# Contents

# 1 Computer-Assisted Text Analysis for Comparative Politics

# 2 Introduction

- Fokus auf Tools für Komparatisten, um *textual* data zu nutzen
- Hervorhebung des *unsupervised topic modeling*
- Verwendung des Structural Topic Model um das Potential von Topic Modeling für vergleichende Politik aufzuzeigen
  - STM erlaubt Rückschlüsse auf Beziehung zw Metadaten und Textkorpus
- wie unterscheidet sich Textanalyse und Text Processing in versch Sprachen
  - R package 'translateR'

# 3 Text and Language Basics

## 3.1 Research Questions and Data Analysis

- automatische Inhaltsanalyse und vergleichende Politik sind eine gute Kombination
- Länder produzieren Texte in noch nie dagewesenem Umfang
- traditionelle Regierungsstatistiken sind häufig nicht vorhanden, unvollständig, manipuliert oder falsch gemessen
  - Regierungen produzieren allerdings große Mengen an Textdaten, welche für deskriptive und kausale Inferenzen genutzt werden können
  - Anreiz für Gelehrte andere Formen von Data zu verwenden
- Gelehrte der vergleichenden Regierungslehre/Politik verwenden bereits automatische Methoden für Textanalysen
  - weitverbreiteste Form von Text zu Politiker sind wahrscheinlich Aufzeichnungen von Reden oder anderen Statements
- Auflistung einiger interessanten Studien die automatische Textanalyse ultilisiert haben

## 3.2 Text Processing Basics: A Multilanguage View

- Analytiker muss zuerst sicherstellen das zu analysierender Text maschinell lesbar ist
  - statistische Methoden der Textanalyse sind meist unabhängig von der Sprache
    - aber Tools des Prepocessings *nicht*

3 Herausforderungen bei der Arbeit mit verschiedenen Sprachen:

1. Umgang mit Zeichenkodierung (dealing with encodings)
2. Präprozessing zur Reduktion der Dimensionalität
3. Umgang mit großen Korpora

## 3.3 Umgang mit Encodings

- Sprachen können unterschiedliche Zeichenkodierung haben und unterschiedliche Computer händeln dies auf underschiedliche Art & Weise
  - unterschiedliche default encodings
- wenn der Analyst Daten aus versch Quellen bezieht ist es von nöten, dass das Encoding angepasst wird, sodass es in allen Dokumenten gleich ist
  - anschließend muss sichergestellt werden, dass die Software die Zeichenkodierung korrekt liest

## 3.4 Prepocessing to extract the most information

### 3.4.1 Stopword removal

- Entfernung von Worten die extrem häufig auftreten aber nicht relevant im Bezug auf das Erkenntnisinteresse sind (zb "and", "the", "und", "zum")
  - viele Sprachen haben eine Liste üblicher stop words
- eine Liste von stop words die entfernt werden, sollte sorgfältig gewählt werden, da unterschiedliche stop words zu unterschiedlichen Ergebnissen führen können und manchmal im Kontext entscheidend sein können

### 3.4.2 Stemming & lemmatization

Stemming:

- Entfernung der Enden von konjugierten Verben oder Nomen in der Pluralform, so dass nur der "Stamm" überbleibt
- nützlich in jeder Sprache in der das Ende von Worten geändert wird für eine Veränderung der Zeit oder Anzahl (Englisch, Spanisch, Französisch etc)
- nicht in jeder Sprache nötig/nützlich
  - chinesische Verben werden zB nicht konjugiert und Nomen in chinesisch werden nicht durch eine Endung pluralisiert
- Nützlichkeit ist anwendungs- und sprachabhängig
- Stemming ist ein Vefahren/Näherung an ein allgemeineres Ziel was als Lemmatization (Lemmatisierung) bezeichnet wird

Lemmatization:

- Identifikation der Grundform eines Wortes und Gruppierung dieser Worte
- komplexer Algorithmus, der nicht einfach das Ende eines Wortes abschneidet, sondern die Herkunft des Wortes identifiziert und nur das Lemma (Grundform) des Wortes zurück gibt
- kann außerdem Kontext schlussfolgern:
  - zB "saw" als Nomen = "Säge" bleibt so, als Verb = "sah" wird zu $\rightarrow$ "sehen/see"
- für Englisch funktioniert Stemming fast so gut wie Lemmatization in anderen Sprachen wie zB Koreanisch oder Türkisch ist Lemmatization hilfreicher

### 3.4.3 Compound words

- einige (compound) Sprachen setzen oft Worte zusammen (compounding) um ein neues Wort zu bilden zB Kirche + Rat = Kirchenrat
  - decompounding macht in diesem Fall keinen Sinn da die Worte zusammengehören

- in decompoung Sprachen widerrum können *mehrere* getrennte Worte zu *einem* Konzept gehören:
  - "social security" und "national security"
    - beide enthalten "security" aber haben trotzdem unterschiedliche Bedeutung, daher möchte der Analyst die Worte evtl. compounden (zusammenführen), zu "nationalsecurity" und "socialsecurity", um die Bedeutung an *ein* Wort zu koppeln

#### 3.4.4 Segmentation

- einige Sprachen wie zB Chinesisch werden nicht durch Leerzeichen segmentiert und erfordern deshalb automatische Segmentierung bevor sie von einem Statistikprogramm weiterverarbeitet werden können

### 3.5 Building the document-term matrix

- nach dem das Prepocessing abgeschlossen ist, werden die übrig gebliebenen Worte genutzt, um eine document-term matrix (DTM) zu konstruieren
- in einer document-term matrix repräsentiert jede Reihe ein Dokument und jede Spalte ein ein einzigartiges Wort
  - jede Zelle enthält die Anzahl des Auftreten des jeweiligen Wortes (Spalte) im jeweiligen Dokument (Reihe)
    - üblicherweise enthalten viele Zellen eine 0

Beispiel:

| Berlin | Brüssel | Merkel | Schulz |
|---|---|---|---|
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |

- Reihenfolge der Worte beachtet die DTM nicht
- da diese DTM schon bei Korpora moderater Größe sehr groß werden kann, ist es ratsam nur Einträge zu speichern die nicht 0 sind (sparse representation)
- die DTM oder ihre sparse representation sind der primäre Input für automatische Textanalysemethoden

### 3.6 Multilanguage Preprocessing Tools

#### 3.6.1 Language-specific processing

- das R Package ‘tm‘ kann Stemming für 11 und stop words removal für 13 Sprachen durchführen
- die Python basierde Applikaton ‘txtorg‘ unterstützt 32 Sprachen
  - alle Sprachen durchlaufen Schritte des best-practice preprocessing
    - geeignete Kombination von Stemming, Segmentation und stopword Entfernung

#### 3.6.2 Translation

- Übersetzung der Dokumente in 1 Sprache kann sich als effizient und zugänglicher für den Nutzer erweisen
- cross-lingual comparison wird so gut wie nirgends unterstützt
- Empfehlung: R package ‘translateR‘ gibt Zugang zu sehr ausgereiften Übersetzungssystemen (jene die produziert sind von Google und Microsoft)

## 4 Computer-Assisted Text Analysis

### 4.1 A Brief Overview of Approaches

zwei Herangehensweisen im Hinblick auf automatische Textanalyse:

1. supervised methods:
   - "specify what is conceptually interesting about documents in advance, and then the model seeks to extend our insights to a larger population of unseen documents"
   - bspw. manuelle Unterteilung/Klassifizierung von 100 Dokumenten in 2 Kategorien → und dann automatische Klassifizierung der verbleibenden 9900
2. unsupervised methods:
   - zB topic modeling
   - keine manuelle Klassifizierung der Texte im Voraus

- das Modell wird benutzt um eine "low-dimensional summary" zu finden, welche die Dokumente am besten erklärt im Hinblick auf zuvor formulierte Annahmen

bei supervised methods liegt der Großteil der von Hand zu verrichtenden Arbeit in der Konstruktion des training sets, bei unsupervised in der Interpretation der model results

Autoren benutzen ein bestimmten Typ des unsupervised topic modeling, welches auf dem Latent Dirichlet Allocation (LDA) model basiert

- LDA = mixed-mempership model
  - d.h jedes Dokument wird repräsentiert als eine Kombination/Mischung aus einem Pool von Themen und jedem Wort, welchem ebenfalls ein bestimmtes Thema zugewiesen wird
- each topic is a distribution over the words in the vocabulary
  - dies wird gelernt und nicht vom model angenommen

## 4.2 Multilangual Text Modeling

Herangehensweisen:

1. Analyse in der nativen Sprache des Textes
   - machbar bei supervised Vorgehen (trotzdem aufwändig), aber keine klare Methodik für unsupervised Vorgehen
2. Übersetzung der Texte in eine common Sprache
   - manuelle Übersetzung extrem teuer, daher maschinelle Übersetzung
3. Explicit multilangual representation
   - develop a model which maintains an explicitly multilingual representation
   - Vereinheitlichung der konzeptuellen Models über die versch Sprachen hinweg, sodass scaling und Themen (topics) in der einen Sprache vergleichbar mit den Repräsentationen der anderen Sprache sind

Übereinstimmung(Korrespondenz) mehrsprachiger Topics hängt von *particular alignment informations* des Anwenders ab und muss (manuell) validiert werden

- für jedes Topic muss überprüft werden, dass topic word distributions über die Sprachen hinweg einheitlich sind

### 4.3 The STM

- das STM Framework ist ein mixded-membership topic model (so wie LDA) mit dem Zusatz, dass es dokumentabhängige Metadaten berücksichtigen und mit einbeziehen kann
  - kann Qualität der erlernten Topics erhöhen und Hypothesentests erleichtern
  - erhältlich über das R package ‘stm‘

#### 4.3.1 Rolle von Covariates (unabhängigen Variablen)

- STM kann im Gegensatz zu LDA "include document-level covariates in the model as a method for pooling information"

#### 4.3.2 Topic correlations

- STM kann darüberhinaus die Korrelation von Themen explizit kalkulieren

## 5 Applications

Vorstellung von 2 Anwendungsbereichen des STM

1. Analyse von islam. *fatwas* in nativer Sprache
2. Analyse von Texten in 2 Sprachen (Arabisch & Chinsesisch)

### 5.1 Jihadi Fatwas

Kombination von Daten über musl. Gelehrte und Kodierungen im Bezug darauf ob Gelehrte Jihadisten sind, um herauszufinden wie der Themensinhalt von Texten die von jihadist. Gelehrten sich von Nichtjihadisten unterscheiden:

1. Data von Nielsen: Daten über das Leben und Schriften von 101 bekannten jihad. und nicht-jihad. musl. Gelehrten inklusive 27.248 Texte von diesen
   - Großteil dieser Texte sind fatwas

- aber auch Bücher, Artikel und Schriften
- diese Texte zeigen wie Gelehrte mit religiösen Konstitutionen(Wählerschichten?) interagieren

2. Unabhängige Kodierungen ob diese Gelehrten Jihadistens sind
3. 2 Quellen:
   - *Militant Ideology Atlas* = führt 56 Individuen an die regelmäßig/oft von Jihadisten zitiert werden
   - Auflistung von bekannten Gelehrten in 8 ideologischen Kategorieren (Jarret Brachman):
     - Salafist, Madkhali Salafist, Albani Salafist, scientific Salafist, Salafist Ikhwan, Sururis, **Qutubis** und **Global Jidahist**
       * die letzten 2 Kategorien werden als Jihadisten eingeordnet, die anderen nicht
   - zusammen ergeben diese beiden Quellen Assesments von 33 der Gelehrten (20 Jihadisten, 13 Nichtjihadisten) zu denen Nielsen 11.045 Texte gesammelt hat

detailliertes Vorgehen und Ergebnisse auf Seite 11 - 15

### 5.2 Reaktionen auf Snowden in China und in Nahost

- illustratives Beispiel das zeigt wie maschinelle Übersetzung genutzt zsm mit STM werden kann um Vergleiche über Länder und Sprachen hinweg anzustellen
- es sei wichtig zu verstehen wie andere Länder die USA finden/bewerten
  - eine Möglichkeit um dies rauszufinden ist mit Hilfe eines Vergleichs von Responses zu bestimmten Ereignissen
- Sammlung von tausenden Social Media Posts in Arabisch & Chinesisch im Juni 2013, dem Monat als Edward Snowden die Whistle geblowt hat
  - Fokus auf Antworten aus China und Nahost (wichtige strateg. Regionen für die USA)

### 5.2.1 Two approaches to machine translation

- idealerweise Analyse der beiden Datensätze im selben topic model
  - ohne zu übersetzen würde es allerdings keine (vokabularen) Überschneidungen geben
    - dann hätte jeder Korpus eigene indivuelle Topics weil "Snowden" auf Arabisch /neq "Snowden" auf Chinesisch (aber beide selbes Topic "Snowden") und Inhaltsvergleich der Themen somit nicht realisierbar

daher Übersetzung der gesamten Korpora, sowie Übersetzung von Begriffen der DTM mit Hilfe von Google Translate durch ‘translateR‘

- ersteres weil beim gesamten Text der Kontext fuer die Übersetzung erhalten bleibt
- zweiteres weil es nur die minimal benötigten Worte (am meisten relevant) übersetzt
  - individuelle Erstellung der DTM für jede Sprache → Übersetzung der jeweiligen Terms → Merge der übersetzten DTM's
- Sicherstellen das kein Topic exklusiv im Kontext einer Sprache existiert

### 5.2.2 Correcting for systematic differences between languages

- words that mean the same thing in the Chinese and Arabic corpus could sometimes map onto different words in English that are synonyms of each other

→ "Within the STM, we can use a content covariate to capture variations in word use attributable to observed covariates. Here we include the document's original language as a content covariate in order to capture linguistic differences in describing a topic. This allows us to effectively marginalize over differences in word rate use that arise due to linguistic differences or errors in translation"

### 5.2.3 Results

detailliertes Vorgehen & Ergebnisse auf Seite 17 - 21

# 6 Conclusion

Seite 21 - 22 nicht so relevant da nur eine Zusammenfassung des bereits aufgeführten Vorgehens