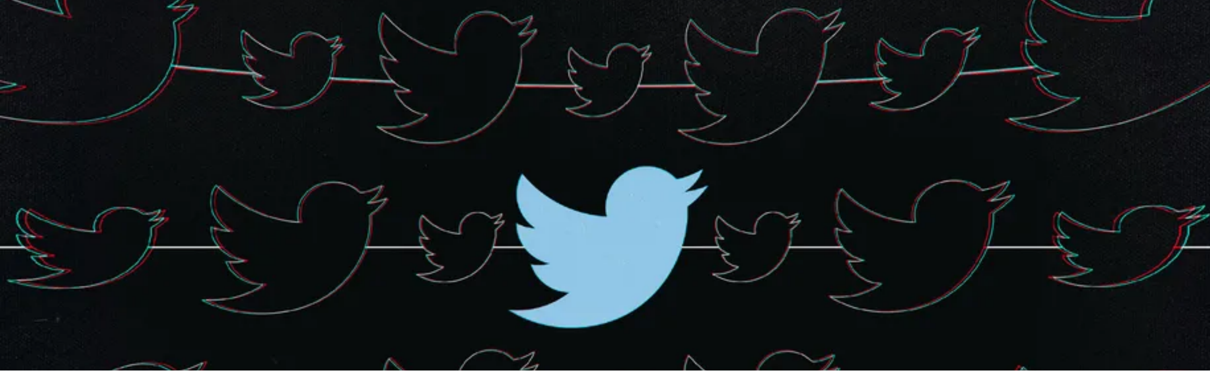


Topic-specific sentiment analysis for tweets by German MPs

Statistical consulting

Asmik Nalmpatian & Lisa Wimmer | July 12th, 2021

Project partner: Prof. Dr. Paul Thurner, Department of Political Science, LMU
Supervisors: Matthias Aßenmacher, Prof. Dr. Christian Heumann

OUTLINE

1 INTRODUCTION & PROJECT OUTLINE

1 INTRODUCTION

- Social media: constant stream of publicly available **text data**
- **Twitter** established as a medium for political discourse and constant source of information
- Frequently resurfacing **research questions:**
  - Which **topics** are being addressed?
  - What kind of **sentiment** is expressed about these topics?

1 PROJECT OUTLINE

- **Primary goal:** analysis of public sentiment in a topic-aware manner for posts scraped from Twitter by German Members of Parliament (MPs)

  → Explore how **topic-specific sentiment analysis** can be implemented with (1) standard ML techniques and (2) more complex DL models.

- **Secondary goal:** make analysis of social media texts in a political context more easily accessible to researchers

  → Provide teaching material on both approaches, composed as a coherent online course

2 GENERAL THEORETICAL CONTEXT

2 GENERAL THEORETICAL CONTEXT

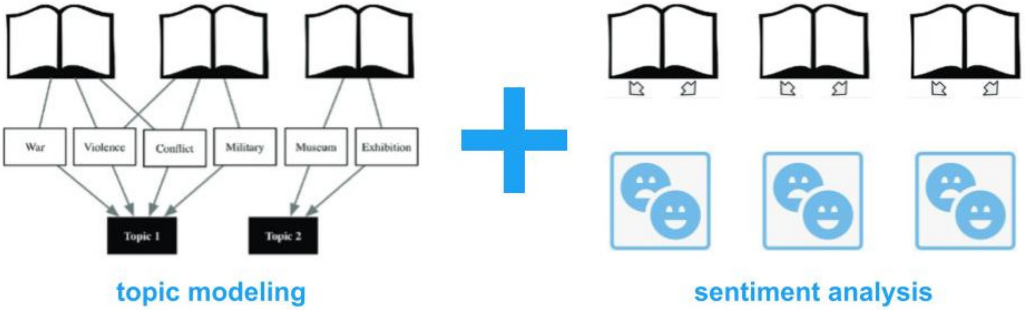


Figure 1: adopted and modified from Min and Park (2016)

→ **Topic-specific sentiment analysis**

2 TOPIC MODELING: IDEA

- **Goal**: discover latent semantic structures in a corpus & group documents into topical clusters with characteristic topic-word distributions
    - Exploratory tool → unsupervised learning task
    - Means of dimensionality reduction
- For each document $d \in \{1, 2, \dots, D\}$, assign a topic label $k \in \{1, 2, \dots, K\}$
    - K: key **hyperparameter**
    - Interpretability up to human practitioner

2 TOPIC MODELING: TAXONOMY

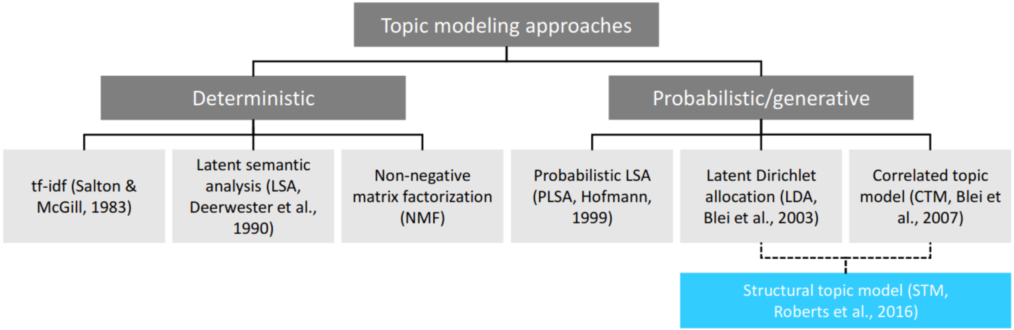


Figure 2: own representation, published on https://lisa-wm.github.io/nlp-twitter-r-bert/

2 TOPIC MODELING: GENERATIVE APPROACHES

Idea: reverse-engineer the imaginative process of document generation with hierarchical Bayesian mixture models

1. For each document $d \in \{1, 2, \ldots, D\}$, draw a vector of topic proportions from some assumed distribution

2. For each word position $n \in \{1, 2, \ldots, N_d\}$, $N_d \in \mathbb{N}$,

    1. draw a topic assignment from the distribution associated with the document-specific topic proportions

    2. draw a word from the distribution associated with the topic

2 TOPIC MODELING: GENERATIVE APPROACHES

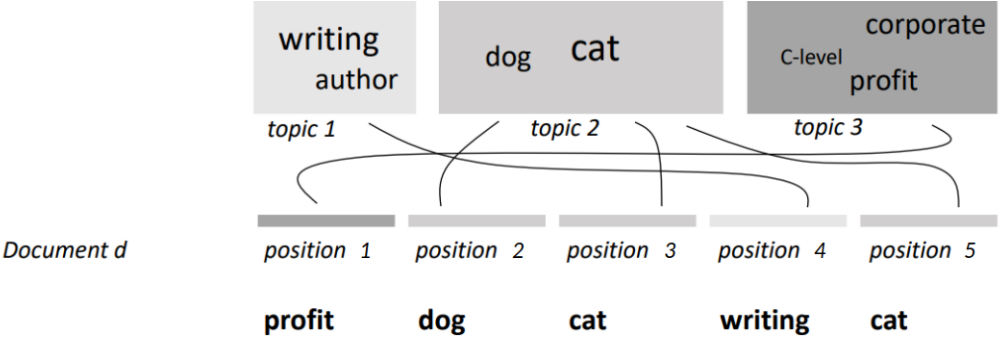


Figure 3: own representation, published on https://lisa-wm.github.io/nlp-twitter-r-bert/

2 SENTIMENT ANALYSIS

- **Goal**: assign sentiment labels to documents - in our case, out of $\{$positive, negative$\}$, formalized as $y \in \mathcal{Y} = \{0, 1\}$

- Standard **classification** task

- Find $f : \mathcal{X} \rightarrow \mathbb{R}^g$, $\mathcal{X} \subseteq \mathbb{R}^p$ for $p \in \mathbb{N}$

- Methods considered:

    - Standard ML: random forests & regularized logistic regression

    - BERT: fine-tuning to sentiment analysis

2 TOPIC-SPECIFIC SENTIMENT ANALYSIS

- **Idea:** combine **topic modeling & sentiment analysis**

- Subsequent modeling mostly due to the complexity of joint models

- Standard ML:
  - Build clusters of tweets based on topic modeling
  - Use clusters to generate topic-specific word embeddings

- BERT:
  - Aspect-based sentiment analysis (ABSA)
  - Aspect extraction & aspect sentiment classification

3 ANALYSIS

3.1 DATA

3 DATA COLLECTION: WEB SCRAPING

Idea: collect tweets by members of the German parliament (*Bundestag*) issued after the last federal election in September 2017

1. Gather MPs' names and basic information from the official Bundestag website
2. Find Twitter account names
3. Acquire socioeconomic information for the time of the last federal election on a per-district level
4. Scrape actual tweets along with some additional variables

→ **Manual labeling process**

3 DATA COLLECTION: WEB SCRAPING

Figure 4: urlhttps://www.bundestag.de/abgeordnete/

Figure 5: https://www.twitter.com/

3 DATA LABELING

- For each tweet: assign polarities **positive** or **negative**, and also **topic** descriptions required for BERT's ABSA task

- Note: large number of tweets with no apparent sentiment, aspect detection often difficult

- Class label distribution: **72%** negative labels

username	party	created_at	text	followers	unemployment_rate	label
karl_lauterbach	spd	2019-12-01 09:44:00	"Die Wahl ..."	337001	8.5	negative
Martin_Hess_AfD	afd	2018-08-17 07:15:00	"Vor den ..."	6574	3.5	negative
BriHasselmann	gruene	2019-09-25 15:35:00	"Ich finde ..."	20299	8.6	positive
danielakolbe	spd	2020-05-12 06:05:00	"Aber verpflichtend ..."	8158	8.3	negative
JuergenBraunAfD	afd	2020-08-13 22:05:00	"Panik-Latif + ..."	3188	3.4	negative

3 DATA: DISTRIBUTION OVER TIME

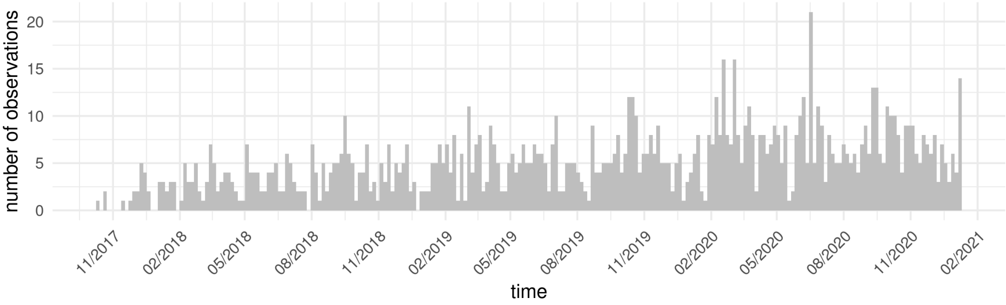


Figure 6: tweet issuance over time

Periodical fluctuations in the number of tweets over time and a general upward-sloping trend

3 DATA: DISTRIBUTION ACROSS PARTIES

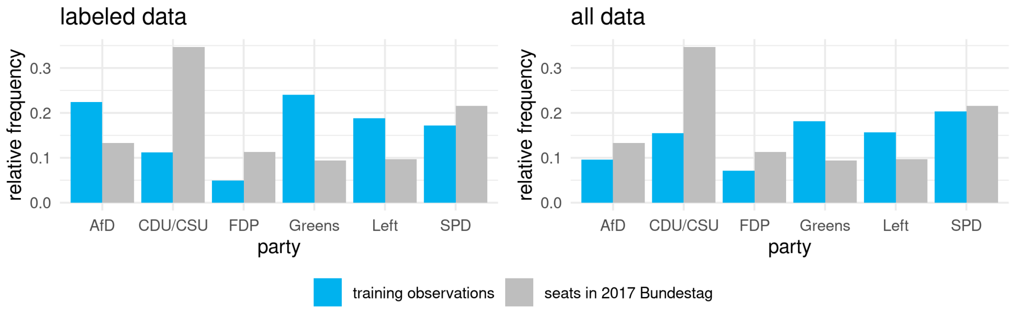


Figure 7: tweet issuance across parties

Observations per party in labeled training data (left) and entire scraped data example (right), both depicted against seat distribution in current parliament

3 DATA PRE-PROCESSING

- **Basic text cleaning:** transcription of German umlauts and ligature s into standard-Latin characters and removal of non-informative symbols

- **Twitter-specific preparation:** identification, separate storage and subsequent removal of special characters (i.e., hashtags, emojis and user tags)

Wir gedenken Willy Brandt, der heute vor 28 Jahren, am 8. Oktober 1992, verstarb. Mit seinen Reformen in der Sozial-, Bildungs- und Rechtspolitik hat er innenpolitisch neue **Massstaebe** gesetzt. **Kniefall Friedensnobelpreis mehrdemokratiewagen spd willybrandt**

3 DATA CHALLENGES

- **Language-specific:** many approaches predominantly tailored to English

  → possible complications with regards to German grammar and syntax

- **Twitter-specific:** limit of 280 characters; no explicit mentioning of the event or topical entity the author is referring to; informal language style

- **Context-specific:** requirement of domain knowledge within political context (specific vocabulary); sarcasm and irony

3.2 STANDARD MACHINE LEARNING SOLUTION

3 ANALYTICAL CONCEPT

Conceptualization as analytical **pipeline**

→ Exchangeability of components

→ Usability as integrated object

→ Preserving train-test dichotomy

→ Seamlessly integrated in `mlr3`

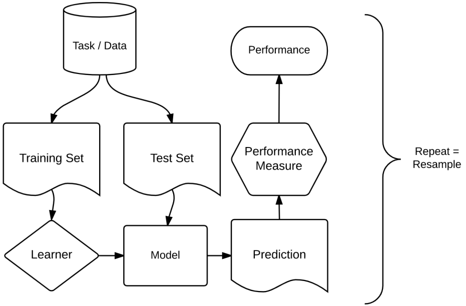


Figure 8: Becker et al. (2021)

3 FEATURE EXTRACTION

We discern two stages of feature extraction:

1. **Static features:** all quantities that may be derived from a single observation
2. **Dynamic features:** quantities that are computed across a range of observations

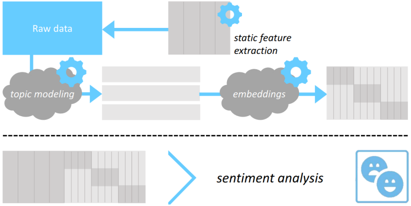


Figure 9: own representation, published on https://lisa-wm.github.io/nlp-twitter-r-bert/

→ **Difference important in resampling processes**

3 STATIC FEATURES

- **Lexicon-based polarity:** counts of positive / negative terms and emojis
- **Twitter variables:** hashtags, retweets, ...
- **Syntactic features:** intensification, negation
- **Character unigrams:** number of respective character occurrences
- **Part-of-speech (POS) tags:** number of adjectives, nouns, ...

3 DYNAMIC FEATURES

Idea: capture topical context by computing a set of word embeddings for each topic cluster

- Topic modeling with **structural topic model** (Roberts et al., 2013)

  $\rightarrow$ Additional consideration of document-level meta variables

- Embeddings with **GloVe** (Pennington et al., 2014)

3 STRUCTURAL TOPIC MODEL (STM)

- Generative model based on latent Dirichlet allocation (LDA, Blei et al. (2003))
- Recall: characterization of topics by individual topic-word distributions
- Two key enhancements:
    - Allowing for inter-topic **correlations**
    - Incorporating document-level **meta data**, either as **topical prevalence** formula or as **topical content** variables

. $\sim$ `party + bundesland + s(unemployment) + s(pop_migration)`

3 STRUCTURAL TOPIC MODEL (STM)

1. Draw non-normalized topic proportions $\boldsymbol{\eta}_\mathbf{d} \sim \mathcal{N}_{K-1}(\boldsymbol{\Gamma}^T \mathbf{x}_d^T, \boldsymbol{\Sigma})$.
2. Normalize $\boldsymbol{\eta}_\mathbf{d}$ through a softmax operation, yielding $\boldsymbol{\theta}_d$ with $\theta_{d,k} = \frac{\exp(\eta_{d,k})}{\sum_{j=1}^{K} \exp(\eta_{d,j})} \in [0, 1]$, $k \in \{1, 2, \ldots, K\}$.
3. For each word position $n \in \{1, 2, \ldots, N_d\}$:
   1. Draw $\mathbf{z}_{d,n} \sim Multinomial(\boldsymbol{\theta}_d)$ to assign the n-th position to a topic.
   2. Draw a word $w_{d,n}$ from the word distribution corresponding to the assigned topic: $w_{d,n} \sim Multinomial(\beta(d, n))$.

3 WORD EMBEDDINGS

- **Goal:** model semantic importance of words in dense numeric representation

- Also achieved by bag-of-words (BOW) approach, but with high dimensionality

- Dimensionality reduction by embedding observations into low-dimensional latent space

  - Characterize words by their surrounding context
  - Find latent dimensions
  - Similar meaning = similar representation in the vector space

3 WORD EMBEDDINGS

GloVe: Global Vectors

- Based on word co-occurrence matrix
- Neighborhood relations between words
- Defined via window size
- Underlying assumption: stronger relationship between close-lying words

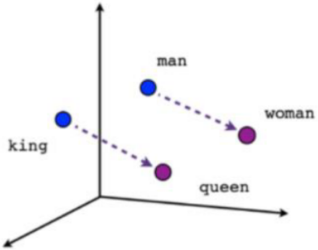


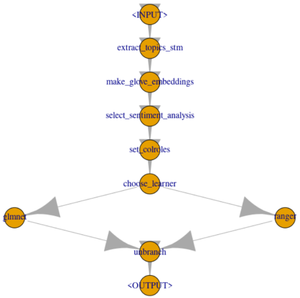
Figure 10: https://towardsdatascience.com/the-magic-behind-embedding-models-c3af62f71fb

The $\boxed{\text{quick } \textbf{brown } \textbf{fox } \textbf{jumps} \text{ over}}$ the lazy dog.

3 AUTOML PIPELINE

Implementation as `mlr3` **graph learner**

- Input: text + static features
- Computation of topic-specific embeddings
- Choice between random forest and logistic regression with elastic net penalty
- Tuning over associated configuration spaces



→ **Train, predict, resample, tune, benchmark**

3 RESULTS

wip

3.3 DEEP LEARNING SOLUTION

4 KNOWLEDGE TRANSFER

5 CONCLUSION

REFERENCES

Aggarwal, C. C. (2018). *Machine Learning for Text*, Springer.

Becker, M., Binder, M., Bischl, B., Lang, M., Pfisterer, F., Reich, N. G., Richter, J., Schratz, P. and Sonabend, R. (2021). mlr3 book.
URL: *https://mlr3book.mlr-org.com*

Benoit, K. and Matsuo, A. (2020). *spacyr: Wrapper to the 'spaCy' 'NLP' Library*. R package version 1.2.1.
URL: *https://CRAN.R-project.org/package=spacyr*

Benoit, K., Watanabe, K., Wang, H., Nulty, P., Obeng, A., Müller, S., Matsuo, A., Lowe, W. and Müller, C. (2021). *quanteda: Quantitative Analysis of Textual Data*. R package version 3.0.0.
URL: *https://CRAN.R-project.org/package=quanteda*

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*, Springer.

Blei, D. M., Ng, A. Y. and Jordan, M. I. (2003). Latent dirichlet allocation, *Journal of Machine Learning Research* **3**: 993–1022.

Breiman, L., Friedman, J. H., Olshen, R. J. and Stone, C. J. (1984). *Classification and Regression Trees*, Chapman & Hall/CRC.

Devlin, J., Chang, M., Lee, K. and Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding, *CoRR* **abs/1810.04805**.
URL: *http://arxiv.org/abs/1810.04805*

Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

Feurer, M. and Hutter, F. (2019). Hyperparameter optimization, *in* F. Hutter, L. Kotthoff and J. Vanschoren (eds), *Automated Machine Learning. Methods, Systems, Challenges*, Springer, pp. 3–34.

Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*, MIT Press. http://www.deeplearningbook.org.

Hastie, T., Qian, J. and Tay, K. (2021). An introduction to glmnet.

Japkowicz, N. and Shah, M. (2011). *Evaluating Learning Algorithms. A Classification Perspective*, Cambridge University Press.

Lang, M., Binder, M., Richter, J., Schratz, P., Pfisterer, F., Coors, S., Au, Q., Casalicchio, G., Kotthoff, L. and Bischl, B. (2019). mlr3: A modern object-oriented machine learning framework in R, *Journal of Open Source Software* .
URL: *https://joss.theoj.org/papers/10.21105/joss.01903*

Lindsey, J. K. (1997). *Applying Generalized Linear Models*, Springer.

Louppe, G. (2014). *Understanding Random Forests. From Theory to Practice*, PhD thesis, University of Liege.

Min, S. and Park, J. (2016). Mapping out narrative structures and dynamics using networks and textual information.

Murphy, K. P. (2021). *Probabilistic Machine Learning: An Introduction*, MIT Press.

Pan, S. J. and Yang, Q. (2010). A survey on transfer learning, *IEEE Transactions on Knowledge and Data Engineering* **22**(10): 1345–1359.

Pavlopoulos, I. (2014). *Aspect-Based Sentiment Analysis*, PhD thesis, Athens University of Economics and Business.

Pennington, J., Socher, R. and Manning, C. (2014). GloVe: Global vectors for word representation, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Doha, Qatar, pp. 1532–1543.
URL: *https://www.aclweb.org/anthology/D14-1162*

R Core Team (2021). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.

Richardson, L. (2007). *Beautiful Soup Documentation*.

Roberts, M. E., Stewart, B. M. and Airoldi, E. M. (2016). A model of text for experimentation in the social sciences, *Journal of the American Statistical Association* **111**(515): 988–1003.

Roberts, M., Stewart, B., Tingley, D. and Airoldi, E. (2013). The structural topic model and applied social science, *Advances in Neural Information Processing Systems Workshop on Topic Models*, pp. 1–20.

Roberts, M., Stewart, B., Tingley, D. and Benoit, K. (2020). *stm: Estimation of the Structural Topic Model*. R package version 1.3.6.
URL: *https://CRAN.R-project.org/package=stm*

Roesslein, J. (2020). *Tweepy: Twitter for Python!*

Ruder, S. (2019). *Neural Transfer Learning for Natural Language Processing*, PhD thesis, National University of Ireland, Galway.

Schulze, P. and Wiegrebe, S. (2020). Twitter in the parliament - a text-based analysis of german political entities, *Technical report*, Ludwig-Maximilians-Universität, Munich.

Selivanov, D., Bickel, M. and Wang, Q. (2020). *text2vec: Modern Text Mining Framework for R*. R package version 0.6.
URL: *https://CRAN.R-project.org/package=text2vec*

van Rossum, G. and Drake, F. L. (2011). *The Python Language Reference Manual*, Network Theory Ltd.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. and Polosukhin, I. (2017). Attention is all you need, *CoRR* **abs/1706.03762**.
URL: *http://arxiv.org/abs/1706.03762*

Vayansky, I. and Kumar, S. A. (2020). A review of topic modeling methods, *Information Systems* **94**.

Xu, H., Liu, B., Shu, L. and Yu, P. S. (2019). Post-training for review reading comprehension and aspect-based sentiment analysis, *Proceedings of NAACL-HLT*, Minneapolis, USA, p. 2324–2335.

Zhang, A., Lipton, Z. C., Li, M. and Smola, A. J. (2020). *Dive into Deep Learning*. https://d2l.ai.