wurden. So mussten auch diese händisch korrigiert werden, da kein automatisierter Lösungsansatz dafür gefunden wurde.

6.3. Visualisierung mit Nodegoat

Als eine zentrale Schwierigkeit bei Nodegoat erwies sich die Animation der Zeitleiste, denn alle Events tauchen, obwohl sie als an unterschiedlichen Daten stattfindend kennzeichnet wurden, gleichzeitig auf der Karte auf.

6.4. Ausblick

Das Trainieren der Named Entity Recognition auf den Text von Burney erlaubt es, weitere Städte zu annotieren und schliesslich zu visualisieren. Somit kann man anhand dieser Visualisierung quantitative Fragestellungen zur Verbreitung von gewissen Instrumenten oder zur Art und Häufigkeit von gespielter Musik durchführen. Schon innerhalb von Venedig sind Vergleiche möglich, beispielsweise in welchen Quartieren am meisten musiziert wird. Weitere Fragestellungen könnten ebenfalls mit zusätzlichem ausgezeichnetem Material behandelt werden:

- Mit welchen Musikinstrumenten wurde Ende des 18. Jh. in einer spezifischen Ortschaft, bzw. zwei Ortschaften im Vergleich oder grundsätzlich in Italien und/oder Südfrankreich musiziert? In welchem Aufführungskontext wurden welche Musikinstrumente verwendet?
- Wie lassen sich Musizierende im 18. Jh. charakterisieren? Welche Rückschlüsse auf die Art und Weise der musikalischen Bildung im 18. Jh. in Italien und/oder Südfrankreich sind möglich?
- An welchen Örtlichkeiten wurde im Italien / Südfrankreich des 18. Jh. musiziert? Welche Orte wurden für welche Art des Musizierens genutzt?