Universität Trier

Semester: Sommersemester 2021

Modul: Computerlinguistische Programmierung

Dozierende: Herr Dr. Sven Naumann

Benutzerhandbuch

Topic Modeling mit deutschen Tweets

Julia Karst, Tobias Niedballa, Sarah Ondraszek, Julia Weyer

Inhaltsverzeichnis

1 Einleitung

Dieses Benutzerhandbuch soll als Hilfestellung dienen, um LDA Topic Modeling auf deutschsprachigen Daten durchzuführen. Das Projekt und sämtliche Skripte finden sich auf GitHub unter `https://github.com/sarahondraszek/comp_ling_LDA`.

2 Anforderungen

2.1 Daten

Das Modell wurde für die Interpretation deutscher Tweets entwickelt. Um ein bestmögliches Ergebnis zu erhalten, sollten die Daten daher in der deutschen Sprache vorliegen. Zudem müssen die einzelnen Datein im csv-Format gespeichert sein.

2.2 Installation

Um eine optimale Nutzung zu ermöglichen, sollten die folgenden Programme und Bibliotheken installiert werden.

- Python 3.9
- gensim 4.0.1
- nltk 3.6.2
- spacy 3.0.6 (und de_core_news_md)
- tkmacosx 1.0.3
- pandas 1.2.4
- pickle 0.7.5
- pyLDAvis 2.1.2

Bei pyLDAvis ist zu beachten, dass unter der neuesten Version einige Probleme auftreten, so bietet sich die Nutzung einer älteren Version an. Damit die GUI auch unter MacOS richtig angezeigt wird, muss zudem auch tkmacosx installiert werden.

3 Nutzung der Programme

Zur einfachen Nutzung bietet sich die Verwendung der GUI an, welche in Abschnitt 4 genauer beschrieben ist. Wer auf die Nutzung der GUI verzichten oder Einfluss auf sämtliche Parameter des LDA-Models haben möchte, hat die Möglichkeit die Python-Programme im 'script'-Ordner zu nutzen.

Hierbei müssen - in dieser Reihenfolge - die Programme *run_preprocessing.py*, *run_tm.py* und *visualization_alternative.py* ausgeführt werden. Letzteres nutzt die pyLDAvis Version 2.1.2. Wer die neueste Version benutzen möchte, kann stattdessen *visualization.py* benutzen, woraufhin die Ergebnisse als HTML gespeichert werden. Die Filter-Optionen in *run_preprocessing.py* `dictionary.filter_extremes(no_below, no_above)` und Modellparameter in *run_tm.py* können nach den eigenen Bedürfnissen angepasst werden. Im Ordner 'data' muss zudem der Ordner 'corpus' durch die eigenen Daten ersetzt werden.

4 GUI

Insgesamt besteht die GUI aus sechs verschiedenen Seiten, die auf dem Ablauf des Topic Modeling basieren: Startseite, Korpusauswahl, Preprocessing, Parameterauswahl, Visualisierung und Endbildschirm. Durch Starten des Python-Programmes *gui.py* wird die GUI geöffnet.

Zunächst erscheint ein Startbildschirm, auf dem einige grundlegende Informationen zur weiteren Nutzung beschrieben sind. Anschließend gelangt man auf die Seite 'Select Corpus'. Klickt man auf den entsprechenden Button 'Select Files', lassen sich mehrere csv-Dateien auswählen, aus denen sich der Korpus zusammensetzt. Wurden alle Dateien ausgewählt, gelangt man durch Anklicken des Buttons 'Done' auf die nächste Seite. Diese befasst sich mit dem Preprocessing. Klicken des entsprechenden Buttons startet den Prozess. Abhängig von der Größe des Korpus kann dies eine Zeit lang (ca. 10 Minuten oder länger) dauern. Das Ergebnis wird anschließend automatisch zwischengespeichert (in der Datei 'docs' im Ordner 'data'), sodass beim nächsten Mal die Schritte 'Select Corpus' und 'Preprocessing' übersprungen werden können und somit Zeit gespart wird. (Wichtig: Solange die 'docs' Datei nicht zuerst gelöscht wird, wird immer wieder auf diese zugegriffen, insbesondere auch dann, wenn in 'Select Corpus' andere Files ausgewählt wurden.)

Im Folgenden gelangt man zur Seite 'Parameters', welche es ermöglicht Modellparameter einzugeben. Zur Auswahl stehen die Anzahl der Themen, die Filter 'no_below' und 'no_above', sowie die Angaben 'Chunk Size', 'Passes' und 'Iterations' für das LDA-Modell. Hierbei müssen sämtliche Felder ausgefüllt werden. Die meisten Angaben erfordern die Eingabe eines positiven ganzzahligen Wertes. Die Ausnahme bildet der Parameter 'no_above', für den eine Prozentangabe einzugeben ist. Zur Orientierung sind bereits einige Default-Werte vorgegeben, welche entsprechend an das verwendeten Korpus angepasst werden sollten. Klickt man daraufhin den Button 'Run the model', so werden die Parameter überprüft (falsche Eingaben führen zu einer detaillierten Fehlermeldung) und die Ergebnisse in der Konsole ausgegeben. Die nächste Seite ermöglicht es, die Topics zu visualisieren. Klicken von 'Show Topics' öffnet dazu unter Verwendung von pyLDAvis ein entsprechendes Fenster im Browser. Hinterher muss der Server wieder gestoppt werden. Sollten in diesem Schritt Probleme auftreten, liegt dies möglicherweise an der verwendeten Version von pyLDAvis.

Auf dem Endbildschirm hat man schließlich die Möglichkeit, das Programm zu schließen, zurück zur Parameterauswahl zu gehen oder einen neuen Korpus zu nutzen. Ist letzteres der Fall, so muss vorher der Button 'Delete saved data' gedrückt werden, welcher die oben erwähnte 'docs' Datei löscht.