Einführung

Das von uns durchgeführte Projekt beschäftigt sich mit den NLP-tools und deren Anwendung. Der Fokus liegt dabei auf der Named Entitity Recognition der SpaCy Features. Es verfolgt das Ziel zu zeigen, dass sich die Kategorien der Named Entities anhand desselben inhaltlichen Textens, jedoch in unterschiedlichen Sprachen (Deutsch, Englisch, Russisch), unterscheiden. Unsere Analyse geht der Frage nach, inwiefern die Zugehörigkeit von Named Entities und Kategorien übereinstimmen. Für die Vorgehensweise haben wir die dazugehörigen Sprachmodelle importiert, als auch die drei Texte in verschiedenen Sprachen. Beim Programmierteil haben wir uns dazu entschieden eine Funktion zu schreiben, welche Named Entities und deren Labels erkennt und diese in einem Dictionary speichert. Alle Komponente des Codes wurden gemeinsam erstellt, ausprobiert und analysiert.

Daten und Ressourcen

Um unserer Frage nachzugehen haben wir uns dazu entschieden einen Textausschnitt des literarischen Buches „Die Reise nach Petuschki“, geschrieben von Wenedikt Jerofejew und erstmals veröffentlicht im Jahre 1973, zu nehmen. Unsere Textausschnitte stammen aus den ersten zwei Kapiteln „Vorbemerkung des Autors“ und „Moskau. Auf dem Weg zum Kursker Bahnhof“. Die Bücher stehen in vielen online Bibliotheken zur Verfügung. Aus Rechtsschutzgründen dürfen wir unseren Buchinhalt nicht veröffentlichen und weitergeben. Die Abschnitte sind in der Hinsicht sehr gut geeignet für unsere Analyse, weil sie besonders viele Eigennamen, Ortsnamen und Städtenamen etc. beinhalten. Selbstverständlich kann man nach Wunsch auch Texte aus anderen Gattungen verwenden.

Für den Programmierteil, in dem wir mit SpaCy gearbeitet haben, muss man vorher die notwendigen Tools installieren. Auf dem Screenshot sieht man die erforderlichen Befehle.

INSTALLATION

```
In [1]: ! pip install -U pip setuptools wheel
In [2]: ! pip install -U spacy
In [3]: ! python -m spacy download en_core_web_sm
In [4]: ! python -m spacy download de_core_news_sm
In [5]: ! python -m spacy download ru_core_news_sm
```

Zeile 1 und 2 installieren SpaCy[1].

Zeile 3, 4, 5 installieren die Sprachmodelle (EN, DE, RU).

Weitere Sprachmodelle findet man auf der SpaCy Webseite[2].

Sobald man SpaCy und die Sprachmodelle installiert hat, kann man auf diese zugreifen und benutzen.

Methode

Wir haben unseren Code im Jupyter Notebook geschrieben und ausgeführt, da es hier einfacher ist, es visuell darzustellen. Unsere Untersuchung kann mithilfe einer Funktion dargestellt werden, die zu Beginn das benötigte Sprachmodell lädt. Weiterhin erkennt sie die Named Entities und speichert sie daraufhin in einem Dictionary zusammen mit ihren Labels. Der genauere Prozess wird in einzelnen Schritten beschrieben:

1)
```
import spacy
import sys
import matplotlib.pyplot as plt
from collections import Counter
```

Zu Beginn importieren wir die benötigten Module. In Zeile 1 importieren wir die SpaCy Bibliothek. Zeile 2 braucht man nur, wenn man die Funktion im VS Code ausführt. Für die Ausführung im Jupyter Notebook kann man diese Zeile auch weglassen. Zeile 3 braucht man für die Erstellung eines Kreisdiagrammes. Zeile 4 importiert einen Counter, welcher die Zahlen für das Kreisdiagramm liefert. Wer kein Diagramm haben möchte, lässt Zeile 3 und 4 weg.

2)
```
def extract_categories(text, language):
```

Die Funktion extract_categories wird definiert. Diese nimmt als Parameter den Path zu der Datei mit dem Textabschnitt und seine Sprache.

3)
```
if language == "russian":
    nlp = spacy.load("ru_core_news_sm")
    file_text = open(text, encoding="utf8").read()
    file_doc = nlp(file_text)
if language == "english":
    nlp = spacy.load("en_core_web_sm")
    file_text = open(text, encoding="utf8").read()
    file_doc = nlp(file_text)
if language == "german":
    nlp = spacy.load('de_core_news_sm')
    file_text = open(text, encoding="utf8").read()
    file_doc = nlp(file_text)
```

Mithilfe von drei If-Bedingungen überprüfen wir, um welche Sprache es sich handelt. Dann wird das entsprechende Sprachmodell geladen. Als nächstes wird die Textdatei importiert und von dem Programm gelesen (encoding verhindert Fehler). Anschließend wird das Sprachmodell an die Textdatei angewandt.

4)

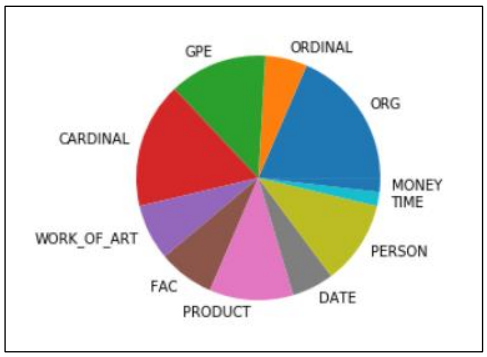
```
categories = {}
for ent in file_doc.ents:
    categories[ent] = ent.label_
```

Ein leeres Dictionary wird erstellt. Mithilfe einer Schleife iterieren wir durch die Entities des Textes. Um die Iteration zu ermöglichen, verwenden wir den eingebauten Befehl von SpaCy (.ents). Danach befüllen wir das leere Dictionary mit den gefundenen Namenswörtern als Keys und deren Kategorien als Values. Auch hier wird die Kategorie durch den eingebauten Befehl von SpaCy (.label_) gefunden.

5)

```
frequency_of_labels = dict(Counter(categories.values()))
plt.pie(frequency_of_labels.values(), labels=frequency_of_labels.keys())
plt.show()
```

Hier wird ein Kreisdiagramm erstellt. Die Anzahl der Wörter, die dieselbe Kategorie haben, wird gezählt. Als Ausgabe bekommt man eine prozentuale Aufteilung der Ergebnisse (siehe unten).



6)

```
return categories
```

Return Befehl gibt das Dictionary zurück.

7)

```
extract_categories(path, sprache)            # so in Jupyter Notebook
extract_categories(sys.argv[0], sys.argv[1])  # so in VS
```

Zeile 1 ist der Funktionsaufruf für Jupyter, welcher aus dem persönlichen Path und der jeweiligen Sprache besteht. Zeile 2 ist der Funktionsaufruf für VS Code, der als Parameter zwei Kommandozeilenargumente nimmt. Je nach dem in welchem Editor man es ausführt, ist nur eine von den Zeilen auszuführen.

Ergebnisse und Diskussion

Nachdem das Programm erfolgreich funktioniert hat, kann man sich nun die Ergebnisse genauer anschauen. Dabei muss man beachten, dass die Sprachmodelle unterschiedlich viele Kategorien mit sich bringen. Für eine farbige und bessere Ansicht der Entities im Text, siehe Anhang[3]. In der untenstehenden Tabelle kann man die Zahlen im direkten Vergleich sehen. Hinweis: Die Zahlen wurden von uns selbstständig manuell ermittelt.

	Englisch	Deutsch	Russisch
Wörteranzahl in der Textdatei	1090	1048	868
Anzahl der vorhergesagten der Named Entities	54	59	32
Anzahl der Kategorien	18	4	3
Korrekte Vorhersage	50	38	30
Inkorrekte Vorhersage	4	21	2
Korrekt vorhergesagte Labels	27	28	26
Inkorrekt vorhergesagte Labels	23	10	4

Auffällig im Russischen ist, dass es nur drei Kategorien (LOC, PER, ORG) gibt. Das ist auch der Grund weshalb in anderen Sprachen, wie im Englischen, viel mehr Named Entities erkannt werden, z.B. Datum, Zeit, Zahlen, etc. Alle vier inkorrekt vorhergesagten Labels sind literarische Titel und können darauf zurückgeführt werden, dass es im Russischen kein extra Label für Kunstwerke gibt. Deswegen wurden den Überschriften in unserem Text falsche Kategorien zugeordnet. Im Englischen dagegen, wo es das Label ‘WORK_OF_ART’ gibt, wurden die Vorhersagen richtig angegeben.

Weiter geht es mit den Auffälligkeiten im deutschen Textabschnitt. Interessant ist hier die Häufigkeit der Interjektionen (‘oh‘, ‘quatsch‘), welche von SpaCy auch als Named Entities eingeordnet worden sind, wobei es in Wirklichkeit keine sind. Hier handelt es sich um die Labels PER und MISC. Die Gemeinsamkeit solcher Fälle ist die Satzstruktur: Ausrufworte stehen zu Beginn des Satzes, die von einem Komma gefolgt werden („Oh, unheilvolle Illusion.“). Der Grund für die falsche Zuordnung könnte sein, dass solche Sätze sehr oft mit einem Namen als Anrede anfangen. Somit könnte es sein, dass das Programm das „oh“ mit einer Anrede verwechselt. Das würde die Zuweisung des Wortes zu der Kategorie PER erklären.

Im Englischen sind die häufigsten Label Fehler an Orts-/und Eigennamen gebunden. Eine mögliche Erklärung für die falsche Einordnung der Ortsnamen, könnte zu einem daran liegen, dass ein Artikel davorsteht. Dadurch wirkt es mehr wie eine Organisation und weniger wie ein Ort, weil Orte (Städte, Stadtteile) normalerweise ohne Artikel geschrieben werden. Zum anderen jedoch könnte die fehlerhafte Erkennung, z.B. des Kremls als ORG, darauf

zurückgeführt werden, dass man in der jeweiligen Kultur eine unterschiedliche Definition des Kremls hat. Eventuell könnte es sein, dass im Englischen der Kreml als ein Komplex wahrgenommen wird und nicht als ein Stadtteil, wie in der russischen Denkweise.

Herausforderungen und offene Fragen

Eine der Fragen, die uns nach den erhaltenden Ergebnissen beschäftigt, lautet: Wären die Vorhersagen korrekter gewesen, wenn die drei Sprachemodelle genau die gleichen Kategorien enthalten würden? Vermutlich wäre die Anzahl der erkannten Named Entities in allen drei Texten ähnlich, jedoch könnte es aber sein, dass die Labels immer noch nicht so präzise vorhergesagt wären. So wie wir gesehen haben, wurde in einigen Sprachen den gleichen Wörtern (z.B. Eigennahmen) unterschiedliche bzw. falsche Labels zugeordnet, obwohl die richtige Kategorie der Wörter in den Sprachmodellen vorhanden ist (z.B. PER für Personennamen).

Eine andere Überlegung wäre, ob die Named Entity Recognition an allen Textarten gleich funktioniert. Eine Untersuchung eines wissenschaftlichen Artikels oder einer politischen Rede könnte vielleicht ganz andere Ergebnisse liefern. Durch die Wörter in unserem Textabschnitt, die etwas veraltet oder als sehr spezifisch nur für die russische Sprache gelten, wurde möglicherweise die Fähigkeit von SpaCy beeinträchtigt und schlussfolgend zu einem falschen Ergebnis geführt.

Die Idee, weitere Texte in Form von politischen Reden zu untersuchen, hatten wir zu Beginn des Projekts, jedoch ist dieses Thema aus Zeitgründen nicht umsetzbar gewesen. Die Textquellen, die wir uns bereits ausgesucht haben, verlinken wir im Anhang[4]. Auf der Webseite findet man viele parallele Texte. Wir hatten vor uns mit dem Korpus Deutsch-Englisch zu beschäftigen. Selbstverständlich könnte man es mit jedem Korpus ausprobieren, dafür müsste man jedoch die Funktion und die Sprachmodelle anpassen.

Zusammenfassung

Rückblickend auf unsere Ausgangslage war unser Ziel, zu zeigen, dass sich die Kategorien der Named Entities, anhand desselben inhaltlichen Textens, jedoch in unterschiedlichen Sprachen (Deutsch, Englisch, Russisch), unterscheiden. Unsere Leitfrage dazu lautete: Inwiefern stimmt die Zugehörigkeit von Named Entities und der Kategorien überein.

Anhand unserer Analyse ist es uns durchaus gelungen zu beweisen, dass sich die Kategorien in den jeweiligen unterschiedlichen Sprachen unterscheiden, obwohl die Texte inhaltlich gleich sind. Grund hierfür ist einerseits, dass es in den Sprachen unterschiedlich

viele Labels gibt. Dadurch entsteht automatisch eine unterschiedliche Zuordnung zwischen Wort und Label. Andererseits ist die Wortbedeutung in den Sprachen auch nicht ganz identisch, weshalb auch hier die Fehlbenennung der Kategorie vorprogrammiert ist. Diese zwei Gründe scheinen uns nachvollziehbar zu sein, jedoch sind wir der Meinung, dass die Sprachmodelle weiter ausgebaut werden sollten. Man sollte versuchen die Anzahl der Kategorien in allen Sprachen gleich zu machen, das würde automatischen zu besseren Ergebnissen führen. Die Tatsache, dass einige Worte in unterschiedlichen Sprachen anders interpretiert werden, kann man wahrscheinlich nur sehr schwierig vermeiden. Bezüglich dieser Fehlerquote sollte man sich von Beginn an bewusst sein.

Rückblickend würden wir unser Vorhaben für den Anfang als nützlich bewerten, aber bei weitem nicht umfangreich genug um unsere Forschungsfrage angemessen zu beantworten. Für eine ausführlichere Antwort braucht man deutlich mehr Texte aus unterschiedlichen Genres, um einen größeren Vergleich zu bekommen. Dadurch kann man die Ergebnisse besser gegenüberstellen und analysieren. Für die Zukunft würden wir uns wünschen, dass an der Forschungsfrage weitergearbeitet wird.

Anhang und Quellenangaben:

1) https://spacy.io/
2) https://spacy.io/models
3) https://spacy.io/usage/visualizers#ent
4) https://www.statmt.org/europarl/