



## Projektbericht

# Sentimentanalyse von Produktrezensionen

Studiengang  
"Angewandte Künstliche Intelligenz"

**Natural Language Processing**  
**DLBAIPNLP01\_D**

Dawid Jedlinski

Matrikelnummer: IU14113900

dawid.jedlinski@iu-study.org

Tutor: Maja Popovic

Abgabedatum: dd.mm.yyyy

# Inhaltsverzeichnis

<b>I. Abbildungsverzeichnis</b>	<b>III</b>
<b>II. Tabellenverzeichnis</b>	<b>IV</b>
<b>III. Abbreviations</b>	<b>V</b>
<b>1. Einleitung</b>	<b>1</b>
1.1. Problemstellung . . . . .	1
1.2. Zielsetzung . . . . .	1
1.3. Vorgehensweise . . . . .	1
<b>2. Explorative Datenanalyse</b>	<b>2</b>
2.1. Quelle/Herkunft . . . . .	2
2.2. Datenumfang und Aufteilung . . . . .	2
2.3. Verteilung der Bewertungen . . . . .	2
<b>3. Datenvorverarbeitung</b>	<b>3</b>
3.1. Data Cleaning . . . . .	3
3.2. Textaufbereitung . . . . .	3
3.3. Numerische Repräsentation . . . . .	3
3.4. Trennung von Eingabe- und Ausgabedaten . . . . .	3
<b>4. Modell und Methodik</b>	<b>4</b>
4.1. Lineare Regression . . . . .	4
4.2. Naive Bayes . . . . .	4
4.3. Trainingsprozess . . . . .	4
<b>5. Evaluation und Ergebnisse</b>	<b>5</b>
5.1. Evaluationsmetriken . . . . .	5
5.2. Gesamtergebnisse . . . . .	5
5.3. Ergebnisse nach Produktkategorien (optional) . . . . .	5
5.4. Ergebnisse von zwei Algorithmen (optional) . . . . .	5
<b>6. Schluss</b>	<b>6</b>

## **I. Abbildungsverzeichnis**

## **II. Tabellenverzeichnis**

### **III. Abbreviations**

<b>EDA</b>	Explorative Datenanalyse
<b>GMM</b>	Gaussian Mixture Model
<b>HR</b>	Human Resources
<b>LLE</b>	Locally Linear Embedding
<b>MDS</b>	Multidimensional Scaling
<b>MH</b>	Mental Health (Mentale Gesundheit)
<b>ML</b>	Machine Learning
<b>NLP</b>	Natural Language Processing
<b>OSMI</b>	Open Sourcing Mental Illness
<b>PCA</b>	Principal Component Analysis

# **1. Einleitung**

- irgendwas... \_\_\_\_\_

## **1.1. Problemstellung**

- Bedeutung von Online-Produktrezensionen - Herausforderung der manuellen Auswertung - Relevanz automatischer Sentimentanalyse

## **1.2. Zielsetzung**

- Entwicklung eines Textklassifikationssystems - Vorhersage numerischer Bewertungen (1–5 Sterne) - Fokus auf Mehrklassen-Klassifikation

## **1.3. Vorgehensweise**

- Überblick über Daten, Methodik und Evaluation - Kurze Beschreibung der einzelnen Kapitel

## **2. Explorative Datenanalyse**

- irgendwas ... ——————

### **2.1. Quelle/Herkunft**

- Amazon Reviews'23 Datensatz (McAuley Lab) - Allgemeine Beschreibung des Datensatzes

### **2.2. Datenumfang und Aufteilung**

- Kategorien: allbeauty, handmadeProducts und healthAndPersonalCare, da sie von der Größe ähnlich sind (knapp 300MB) und unterschiedlich AllBeauty - 701.528 HandmadeProducts - 664.162 HealthAndPersonalHealth - 494.121 - es gibt 10 verschiedene Spalten, hauptsächlich type object, aber auch DateTime und 2 int64 für Rating und HelpfulVote, und bool bei verified\_purchase - zum Glück gibts bei keiner Spalte fehlende Werte

### **2.3. Verteilung der Bewertungen**

- AllBeauty mean bei 3.96 - HandmadeProducts mean bei 4.49 - HealthAndPersonalHealth mean bei 3.99 also sehr positiver Datensatz - std ist bei allen auch nicht groß liegt bei 1.4 ungefähr - die meisten Ratings sind 5, also später die Klassen ausgleichen, und Datenmenge sowieso reduzieren - wichtig zu erwähnen ist, dass bei handmade bei 5% reviews der Einkauf nicht verifiziert wurde, bei AllBeauty und Healthcare sind es sogar 10% - die Textlänge ist bis auf wenige Ausreißer bei AllBeauty und HealthCare gleich, Handmade hat etwas kürze - Bei Länge von Bewertungen, sieht man das min = 0 ist. D.h. obwohl es keine missing-values gab, gibt es trotzdem Bewertungen ohne Text, die man bei der Vorverarbeitung entfernen muss

## **3. Datenvorverarbeitung**

- irgendwas über Preprocessing ——————

### **3.1. Data Cleaning**

- Datenmenge reduzieren und die Verteilung von Rating ausgleichen - alle reviews löschen die nicht verifizierten Kauf haben - alle mit Textlänge > 10 löschen - alle spalten löschen die unnötig sind (es bleiben nur: rating, title, text)

### **3.2. Textaufbereitung**

- Zusammenführung von Titel und Text - lowercasing - Tokenisierung - Entfernen von Stopwörtern - Umgang mit Sonderzeichen (Punctuation) - Lemmatization (langsamer, aber genauer als Stemming und das ist hier wichtiger)

### **3.3. Numerische Repräsentation**

- Verwendung von TF-IDF da viele Wörter redundant (product, use, one, etc.), hebt wichtige Wörter hervor (excellent, terrible, perfect) BoW zählt nur Wörter und beachtet keine Relevanz - Begründung der Feature-Darstellung

### **3.4. Trennung von Eingabe- und Ausgabedaten**

- Definition der Zielvariable (numerische Bewertung) EINGABE (Features, X): Text = Titel + Beschreibung AUSGABE (Target, y): Rating = 1, 2, 3, 4, 5 Sterne das vielleicht erst beim Modell und als Funktion! (text als eingabe und dann die gleiche Reihenfolge wie bei Preprocessing) - Aufteilung in Trainings- und Testsatz

## **4. Modell und Methodik**

- Problemformulierung als Klassifikationsaufgabe - Supervised Learning - Mehrklassen-Klassifikation (5 Klassen)
- 

### **4.1. Lineare Regression**

- [1] discriminative - [1] kategorien selbst bestimmen - Beschreibung des gewählten Klassifikators - Begründung der Eignung für Textklassifikation

### **4.2. Naive Bayes**

- [1] generative - [1] rechnet mit wahrscheinlichkeiten - Beschreibung des gewählten Klassifikators - Begründung der Eignung für Textklassifikation

VERGLEICH - Both methods are simple; Naive Bayes is the simplest one. - Both methods are interpretable: you can look at the features which influenced the predictions most (in Naive Bayes - usually words, in logistic regression - whatever you defined). - Naive Bayes is very fast to train - it requires only one pass through the training data to evaluate the counts. For logistic regression, this is not the case: you have to go over the data many times until the gradient ascent converges. - Naive Bayes is too "naive" it assumed that features (words) are conditionally independent given class. Logistic regression does not make this assumption - we can hope it is better. - Both methods use manually defined feature representation (in Naive Bayes, BOW is the standard choice, but you still choose this yourself). While manually defined features are good for interpretability, they may be no so good for performance - you are likely to miss something which can be useful for the task.

### **4.3. Trainingsprozess**

- Training auf dem Trainingsdatensatz - Keine separate Validierungsmenge - Begründung des Vorgehens

## **5. Evaluation und Ergebnisse**

- irgendwas... ——————

### **5.1. Evaluationsmetriken**

- Accuracy - Confusion Matrix - Begründung der Metrikauswahl

### **5.2. Gesamtergebnisse**

- Accuracy auf dem Testsatz - Darstellung und Interpretation der Confusion Matrix

### **5.3. Ergebnisse nach Produktkategorien (optional)**

- Vergleich der Modellleistung zwischen Kategorien - Kurze Analyse der Unterschiede

### **5.4. Ergebnisse von zwei Algorithmen (optional)**

- Vergleich der Modellleistung zwischen Algorithmen (z.B. Naive Bayes und NN) - Kurze Analyse der Unterschiede

## **6. Schluss**