Wahlprogrammanalyse

Aktuelle Data Science Entwicklungen I

Jan Mühlnikel, Niklas Scholz, Christian Schmid, Luca Mohr

Ziele des Projektes

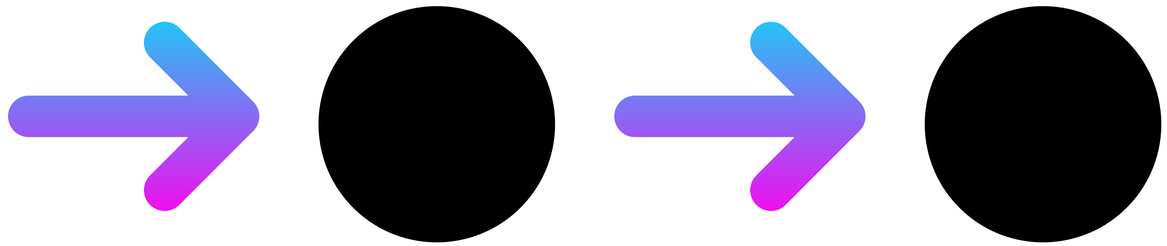
Wahlprogrammanalyse

- Wahlprogramme analysieren
- Fundierte Aussagen anhand dieser Wahlprogramme treffen

Aufbau

Wahlprogrammanalyse

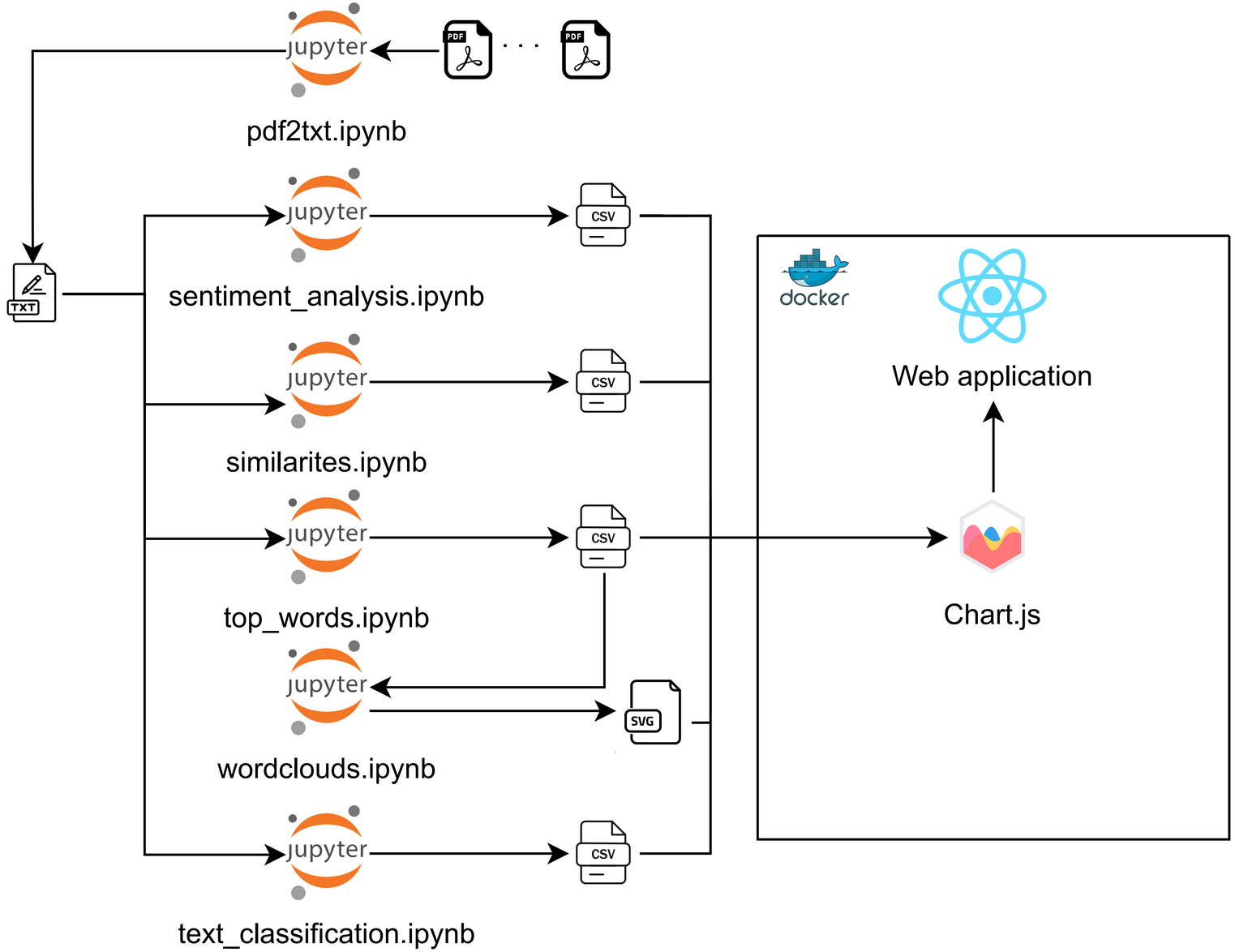
- Datenvorverarbeitung
- Sentimentanalyse
- Ähnlichkeitsanalyse
- Top Wörter
- Wortwolken
- Themenmodellierung



Architektur

Wahlprogrammanalyse

- Python
- ReactJS i.V.m Chart.js
- Docker



Datenvorverarbeitung

Wahlprogrammanalyse

- Textextraktion aus PDFs mittels pypdf
- Textbereinigung mittels Regex (Seitenzahlen, Parteinamen, Kapitelnamen, Seitenumbrüche, etc.)
- Individualität über alle PDFs hinweg
- Individualität innerhalb der einzelnen PDFs

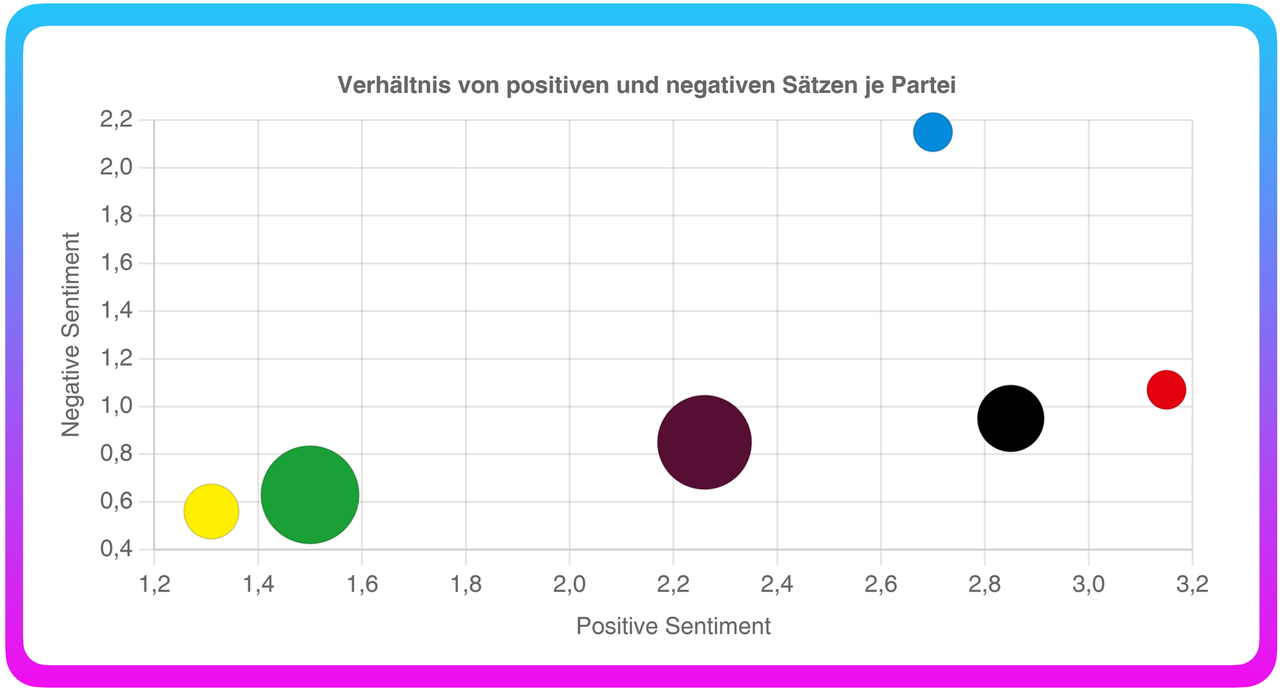
```python
from pypdf import PdfReader
import re

reader = PdfReader("example.pdf")
page = reader.pages[0]
text = reader.extract_text(page)
text = re.sub(r'\s+', ' ', text)
```

Sentimentanalyse

Wahlprogrammanalyse

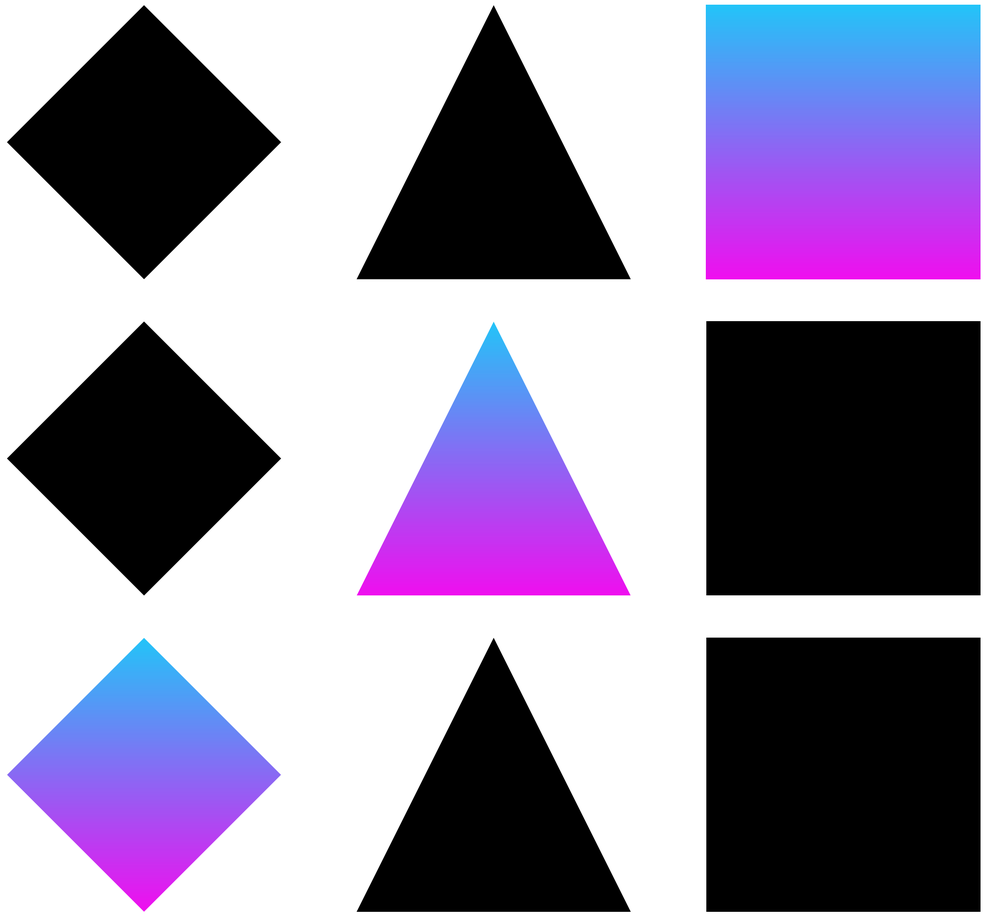
- Satzbasiertes Level der Analyse
- Positiv, negativ, neutral
- „Germansentiment“ Modell
  - Google BERT Transformer Modell
  - Auf 1,8 Mio. Trainingsdaten trainiert
  - Speziell für deutsche Sprache



Ähnlichkeitsanalyse

Wahlprogrammanalyse

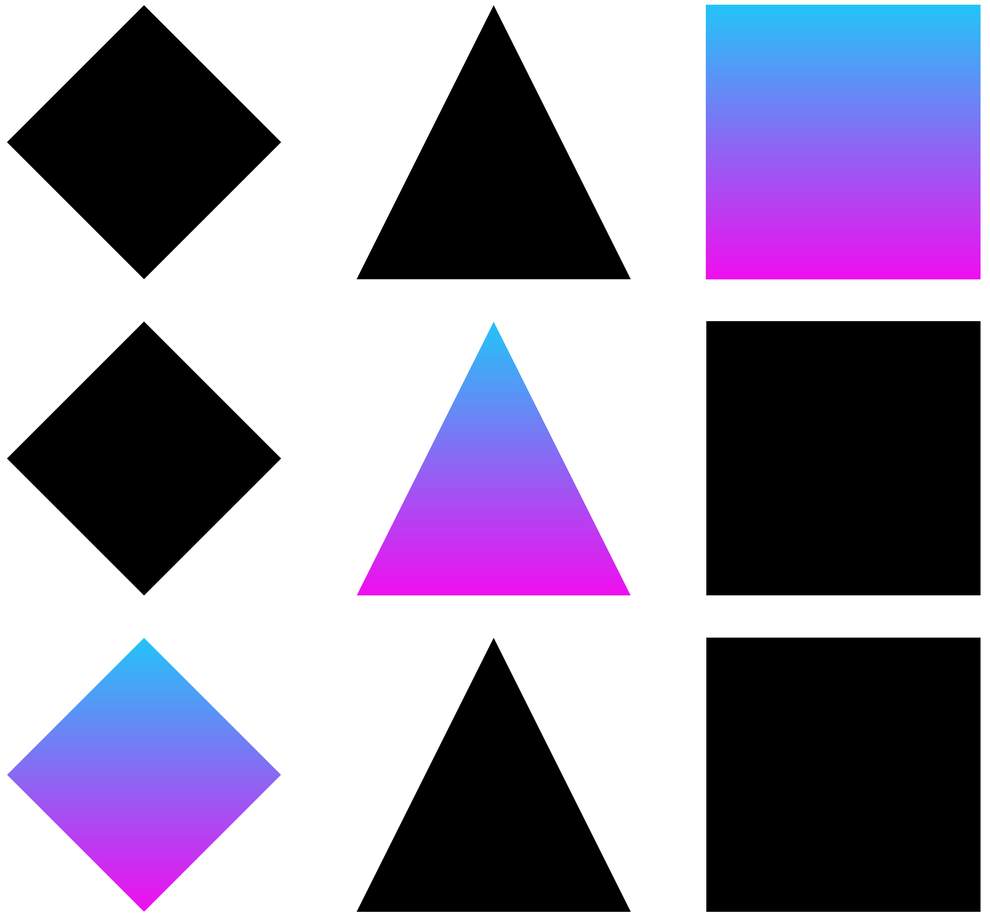
- Vorverarbeitung (Spacy ‚de_core_news_sm' Model)
  - Tokenize
  - Remove tags („/n")
  - Lemmatisierung
  - Stopwörter entfernen



Ähnlichkeitsanalyse

Wahlprogrammanalyse

- Vektorrepräsentation
  - Bag of Words (BoW)
  - TF-IDF
  - Doc2Vec
- Similarity
  - Cosine similarity
- Wahl des Ergebnisses welche Erwartungen am besten reflektiert hat



Herausforderungen

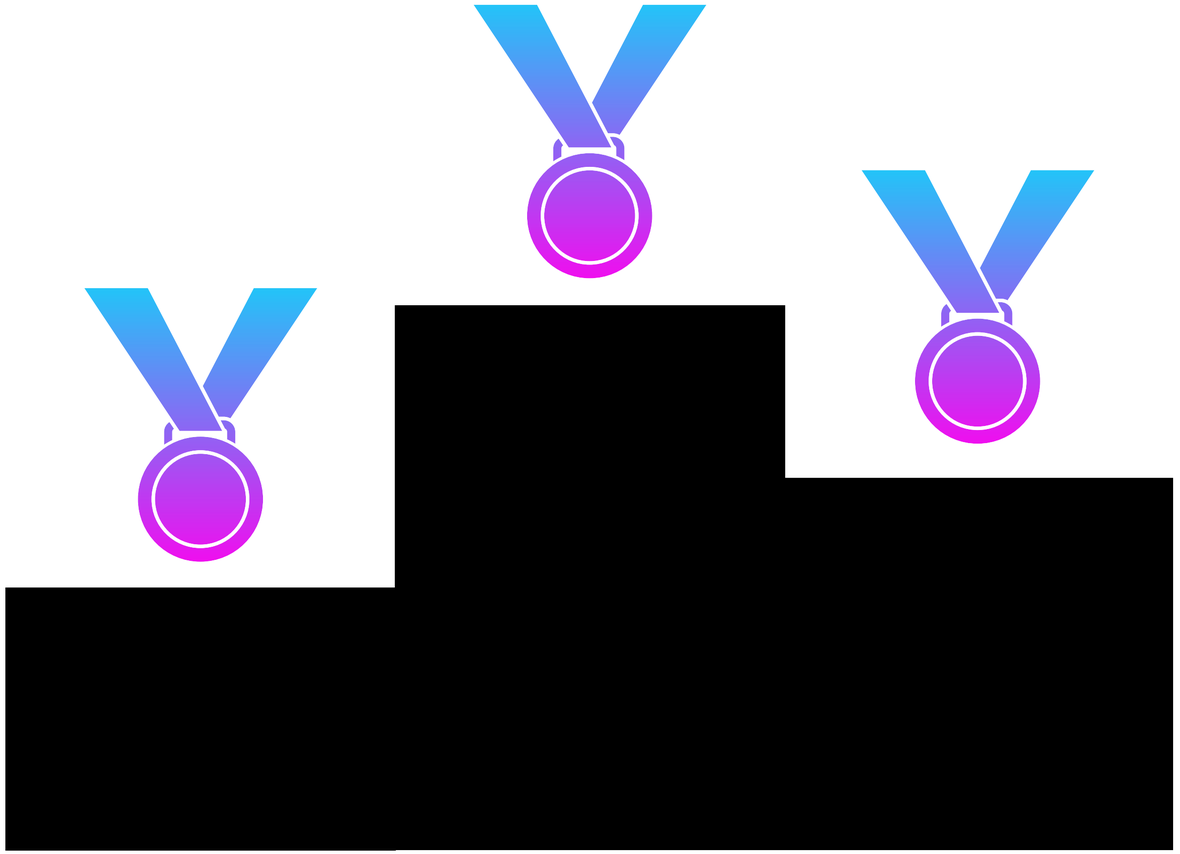
Ähnlichkeitsanalyse

- Nur neutrale Sentiments bei Analyse auf Dokumentenebene
- Entfernung von Seitenzahlen & Inhaltsverzeichnisse

Top Wörter

Wahlprogrammanalyse

- Top 30 Wörter jedes Wahlprogramms
- Entfernung der stopwords mit nltk



Wortwolken

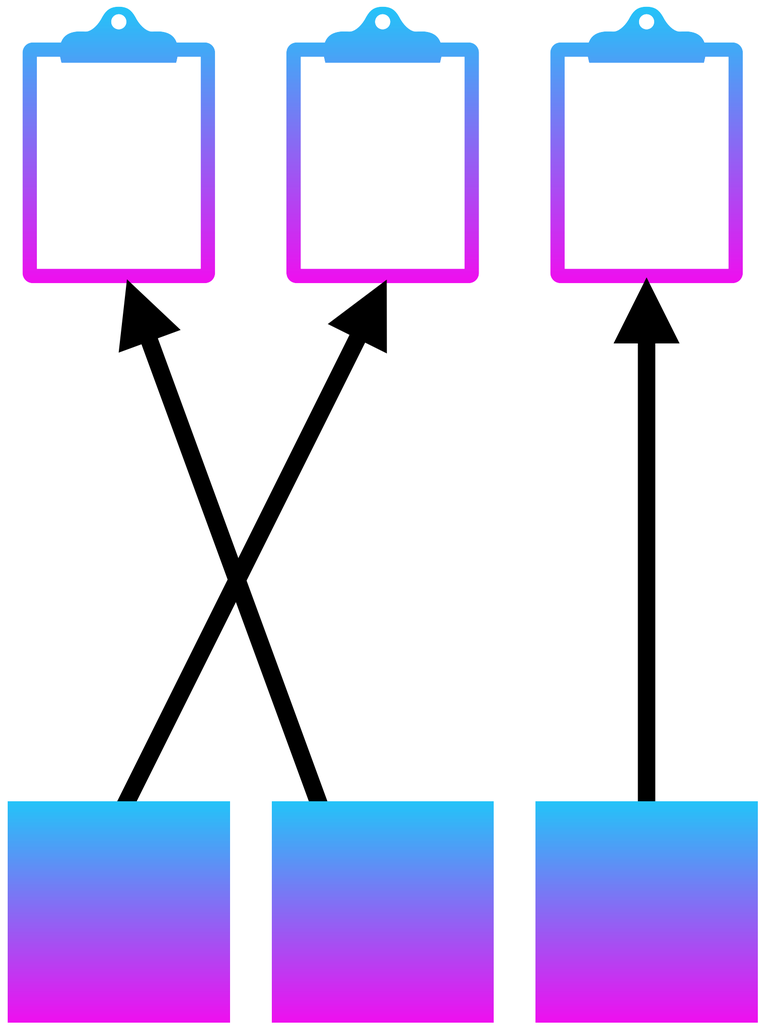
Wahlprogrammanalyse

- Nutzt die Top 30 Wörter
- Bildet eine WordCloud
- Export als .svg

Themenmodellierung

Wahlprogrammanalyse

- Relativer Anteil der Themen
- Deutsches spaCy Modell (Large)
- Themen:
  - Wirtschaft
  - Gesundheit
  - Umwelt
  - Bildung



Herausforderungen

Themenmodellierung

- Richtiges Vorgehen finden für Topic Modeling
  - LDA vs spaCy
- Overfitting beim Topic Modeling
  - Kreuzvalidierung
- Herausforderungen beim Trainieren des Modells

Visualisierung

Wahlprogrammanalyse

- Frontend framework: React.js Hooks / Tailwind css
- Visualisierung: Chart.js

Live Demo

Vielen Dank für eure Aufmerksamkeit

https://github.com/nklsdhbw/election-manifestos-analysis