Shakespeare Projekt

Thomas Döring, Lukas Fischer, Lukas Kairies, Robert Terbach, Simon Vetter

14. Oktober 2014

1 Installations- und Startanleitung

Installation

Das Projekt läuft grundlegend mit dem Interpreter Python der Version 3. Falls noch nicht geschehen, kann es mithilfe der Paketverwaltung des Betriebssystems installiert werden, z.B. durch

```
apt-get install python3
```

Python 3 benötigt weiterhin folgende vier Erweiterungspakete:

- Django[1]
- Pandas[2]
- Numpy[3]
- Scikit-Learn[4]

Diese können zum Beispiel durch folgende Befehle installiert werden

```
pip3 install django
pip3 install pandas
pip3 install numpy
pip3 install sklearn
```

Anschließend kann mittels

```
git clone https://github.com/
    DHLeipzig-CurrentTopics-SS2014/shakespeare_project
        .git
```

der aktuelle Projektstand heruntergeladen werden.

[1] https://www.djangoproject.com/
[2] http://pandas.pydata.org/
[3] http://www.numpy.org/
[4] http://scikit-learn.org/

Startanleitung

Das Shakespeare Project läuft als Webservice, daher muss vor der Benutzung ein Webserver aufgesetzt werden. Django bietet in Python 3 einen eingebauten Testserver, der durch den unten stehenden Befehl gestartet werden kann:

```
python3 src/django/manage.py runserver
```

Vor dem ersten Start, muss die Datenbank migriert werden. Notwendig ist es sonst nur, wenn sich die Datenbankstruktur geändert hat, es erscheint in diesem Fall eine entsprechende Aufforderung beim Starten des Servers in der Konsole. Führen Sie dazu folgenden Befehl aus (der Server muss danach neu gestartet werden):

```
python3 src/django/manage.py syncdb
```

Anschließend ist der Webservice über `localhost:8000/corpora/corpora` erreichbar.

Für den Produktivbetrieb sollte der Testserver nicht verwendet werden. Stattdessen sollte ein Webserver (z.B. *apache* oder *nginx*) eingerichtet werden, der die Django-Funktionialitäten als Skript aufruft.

Nun können die zu untersuchenden Corpora in die Datenbank geladen werden. Dazu sollten sie vorher in das folgende einheitliche Format übertragen werden:

```
<xml>
   <author>...</author>
   <year>...</year>
   <title>...</title>
   <text>
      ...
   </text>
</xml>
```

Um diese Arbeit nicht manuell durchführen zu müssen, stehen für die von uns genutzten Corporaquellen Parser bereit, mithilfe derer sie automatisch in dieses Format übertragen werden können. Die Parser liegen dem Projektarchiv bei. Durch das Hilfskript `parse_all.py` werden alle unsere Corpora automatisch in das Format übertragen. Für den Upload der nun formatierten Corpora kann das Webinterface unter `http://localhost:8000/corpora/corpora` genutzt werden.

2 Entwicklung des Projektes während des Semesters

2.1 Forschungsfragen

Zu Beginn des Semesters wählten wir Shakespeare als Thema für unser Projekt. Shakespeare erfand in seinen Texten eine Vielzahl neuer Wörter und belegte bereits existierende Wörter mit neuen Bedeutungen. Als Schwerpunkt unseres Projektes wollten wir untersuchen, welchen Einfluss Shakespeare und seine Wortschöpfungen auf die englische Sprache haben, sowie in diesem Zusammenhang auch erforschen, welche dieser Wörter am häufigsten in englischen Texten genutzt wurden und welche Autoren sich am meisten von Shakespeare haben beeinflussen lassen. Zusammengefasst haben wir uns folgende Fragen gestellt:

1. Wie beeinflusste Shakespeare die englische Sprache?
2. Welche Wörter von Shakespeare wurden am meisten verwendet?
3. Welche Autoren nutzen die meisten Wörter Shakespeares?
4. In welchen Büchern wurden die meisten Shakespeare-Wörter genutzt?

2.2 Sammeln der Shakespeare-Wörter

Um eine Sammlung von Shakespeare-Wörtern zu erstellen, planten wir zunächst einen Corpus, bestehend aus Shakespeares Texten, mit Referenztexten aus dem selben Zeit-

raum (15. und 16. Jahrhundert) zu vergleichen, um so die von Shakespeare eingeführten Wörter zu erhalten. Letztendlich konnten wir diesen aufwendigen Prozess umgehen, indem wir die Wörterdatenbank des Oxford English Dictionary[5] nach Wörtern mit Bezug auf Shakespeare durchsuchten und so eine Sammlung von Shakespeare-Wörtern erstellen konnten. Da Shakespeare nicht nur Wörter, sondern auch Redewendungen in die englische Sprache einführte, wollten wir diese zunächst auch in unsere Betrachtung einfließen lassen. Aufgrund des Mehraufwandes haben wir dies aber nicht umgesetzt.

2.3 Corpus

Als Vergleichstexte für die weitere Betrachtung nutzen wir mehrere Corpora, bestehend aus Texten, Briefen und schriftlich festgehaltenen Unterhaltungen aus den Jahren 1560 bis 1920. Ursprünglich wollten wir Projekt Gutenberg[6] und archive.org als Quellen für diese Texte zu nutzen. Nach genauerer Recherche entschieden wir uns jedoch dagegen, da viele Texte dieser Quellen nicht mit Erscheinungsjahren aufgelistet und daher für unsere Betrachtung ungeeignet sind. Ausserdem sind die Texte in Projekt Gutenberg nicht einheitlich formatiert, sie haben sowohl vor, als auch nach dem Text Abschnitte, die den Text, den Autor, eine Versionsgeschichte, Lizenzdetails, Anmerkungen der Projekt Gutenberg Mitarbeiter, sowie andere verschiedene Informationen beschreiben. Leider sind auch nicht Textautor, Jahr und Titel einheitlich gestaltet, sodass jeder Text hätte einzeln manuell bearbeitet hätte werden müssen. Stattdessen bezogen wir die von uns genutzten Corpora von der *Katholieke Universiteit Leuven*[7] und dem *University of Oxford Text Archive*[8]. Insgesamt besteht unsere Corporasammlung aus mehr als 900 Texten von über 400 Autoren mit insgesamt mehr als 62 Milllionen Wörtern. Die folgenden Corpora wurden gewählt, weil sie eine große

[5] http://www.oed.com
[6] https://www.gutenberg.org
[7] https://www.kuleuven.be
[8] http://www.ota.ahds.ac.uk

Textanzahl haben sowie in für uns interessanten Zeitabschnitten liegen:

- CLMET - The Corpus of Late Modern English Texts, version 3.0[9]
- CEN - The Corpus of English Novels[10]
- The English language of the north-west in the late Modern English period: a Corpus of late 18c Prose[11]
- CED - A Corpus of English Dialogues 1560-1760[12]
- The Lampeter Corpus of Early Modern English Tracts[13]

2.4 Entwicklung unseres Webservices

2.4.1 Grundidee

Während der Bearbeitung unserer Forschungsfragen mit Bezug auf Shakespeare haben wir uns entschieden, einen allgemein einsetzbaren Webservice zu entwickeln und unsere Fragen als einen Anwendungsfall für diesen Webservice zu betrachten. So können wir unsere Fragen beantworten und gleichzeitig einen Service bieten, der unabhängig von Corpora und Wortliste eine Möglichkeit bietet, ähnliche Fragen mit anderem Themenbezug zu beantworten. Ähnlich wie in Google Ngram Viewer[14] soll hier die Relevanz der betrachteten Wörter über die Zeit angegeben, aber auch mehr Funktionalität geboten werden. So lassen sich etwa eigene Corpora verwenden und Fragen in Bezug auf andere Eigenschaften der Texte stellen, zum Beispiel bezüglich des Autors. Da Python sich in wissenschaftlicher Datenverarbeitung momentan durch Bibliotheken wir `pandas` und `scikit-learn` einen Namen macht, ermöglicht die Verwendung dieser bekannten Pakete auch Anderen einfach eigene Funktionen zu implementieren und dem Funktionsumfang hinzuzufügen. Mit einem Webservice wäre

[9] https://perswww.kuleuven.be/∼u0044428/clmet3_0.htm
[10] https://perswww.kuleuven.be/∼u0044428/cen.htm
[11] http://www.ota.ahds.ac.uk/desc/2468
[12] http://www.ota.ahds.ac.uk/desc/2507
[13] http://www.ota.ahds.ac.uk/desc/3193
[14] https://books.google.com/ngrams

man dann auch in der Lage auf einfachem Weg interaktive Ergebnisse der Öffentlichkeit zur Verfügung zu stellen.

In diesem Zusammenhang haben wir geplant, neben dem zeitlichen Verlauf eine geografische Verteilung mit Bezug auf Veröffentlichungsort und Geburtsort des Autors zu visualisieren. Da wir hierfür aber keine hinreichende Datenquelle gefunden haben, verwarfen wir diese Idee.

2.4.2 Methodik

Um Komfortabel und mit benutzerdefinierten Bedingungen performant auf die Corpora zugreifen zu können entschieden wir uns die Texte in einer Datenbank zu hinterlegen. Gleichzeitig wollten wir in der Lage sein auf einfache Art und Weise Ergebnisse zu visualisieren. Da ansprechende Visualisierungen momentan stark im Web verwendet werden entschieden wir uns auch auf entsprechende JavaScript-Bibliotheken zu setzen und mit einem Webframework zu arbeiten. Als dritter Baustein war eine Bedingung die aus dem Seminar bekannten Text-Wort-Matrizen verwenden zu können beziehungsweise die `pandas`-Bibliothek und die TF-IDF funktionalität von `scikit-learn`. Die Schnittstelle dieser Überlegungen erfüllt das Python-Webframework `django` ideal. Django abstrahiert komplexe SQL-Abfragen und versteckt sie hinter Python-Code. So können zum Beispiel alle Texte nach Erscheinungsjahr sortiert, oder gefiltert abgerufen werden.

Da wir mehrere verschieden formatierte Corpora verwenden mussten wir dafür sorgen, dass wir sie gleichartig in unser Datenmodell übertragen können. Wie oben beschrieben haben wir dafür ein eigenes einfaches XML-Format entworfen und Parser für die einzelnen Corpora geschrieben, die sie in das Format umwandeln können. Alternativ hätten wir Parser schreiben können, die jeden Corpus direkt in die Datenbank einpflegen. Da jedoch das interne Datenbankmodell komplexer ist als das XML-Format konnten wir durch ein einheitliches Vorformat sicherstellen, dass alle Texte auf gleichem Weg in die Datenbank überführt werden.

Je nach Bedarf der durchzuführenden Rechnungen kann nun direkt auf den das Datenmodell darstellenden Pythonobjekten gearbeitet werden oder zum Beispiel `pandas-DataFrames` verwendet werden.

3 Probleme und Fehler

Für uns alle sind Fragestellungen aus einem geisteswissenschaftlichen Gebiet Neuland, es fiel uns schon zu Beginn schwer zu entscheiden, was wir für Fragen an den Corpus stellen könnten, noch schwerer allerdings zu entscheiden, wie sinnvoll diese sind und wie schwer zu beantworten. Auch die verwendeten Technologien waren ziemlich neu für uns, so zum Beispiel die vielfältigen Funktionen von `pandas`, oder die Visualisierung mit JavaScript. So haben das Erlernen von und der Umgang mit Django, pandas und den anderen Bibliotheken recht viel Zeit in anspruch genommen, zumal sich neue Probleme immer wieder im Verlauf des Projektes zeigten. Ein großes Problem ist nach wie vor die Performanz der Applikation. Wir entschieden bewusst unsere Daten in einer Datenbank zu speichern, damit wir nicht im Vorfeld jeder Berechnung die kompletten Datensätze neu einlesen mussten. Dabei mussten wir lernen, wie man mit den Daten performant umgehen kann. Die Datenbank speichert die Modelle in ganz herkömlicher Art als Tabellen mit Textfeldern, ein Abruf der Daten in die zugehörigen Pythonobjekte nimmt jedoch viel Zeit in Anspruch. So mussten wir möglichst viel Arbeit die Datenbank mit SQL-Anfragen erledigen lassen und die gefilterten Daten möglichst als einfache Datentypen, wie String und Listen verwenden.

Wir unterschätzten ausserdem sehr den Aufwand, der zum Aufbau einer geeigneten Datenbasis nötig ist. Wie bereits beschrieben konnten wir nicht *einfach* Projekt Gutenberg Texte verwenden, sondern mussten andere Corpora verwenden. Dadurch, dass wir mehrere Corpora gleichzeitig verwenden und mangels Fachwissen wenig Kontrolle über die enthaltenen Texte haben, sind unsere Ergebnisse an Qualität sicher

streitbar. Ebendieser Aufwand führte auch dazu, dass wir die spannendste unserer Fragen verwerfen mussten. Eigentlich wollten wir geographisch nachverfolgen wie sich Wörter ausgebreitet haben. Lässt sich also im Verlauf der Zeit eine größer werdene Verbreitung eines Wortes verfolgen? Dazu hätten wir Orte an denen die Autoren sich aufgehalten haben mit Zeitspannen in den Daten haben müssen. Diese Daten für die Autoren von Hand einzufügen wäre sehr Aufwändig gewesen, noch schwerer allerdings sind die Daten zu bekommen, die Namen sind oft nicht eindeutig.

In der Visualisierung unserer Ergebnisse konnten wir aus Zeitgründen bis auf einfache Graphen und Tabellen leider keine weiteren Möglichkeiten implementieren.

4 Vision unseres Projektes

Unsere Ergebnisse zeigen bereits, was mit dem Service möglich ist, allerdings sind unsere Ergebnisse zum Teil wenig aussagekräftig. An den Graphen ist erkennbar, dass in einigen Zeitabschnitten nur wenige Texte vorhanden sind, was unsere Ergebnisse verfälscht. Um dies zu Lösen muss man einen hochwertigen Corpus verwenden, der den gewünschten Zeitraum in stabiler Qualität abbildet.

Mit unserem Webservice wollen wir eine Möglichkeit bieten, mit wenig Wissen über die Techniken und Methodiken TextCorpora computergestützt anhand selbst gewählter Wortlisten zu untersuchen. Dafür bietet sich der Browser als Interface gut an. Der Service lässt sich leicht mit eigenen Algorithmen zur Analyse erweitern und bietet so neben den bestehenden Verfahren die Möglichkeit, ihn nach den eigenen Bedürfnissen zu erweitern. Unser Service lässt sich sowohl lokal, als auch auf einem Webserver bereitstellen, also sowohl zum iterativen Erforschen für sich selbst, als auch als Platform zur Präsentation seiner Ergebnisse. Insbesondere ist auch die Verwendung eigener Corpora für eigene Fragen interessant.

Der Webserver erlaubt nicht nur Betrachtung der berechneten Ergebnisse, es kann ebenso auch Textseiten geben, die die Forschung beschreiben und analysieren. Solche

Möglichkeiten wissenschaftliche Ergebnisse in ansprechender und leicht verfügbarer Form zur Verfügung zu stellen kann helfen die Arbeiten leichter für Laien verständlich zu machen. Die bisherige Praxis Artikel in Fachzeitschriften zu publizieren stellt für Menschen ohne wissenschaftlichen Hintergrund, oder aus anderen Feldern oft eine hohe Hürde dar. Auch um Nachwuchs für spezielle Themengebiete zu begeistern kann der Service hilfreich sein. Momentan muss man seinen zur Berechnung verwendeten Corpus fest einstellen und bekommt darauf Ergebnisse berechnet. Analog zu dem Google NGram Viewer wäre es wünschenswert, wenn man die Daten auch explorativ betrachten kann. Beispielsweise während der Anzeige des Ergebnisses auf einen Zeitabschnitt genauer eingehen, oder Gruppen von Texten zur Berechnung hinzufügen kann. Auch verschiedene Textmengen miteinander vergleichen zu können scheint eine interessante Erweiterung des Funktionsumfangs.

Das Projekt bietet dementsprechend noch viel Raum für Weiterentwicklung, sowohl auf inhaltlicher Seite, wo weitere Algorithmen implementiert werden können, als auch in technischer Hinsicht, moderne Webtechnologien ausnutzend um die Datenbetrachtung zu verbessern und interaktives Untersuchen zu ermöglichen.