

# Spezifikationsvortrag

Fabian Düker, Uli Steinbach

Universität Heidelberg, Institut für Computerlinguistik  
Softwareprojekt, SoSe 2018

Prof Dr. Katja Markert

12.06.2018

# Übersicht I

## Übersicht

Aufgabe

## Inhaltliche Spezifikation

inh. Spezifikation

## Modularisierung und Aufgabenverteilung

Modularisierung und Aufgabenverteilung

## Programmarchitektur, Datenstrukturen

Programmarchitektur und Datenstrukturen

# Übersicht

Autom. Erstellung eines Lexikons für die Erkennung von  
Abusive Words

Anwendung auf Germeval 2018 Task I  $\Rightarrow$  Binäre Klassifikation von  
5000 Tweets

# Problemstellung

- ▶ Problem: Hatespeech ist in ständiger Veränderung begriffen (Neologismen, Ambiguität, Kontext/Domäne)
- ▶ Wiegand et al. 2016: Erstellung eines englischen Lexikons mit guten Ergebnissen auf cross-domain Evaluation
- ▶ SentiWS: Lexikon mit negativen Wörtern für das Deutsche

# Lösungsansatz

- ▶ Erstellung Baselexikon aus SentiWS neg. Sentiment-Lexikon
- ▶ Halbautomatische Erweiterung des Baselexikons mit deutschen Schimpfwörtern
- ▶ Autom. Erweiterung mittels graphbasiertem Label-Propagation-Algorithmus
- ▶ Anwendung auf Germeval 2018 Datenset und Evaluation

# Erstellung des Baselexikons

## SentiWS

- ▶ Extraktion negativer Wörter aus SentiWS
- ▶ 686 Nomen
- ▶ 420 Verben
- ▶ 708 Adjektive
- ▶ Problem: Zu wenige explizite Schimpfwörter
- ▶ Lösung: Mehr Schimpfwörter hinzufügen

# Erweiterung des Baselexikons

- ▶ Genius API: Erstellung eines Deutschrapkorpus
- ▶ Deutschrap: zeitgemäße Verwendung von Schimpfwörtern (genrespezifisch, aber auch politisch, rassistisch, sexistisch)
- ▶ autom. Extraktion von Kandidaten mittels syntaktischer Pattern

# Erweiterung des Baselexikons

- ▶ Automatischer Abgleich aller Nomen im Rapkorpus
- ▶ mit Schimpfwortliste aus dem Internet
- ▶ mit Pattern "du [NN]"
- ▶ etwa 280 potentielle Schimpfwörter
- ▶ manuelles Aussortieren von false positives (z.B. "Rapper")
- ▶ Auswahl der 200 häufigsten Schimpfwörter
- ▶ Erweiterung durch beleidigende Adjektive
- ▶ Suche nach Pattern "du [ADJ] Schimpfwort"
- ▶ etwa 280 potentiell beleidigende Adjektive



# Erweiterung des Baselexikons

- ▶ Lemmatisierung mit IWNLP
- ▶ Lemmatisierung der nicht erkannten Adjektive von Hand
- ▶ Beseitigung von Duplikaten
- ▶ Finales Baselexikon:
  - ▶ 887 Nomen
  - ▶ 413 Verben
  - ▶ 824 Adjektive
  - ▶ - 2124 Wörter

# Rapkorpus

- ▶ Texte von 30 Rappern (Auswahl angelehnt an Daten-Journalismus Studie des BR/PULS zum Thema "Diskriminierung im Deutschrapp" aus dem Jahr 2016)
- ▶ Bushido
- ▶ Chakuza
- ▶ K.I.Z.
- ▶ Kay One
- ▶ Kollegah & Farid Bang
- ▶ Prinz Pi
- ▶ Bass Sultan Hengzt
- ▶ Fler
- ▶ Azad
- ▶ Kool Savas
- ▶ ...

# Rapkorpus

Auszug der extrahierten Schimpfwörter vor Handselektion

- ▶ *Rapper*
- ▶ *Kopf*
- ▶ **Arsch**
- ▶ **Bitch**
- ▶ **Schwanz**
- ▶ **Scheiße**
- ▶ *Gangster*
- ▶ *Block*
- ▶ *Baby*
- ▶ **Nutte**

# Baseline

- ▶ Baseline 1: Unigram und Bigram SVM
- ▶ Baseline 2: Feature Selection (Mutual Information) SVM
- ▶ Preprocessing: Autosarkasmus-SP (SoSe 2016): Alle Tweets wurden tokenisiert, normalisiert und pos-getagged. Zusätzlich Lemmatisierung und stopword removal
- ▶ 10-Fold Cross Validation mit random-seed für bessere Vergleichbarkeit

# Baseline

- ▶ Unigram und Bigram SVM: Es wurde ein SVM Klassifizierer mit Standard-Parametern (Regularisierungs-Parameter  $C = 1.0$ , linear Kernel) trainiert. Input-Features sind die auf der Dokument-Term Matrix berechneten tf-idf Werte für Uni- bzw. Bigramme. Für die Unigramme wurde ein unterer cutoff von mind. 2 Vorkommen festgelegt.
- ▶ Feature Selection Algorithmus: Berechnung des mutual information score (Manning et. al 2011) zwischen Label und Wort  $\Rightarrow$  1500 Wörter mit den höchsten mi-scores wurden als Input-Features für den SVM Klassifizierer (Regularisierungsparameter  $C=1.0$ , linearer Kernel) benutzt.

## Evaluation der Baseline

Table: Baseline: tf-idf unigram SVM

[illegible]

## Evaluation der Baseline

Table: Baseline: tf-idf bigram SVM

[illegible]

## Evaluation der Baseline

Table: Baseline: Feature-Selection m. Mutual Information

[illegible]



# Graph-basierter Ansatz für autom. Erweiterung des Baselexikons

- ▶ Erstellung von pos + neg seed-Liste mit annotierten Schimpfwörtern aus Baselexikon (+) und häufigsten Wörtern (-)
- ▶ Graph mit Kanten zw. Wörtern auf Basis von Kosinusähnlichkeit zwischen Word Embeddings Vektoren (auf Twitter Korpus trainiert)
- ▶ Propagierung der Seed-Labels auf ungelabelte Knoten/Wörter mittels graphbasiertem Label-Propagation-Algorithmus (Adsorption Algorithmus, Talukdar et al. 2008)

# Graph-basierter Ansatz für autom. Erweiterung des Baselexikons

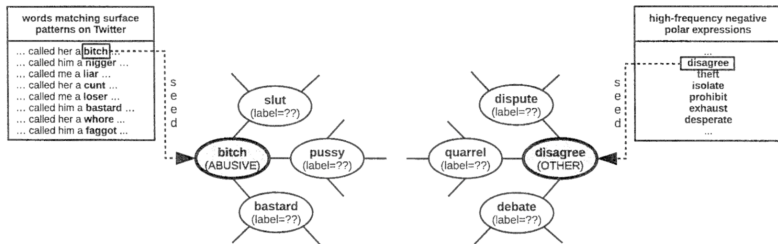


Figure: Label Propagation Graph aus Wiegand et al. 2018

- ▶ Paketierung: Aufsplittung des Gesamtpakets in kleinere Module
- ▶ Python-Programmierung (objektorientierte Implementierung)
- ▶ "split und shared tasks"

Modul		Uli	Fabian
<b>Erstellung Baselexikon</b>	PAT-basierte Extraktion von NNs/ADJs	x	x
	Skript für Erstellung eines Annotations-Testsets	x	
	Skript für autom. Auswertung d. Annotations-Testsets		x
<b>Erstellung Baselines</b>	Implementation manuelle 10-Fold Cross Validation	x	
	Integration Autosarkasmus-Tweet Preprocessing	x	
	Erweitertes Preprocessing (Lemmatisierung, Stopwörter)	x	
	Implementation Unigram/Bigram SVM Baseline	x	
	Implementation Mutual Information (Manning et. al 2011)	x	
	Implementation Feature Selection (MI) SVM Baseline	x	
	Evaluation und Output	x	
<b>Erstellung Word-Similarity Graph</b>	Word Embeddings auf Twitter Daten	x	x
	Erstellung des Wortähnlichkeitsgraphen auf Basis von Kosinusähnlichkeiten	x	x
	Unknown Words Handhabung (character-level embeddings?)	x	x
	Erstellung der Seed Listen (pos + neg)	x	x
<b>Label Propagation</b>	Implementation Adsorption Algorithmus (Talukdar 2008)	x	x
	Erweiterung des Baselexikons mit Output	x	x
<b>Anwendung und Evaluation</b>	Test auf Germeval Daten	x	x
	Verbesserungen und Erweiterungen	x	x
	Visualisierung des Outputs	x	x
	Präsentation der Ergebnisse	x	x
	Abschlussbericht	x	x

# Aufgabenverteilung

# Zeitplan

# Programmarchitektur

# Datenstrukturen



# Literatur

- ▶ Talukdar 2008, Manning et al. 2011, Autosarkasmus Gruppe 2016, BR/PULS Studie 2016, Wiegand et al. 2018