

# Sentimentanalyse auf Amazon-Reviews

Caroline Berg  
Simon Will

Institut für Computerlinguistik  
Ruprecht-Karls-Universität Heidelberg  
Dozentin: Éva Mújdricza-Maydt  
WS 2015/2016

3. Februar 2016

**1** Überblick

**2** Projektablauf

**3** Evaluation und Experimente

# Table of Contents

**1** Überblick

2 Projektablauf

3 Evaluation und Experimente

# Einführung

## ■ Unser Projekt: Sentimentanalyse auf Amazon-Reviews

10 von 11 Kunden fanden die folgende Rezension hilfreich.

★★★★★ **Samsung Galaxy J5 Smartphone (5 Zoll (12,7 cm))...**

Von [K. Reinhold](#) am 10. November 2015

Farbe: Schwarz | **Verifizierter Kauf**

Dieses Handy ist sehr gut, fast genauso wie ein S5 nur billiger, Der Akku hält für meine Bedürfnisse auch sehr lange. Es ist genau die richtige Größe, sehr handlich und gut. Ich bin sehr zufrieden damit.

- Ziel: Anzahl der für ein Produkt vergebenen Sterne (1–5) aus dem Reviewtext vorhersagen
- Für einen Teil der Auswertung wurden die fünf Klassen binarisiert zu *gut* und *schlecht*.

# Vergleichbare Projekte

- Callen RAIN: Sentiment Analyses in Amazon Reviews Using Probabilistic Machine Learning.
- Reviews aus Büchern, Filmen, CDs, ...
- Klassifizierung:
  - 1–2 Sterne: score 0
  - 5 Sterne: score 1

# Vergleichbare Projekte

- Feature-Extraction bei RAIN
  - *bag of words*: Vorkommen der 2000 häufigsten Wörter in Reviews (binär)
  - *collocations*: *bigram function* von NLTK (500 häufigste Bigramme, analog zu bag of words)
  - *negation*: nach einer Negation werden die drei darauffolgenden Wörter mit (*NOT*) annotiert
  - *sentence length*:
    - sentence feature für kurze Sätze ( $< 10$ )
    - sentence feature für lange Sätze ( $> 20$ )
  - Ergebnisse (Accuracy):
    - Amazon Kindle (Naive Bayes): 84 %
    - MacBook (Naive Bayes): 88.2 %

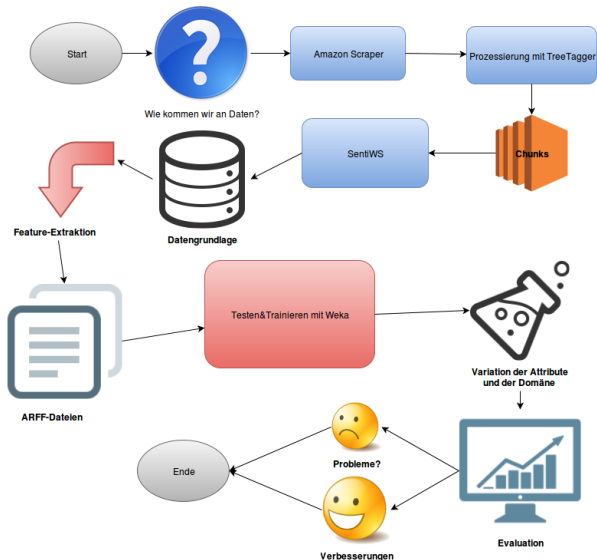
# Table of Contents

1 Überblick

2 Projektablauf

3 Evaluation und Experimente

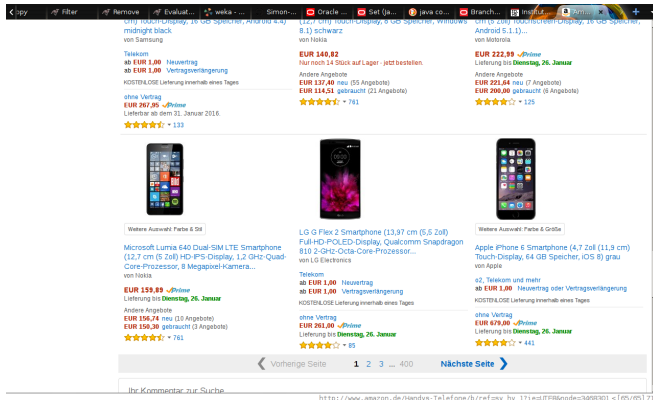
# Flussdiagramm





# Amazon Scraper

- geschrieben in python3 mit urllib2, BeautifulSoup und lxml
- ruft eine Überblicksseite auf Amazon auf, auf der Projekte aufgelistet sind und lädt alle Reviews herunter.



http://www.amazon.de/Handys-Telefone/b/ref=sv\_hv\_1?ie=UTF8&node=3408301<[65/65] 71%

# Vorverarbeitung

- Mit dem TreeTagger und den mit ihm gelieferten Skripten wurden die Reviews
  - tokenisiert,
  - mit POS-Tags versehen und
  - lemmatisiert.

```
1  Dieses  PDAT    dies
2  Handy   NE     Handy
3  ist     VAFIN   sein
4  absolut ADJD    absolut
5  empfehlenswert ADJD empfehlenswert
6  .       $.     .
7  Sehr    ADV     sehr
8  leichte ADJA    leicht
9  Bedienung NN    Bedienung
10 und     KON     und
11 sieht   VVFIN   sehen
12 auch    ADV     auch
13 noch    ADV     noch
14 top     ADJD    <unknown>
15 modisch ADJD    modisch
```

# Einteilung in Chunks

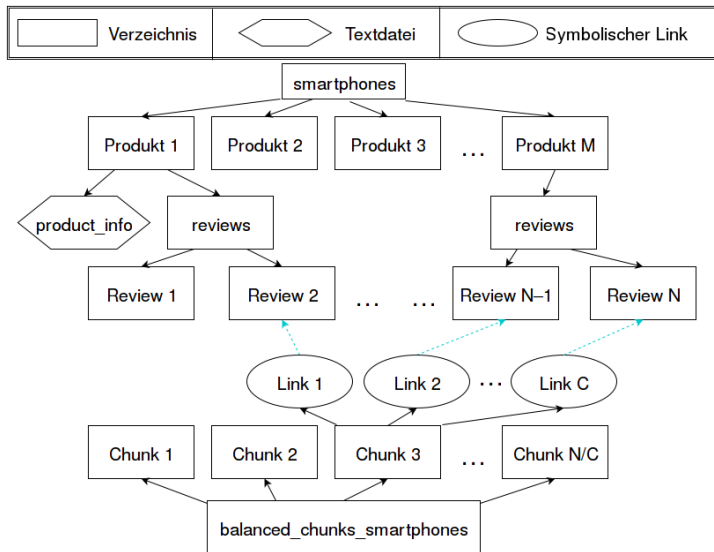
Sternverteilung der Smartphone-Reviews:

| Sterne | 1     | 2    | 3    | 4     | 5     | Gesamt |
|--------|-------|------|------|-------|-------|--------|
| Anzahl | 13231 | 6645 | 8794 | 21826 | 77310 | 127806 |

⇒ Eine Ausbalancierung ist nötig.

- Erstellung einer Struktur aus symbolischen Links
- Jeder Symlink zeigt zufällig auf ein Review.
- Anordnung der Symlinks in balancierten Verzeichnissen (*Chunks*)
- geschrieben in perl5
- noch 33500 Reviews nach dem Ausbalancieren übrig

# Einteilung in Chunks



# Ressource: SentiWS

- *SentiWS* (kurz für „Sentimentwortschatz“) ist eine Datenbank, die deutschen Wörtern Zahlen von  $-1$  bis  $1$  zuordnet, die das Sentiment der Wörter darstellen soll.
- Enthalten sind etwa 33000 Wortformen zu etwa 3500 Lemmata.

```
1  Zusammengehörigkeit|NN    0.0040
2  Zusammenhalt|NN  0.1947  Zusammenhaltes ,Zusammenhalte ,
   Zusammenhalts ,Zusammenhalten
3  Zuschuß|NN    0.0040  Zuschüße ,Zuschußes ,Zuschüßen ,
   Zuschuß
4  Zustimmung|NN    0.0040  Zustimmungen
5  Zuverlässigkeit|NN    0.0040
6  Zuversicht|NN    0.0727
7  abgesichert|ADJX    0.0040
8  abmachen|VVINF  0.0040  abgemacht ,abmachst ,abmach ,
   abmachet ,abmachen ,abmacht ,abmache ,abmachtest ,
   abmachte ,abmachest ,abmachtet
9  abschließen|VVINF    0.0040
```

# Feature-Extraktion

- geschrieben in python3 (und sh, bash und awk)
- Attribute: *token\_number*, *overall\_sentiment*, *adjective\_sentiment*, *noun\_sentiment*, *verb\_sentiment*
- Die Sentiment-Attribute existieren auch in einer auf die *token\_number* normierten Version.<sup>1</sup>
- Klassenattribute: *stars* und *binary\_judgement*

```
1 adjective_sentiment 0.6574 numeric
2 binary_judgement    good    {good, bad}
3 normalized_adjective_sentiment 28582.6 numeric
4 normalized_noun_sentiment    0    numeric
5 normalized_overall_sentiment 28582.6 numeric
6 normalized_verb_sentiment    0    numeric
7 noun_sentiment    0    numeric
8 overall_sentiment    0.6574 numeric
9 stars    5    {1, 2, 3, 4, 5}
10 token_number    23    numeric
11 verb_sentiment    0    numeric
```

<sup>1</sup>  $\text{normalized\_overall\_sentiment} = \frac{\text{overall\_sentiment} * 10^6}{\text{token\_number}}$

# ARFF-Datei

```
1 @relation Review_sentiment
2
3 @attribute token_number numeric
4 @attribute normalized_overall_sentiment numeric
5 @attribute normalized_adjective_sentiment numeric
6 @attribute normalized_noun_sentiment numeric
7 @attribute normalized_verb_sentiment numeric
8 @attribute stars {1,2,3,4,5}
9
10 @data
11 510,2695.1,2297.45,3.92157,403.137,1
12 33,-11590.9,121.212,-11712.1,0,1
13 148,-5214.86,1139.19,54.0541,-3173.65,2
14 152,1219.74,3196.05,-2055.26,78.9474,1
15 57,-7524.56,-4059.65,0,-3464.91,2
16 36,8580.56,8469.44,111.111,0,4
17 70,4445.71,13665.7,-5521.43,3140,4
18 28,55332.1,39889.3,0,15442.9,4
19 92,20113,13018.5,86.9565,7007.61,5
20 48,-13893.8,-13793.8,-100,0,3
```

# Table of Contents

1 Überblick

2 Projektablauf

3 Evaluation und Experimente



# Benutzte Klassifizierer

- J48 mit
  - U „Use unpruned tree“
- RandomForest mit
  - I 10 „Number of trees to build“
  - K 0 „Number of features to consider“
  - S 1 „Seed for random number generator“
  - depth 200 „The maximum depth of the trees“
- NaiveBayes ohne Optionen

# Hauptsächliche Ergebnisse

Mit zehnfacher Kreuzvalidierung ergeben sich folgende Ergebnisse:

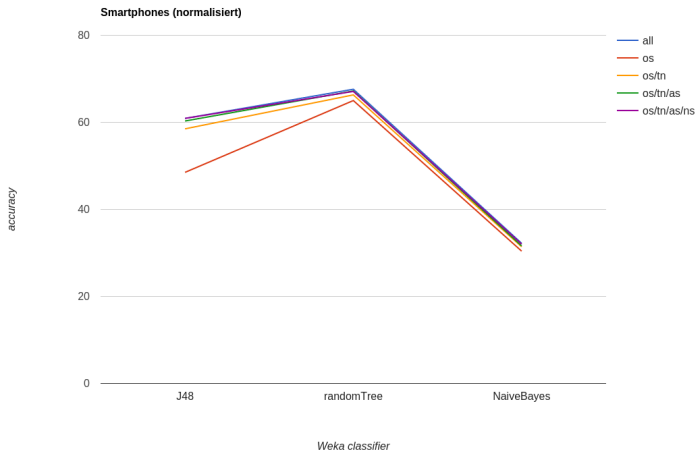
| Klassen | normiert | Majority Voting | J48    | RandomForest | Naive Bayes |
|---------|----------|-----------------|--------|--------------|-------------|
| 1-5     | nein     | 20 %            | 61.0 % | 67.4 %       | 31.6 %      |
| 1-5     | ja       | 20 %            | 61.0 % | 67.6 %       | 32.5 %      |
| binär   | ja       | 60 %            | 75.0 % | 84.6 %       | 69.8 %      |

Wird ein Teil der Daten eigens zum Testen abgetrennt, verringert sich die Accuracy etwas (von 67.6 % beim RandomForest auf 64.7 %). Das liegt wohl einfach daran, dass dann auf weniger Daten trainiert wird.

# Variieren der Trainingsdatenmenge



# Attributselektion



# Test auf fremder Domäne

- Test des mit den normierten Sentimentattributen und *token\_number* auf Smartphones trainierten Modells auf Armbanduhren
- Erreichte Accuracy: 27.7 %
- Ein Faktor eventuell: Armbanduhren-Reviews sind nur etwa ein Drittel so lang wie Smartphone-Reviews.

# Fehlersuche

## Review mit 5 Sternen

*Super Handy, Habe mir dieses Handy bestellt, weil wir schon das gleiche in der Familie haben. Einfache Bedienung, tolle Menüführung. Einfach Klasse für einen guten Preis.*

# Fehlersuche

## Review mit 5 Sternen

*Ich persönlich komme vom Nokia E51 :) und habe über nun fast ein halbes Jahr verschiedene Smartphones [...] begutachtet. Ebenfalls im Focus hatte ich (Nachteile ggü. dem S3 hier aufgeführt): Motorola RAZR MAXX: kein Wechselakku, Andoid 4.1 ungewiß; HTC One X: kein Wechselakku, kein Glonass, kein SD-Schacht Samsung Galaxy Note: schlechtere Performance, Andoid 4.1 ungewiß; Samsung Galaxy Ace 2: schlechtere Performance, wenig Flashspeicher, Andoid 4.1 ungewiß; Meine Eindrücke [...] des S3 kurz zusammengefasst: positiv:- Display: hohe Auflösung, schöne Farben, die Größe möchte ich wirklich nicht mehr missen [...]*

# Schluss

- Überraschend gute Ergebnisse für den naiven Ansatz.
- Weitere mögliche Schritte:
  - Einbezug von Textstruktur
  - Einbezug von Negation und anderen privativen Ausdrücken
- Problem: Textgattung „Amazon-Review“ nicht sehr einheitlich



# Literatur

RAIN, Callen: *Sentiment Analysis in Amazon Reviews Using Probabilistic Machine Learning*. Swarthmore : Department of Computer Science, Swarthmore College, 2013.

REMUS, R.; QUASTHOFF, U.; HEYER, G.: *SentiWS – a Publicly Available German-language Resource for Sentiment Analysis*. In: *Proceedings of the 7th International Language Ressources and Evaluation (LREC'10)*, Valetta : 2010, S. 1168–1171.

SCHMID, Helmut.: *Probabilistic Part-of-Speech Tagging Using Decision Trees*. In: *Proceedings of International Conference on New Methods in Language Processing*, Manchester : 1994.