



UNIVERSITÄT ZU LÜBECK  
INSTITUT FÜR  
THEORETISCHE INFORMATIK

## Topic-Modellierung für die Zuordnung von Kundenanfragen zu Sachbearbeitern

*Topic Modeling ...*

### **Bachelorarbeit**

verfasst am

**Institut für Informationssysteme**

im Rahmen des Studiengangs

**Informatik**

der Universität zu Lübeck

vorgelegt von

**Leonard Brenk**

ausgegeben und betreut von

**Prof. Dr. Ralf Möller**

mit Unterstützung von

**Dr. Jinghua Groppe, Felix Kuhr, Magnus Bender**

Lübeck, den 1. Oktober 2021

### Eidesstattliche Erklärung

*Ich erkläre hiermit an Eides statt, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.*

---

Leonard Brenk

## Zusammenfassung

Es ist nicht leicht, eine Abschlussarbeit so zu schreiben, dass sie nicht nur inhaltlich gut ist, sondern es auch eine Freude ist, sie zu lesen. Diese Freude ist aber wichtig: Wenn die Person, die die Arbeit benoten soll, wenig Gefallen am Lesen der Arbeit findet, so wird sie auch wenig Gefallen an einer guten Note finden. Glücklicherweise gibt es einige Kniffe, gut lesbare Arbeiten zu schreiben. Am wichtigsten ist zweifelsohne, dass die Arbeit in gutem Deutsch oder Englisch verfasst wurde mit klarem Satzbau und gutem Sprachrhythmus, dass keine Rechtschreib- oder Grammatikfehlern im Text auftauchen und dass die Argumente der Autorin oder des Autors klar, logisch, verständlich und gut veranschaulicht dargestellt werden. Daneben sind aber auch gut lesbare Schriftbilder und ein angenehmes Layout hilfreich. Die Nutzung dieser L<sup>A</sup>T<sub>E</sub>X-Vorlage hilft der Schreiberin oder dem Schreiber dabei zumindest bei Letzterem: Sie umfasst gute, sofort nutzbare Designs und sie kümmert sich um viele typographische Details.

## Abstract

It is not easy to write a thesis that does not only advance science, but that is also a pleasure to read. While the scientific contribution of a thesis is undoubtedly of greater importance, the impact of *writing well* should not be underestimated: If the person who grades a thesis finds no pleasure in the reading, that person are also unlikely to find pleasure in giving outstanding grades. A well-written text uses good German or English phrasing with a clear and correct sentence structure and language rhythm, there are no spelling mistakes and the author's arguments are presented in a clear, logical and understandable manner using well-chosen examples and explanations. In addition, a nice-to-read font and a pleasing layout are also helpful. The L<sup>A</sup>T<sub>E</sub>X class presented in this document helps with the latter: It contains a number of ready-to-use designs and takes care of many small typographical chores.

## Danksagungen

This is the place where you can thank people and institutions, do not try to do this on the title page. The only exception is in case you wrote your thesis while working or staying at a company or abroad. Then you should use the Weitere\_□\_Unterstützung key to provide a text (in German) that acknowledges the company or foreign institute. For instance, you could use texts like »Die Arbeit ist im Rahmen einer Tätigkeit bei der Firma Muster GmbH entstanden« or »Die Arbeit ist im Rahmen eines Forschungsaufenthalts beim Institut für Dieses und Jenes an der Universität Entenhausen entstanden«. Do not name and thank individual persons from the company or foreign institute on the title page, do that here.

# Inhaltsverzeichnis

1	Einleitung	1
1.1	Contributions of this Thesis	1
1.2	Related Work	1
1.3	Struktur dieser Arbeit	1
1.4	Motivation	2
1.5	Ziel	3
2	Grundlagen	5
2.1	Notation	5
2.2	Modellvergleich	5
2.3	Grundlagen der Latent Dirichlet Allocation	7
3	Konzept	13
3.1	Daten	13
3.2	Anwendungsfall ZVO	15
4	Implementierung	19
4.1	Topic Modeling Methode	19
4.2	Toolauswahl	19
4.3	Umsetzung Konzept	20
5	Analyse	28
6	Zusammenfassung und Ausblick	37



# 1

## Einleitung

### 1.1 Contributions of this Thesis

### 1.2 Related Work

td-idf word-embeddings

### 1.3 Sturktur dieser Arbeit

Der Bereich Data Science und Datenverarbeitung durchläuft aktuell eine starke Welle an Innovation und Veränderung. Deshalb wird in dieser Arbeit zuerst eine Einleitung die Motivation und das Prinzip der Topic-Modellierung beschreiben. Gefolgt wird die allgemeine Einleitung von der Motivation und Aufgabenstellung in Bezug auf das Anwendungsbeispiel des Zweckverbands Ostholstein (ZVO). Die Anwendung der Algorithmen wird im Laufe der Arbeit mit theoretischen Grundlagen erklärt, die eine klare Notation fordern: diese ist im Teil der Notationen gelistet. Das Prinzip der Topic-Modellierung kann mit unterschiedlichen Ansätzen implementiert werden. Beispiele sind: LDA, NNF und LSA. Da sich diese in ihrem Ansatz und in ihrer Umsetzung unterscheiden, werden im nächsten Abschnitt die bekanntesten dieser Modelle verglichen, indem ihre Vorgehens- und Funktionsweise untersucht werden. Die Methode, die in dieser Arbeit genauer beleuchtet wird, ist Latent Dirichlet Allocation. Der dritte Abschnitt wird von der Herkunft, dem Format und der Verarbeitung der Daten der ZVO handeln. Dabei wird der Anwendungsfall und die ZVO genauer erklärt und ein Einblick in die Daten, deren Reinigung und Verarbeitung gegeben. Darauf folgt das Konzept der Lösungsstrategie, um die Aussagekraft und Qualität der Daten für Topic-Modellierung erfolgreich zu untersuchen. Dafür wird der Ablauf des Algorithmus anschaulich dargestellt und die Prozesskette der Analyse im Detail aufgeschlüsselt. Im vierten Abschnitt findet die Implementierung des im dritten Abschnitt erklärten Konzepts statt. Für diese wird eine Methode der Topic-Modellierung ausgewählt und in einer vorher definierten und begründeten Programmierungsumgebung realisiert. Die Umsetzung wird anhand des gelisteten Algorithmus zeilenweise erklärt. Der Output ist in die Arbeit integriert, um dessen zentrale Rolle in der

Datenanalyse zu unterstreichen und die Ergebnisse verständlich begründen zu können. Die detaillierte Analyse folgt im nächsten Abschnitt, der sich damit befasst, die Ergebnisse der Implementierung im Anwendungsfall der ZVO zu interpretieren. Dabei wird begründet, wie gut sich die Daten für eine Topic-Modellierung der ZVO Daten eignen und was die Ergebnisse über die Qualität der Daten aussagt. Die Arbeit wird abgeschlossen mit der Zusammenfassung der gesamten Analyse und einem Ausblick für die ZVO in Bezug auf Topic-Modellierung. Die gewonnenen Ergebnisse der Datenqualität sollen der ZVO nach der Ergebniserhebung zu einer höheren Effizienz in der Datenverarbeitung verhelfen und Aufschluss über Handlungsbedarf und Optimierungspotential in den Datensätzen geben.

### 1.4 Motivation

Die digitalisierte Welt generiert täglich riesige Mengen an neuen Informationen. Von E-Books, Blogs über Nachrichten-Websites und Magazinen bis hin zu mobilen Anwendungen auf dem Smartphone, immer mehr Menschen verlassen sich auf und richten ihr Leben nach dem Internet aus. Das Zeitalter des Big-Data ermöglicht es Nutzern unbegrenzt viele Daten zu generieren und zu sammeln. Die Kapazitäten, die ein Mensch aufbringen kann, um solche Massen an Daten zu organisieren und zu verstehen, sind schon lange übertroffen. Vor allem durch die steigende digitale Kommunikation und stetig sinkenden Speicherplatzkosten, erhöht sich die Menge an zu speichernden Einsen und Nullen. Laut Statista wurden 2018 33 Zettabyte an Daten generiert mit einer prognostizierten Steigerung bis 2025 um 530% auf 175 Zettabyte. Dieser dramatische Anstieg zeigt die Dringlichkeit für effiziente Algorithmen und Modelle der Datenverarbeitung. Neben der reinen Handhabung solcher Daten steigt aber auch das Bewusstsein, aus diesen Daten Verständnis und Potentiale zu schaffen. Besonders Suchverfahren gewinnen an Bedeutung, wenn in großen, unübersichtlichen Datenmengen bestimmte Informationen gefragt sind. Für die Mehrheit bieten Firmen, wie Google, diese Anwendung an. Zwar kann durch die Keyword-basierten Suche das passende Dokument gefunden werden, jedoch schlägt die Suchmaschine fehl, wenn nach einer Menge von Dokumenten mit einem übergreifenden Thema gefragt ist. Um mehrere Dokumente auf geteilte Themen zu untersuchen, wird die sogenannte Topic-Modellierung verwendet. Topic-Modellierung (dt. Themenmodellierung) beschreibt eine Gruppe von Verfahren, die es ermöglichen, große elektronische Datensammlungen automatisiert zu durchsuchen, organisieren und zu verstehen. Es können Muster innerhalb der Daten entdeckt und Topics extrahiert werden. Ein Topic stellt dabei eine Menge an Wörtern dar, häufig gemeinsam in einem Kontext vorkommen. Einem Topic über Hunde würden beispielsweise die Wörter Bellen, Fell und Halsband zu einer hohen Wahrscheinlichkeit angehören, da sie oft gemeinsam in dem übergeordneten Thema über Hunde gebraucht werden. Die Modellierung solcher Topics wird als Topic-Modellierung bezeichnet, da in einer Datenmenge nach mehreren solcher Topics gesucht wird. Diese können zu unterschiedlichen Anteilen in dem Dokument vertreten sein. Ein Dokument ist dabei ein Teil eines Korpus, also einer Menge mehrerer Dokumente. Dabei stellen Topic-Modelle statistische Modelle dar, die Verwendung in der Inferenz abstrakter Topics in unsortierten Datenmengen finden. To-



pic Modelling gehört zu dem Bereich des Natural Language Processing, also der Verarbeitung natürlicher Sprache. Es verbindet Computerlinguistik, Informatik und Künstliche Intelligenz, um die Potentiale der Sprachverarbeitung mit der heutigen Technik auszuschöpfen. In einer Welt von exponentiell wachsenden Datenmengen finden Methoden der Topic-Modellierung stetig eine breitere Anwendung. Bereits heute wird Topic-Modellierung in vielen Bereichen der Wirtschaft, Wissenschaft und Informationstechnologie verwendet. So findet die Topic-Modellierung unter anderem Anwendung bei Zusammenfassungen, Spam Filtern, Internet of Things (IOT), Healthcare, Blockchain, Chatbots, FAQs oder HR. Dies zeigt wie umfangreich das Anwendungsspektrum der Topic-Modellierung ist. Um semantische Folgerungen aus Datenmengen zu generieren, gibt es verschiedene Ansätze – in dieser Arbeit wird es um das generative Modell Latent Dirichlet Allocation(LDA) gehen. Dabei werden ähnliche Wörter, die in ähnlichen Kontexten vorkommen in Topics gruppiert. Die Grundlagen des LDA liegen bei der Verallgemeinerung des bereits 1999 veröffentlichten Probabilistic Latent semantic Analysis (PLSA). Im Gegensatz zu anderen Machine Learning Methoden im Bereich der Datenverarbeitung, hat Topic-Modellierung die Besonderheit, dass ein Dokument nicht nur zu einem Topic zugeordnet werden kann, wie z.B. bei Clusteralgorithmen. Bei der Topic-Modellierung wird jedes Dokument durch eine Verteilung an Topics beschreiben, das bedeutet in jedem Dokument sind immer alle Topics zu finden - nur zu einem bestimmten Anteil. Genauso sind in einem Topic immer alle Worte zu einer bestimmten Wahrscheinlichkeit vorhanden. Bei einem Artikel, der zu 90% über Sport und 10% über Politik handelt, werden somit neun mal mehr Wörter bezüglich Sport zu finden sein, als über Politik. Topic-Modellierung wird den unüberwachten Lernmethoden des Data Minings zugeordnet, also der Extraktion von Muster und Trends in Datenmengen durch die Anwendung statistischer Algorithmen. Das bedeutet, dass die Topics ohne die Einwirkung von manuell erzeugten Labels gefunden werden. Im Lernprozess werden dann Verteilungen basierend auf den bislang vorgenommenen Zuordnungen iterativ angepasst und verbessert - jedoch alles ohne menschliches Zutun.

### 1.5 Ziel

Diese Arbeit wird die Theorie des Topic-Modellierung anhand des Beispiels des Zweckverband Ostholstein (ZVO) in der Praxis implementieren und die bezüglichen Parameter im Sinne des ZVO bewerten. Der ZVO ist ein Unternehmen, das in Norddeutschland in der Energie-, Entwässerungs-, Internet- und Entsorgungsbranche tätig ist. Es erhält jährlich ein große Menge an Kundenanfrage, die von Mitarbeitern gelesen und klassifiziert wird. Die klassifizierte Kundenanfrage wird dann an die zuständige Abteilung weitergeleitet. Dieser Vorgang ist sowohl zeitintensiv, als auch fehleranfällig und führt zu einer Ineffizienz in der Wertschöpfungskette. Der Prozess soll zukünftig automatisch durch einen Klassifikationsmechanismus funktionieren.

Diese Arbeit verfolgt zwei Ziele:

1. Vorhersage der Qualität der Klassifikation durch Untersuchung der Datenqualität
2. Analyse der vordefinierten Abteilungen durch Topiczuordnung

Diese Arbeit beschäftigt sich mit der Vorhersage der Qualität des Klassifikators, indem die Qualität der manuell erstellten Kategorien und Kundenanfrage-Gruppen untersucht und mit den Ergebnissen verschiedenen Topic Modellierungen verglichen wird. Das Ergebnis eines Topic Models hängt stark von der Qualität der Daten ab, die sie als Input bekommt. Diese Daten durchlaufen eine Reinigungsphase, bevor sie klassifiziert werden, um sie in eine gut zu verarbeitende Form zu bringen. Der Prozess der Reinigung kann Einfluss auf das Ergebnis haben. Ziel dieser Arbeit ist es, Erkenntnisse über die Qualität des Klassifikators zu treffen, in Abhängigkeit zu den verwendeten Daten. Somit wird durch die Nutzung von LDA-Modellen die Qualität der Daten untersucht und Prognosen über einer höheren Qualität der Klassifikation anhand der Daten gemacht.

# 2

## Grundlagen

### 2.1 Notation

- Ein Wort wird mit  $w \in N$  mit  $N$  als Menge aller Worte bezeichnet
- Eine Menge Wörter ergeben ein Dokument  $d \in D$  mit  $D$  als Menge aller Dokumente
- Eine Menge an Dokumenten ergibt einen Korpus  $D$
- Eine Menge an Wörtern ergibt eine Topic  $z \in K$  mit  $K$  als Menge aller Topics
- Ein Topic-Modell wird mit  $M$  beschrieben

### 2.2 Modellvergleich

Das Feld des Information Retrieval bietet vielseitige Verfahren, um Topic-Modellierung zu betreiben. Bei vielen geht es um eine Dimensionsreduktion des Datensatzes, die Aufschluss und Erkenntnis über die Datenmenge geben soll. In dieser Arbeit wird die Latent Dirichlet Allocation(LDA) verwendet, das als Bindeglied zwischen statistischen und probabilistischen Maßen agiert. LDA wurde durch die Optimierung und Vermischung verschiedener vorangegangener Verfahren entwickelt. Die wichtigsten Verfahren in der Entwicklung von LDA sind im Folgenden aufgelistet und erklärt:

#### Term Frequency - Inverse Document Frequency(Tf-idf)

Das Verfahren der (Tf-idf) beschreibt ein statistisches Maß, um die Wichtigkeit von Wörtern in einer Gruppe von Dokumenten zu beurteilen. Die Term-Frequency(Tf) erstellt eine Matrix, die die Anzahl der Wörter auf die Dokumente abbildet, in denen sie enthalten sind. Um zu vermeiden, dass Wörter, die wenig Aussagekraft für den Korpus haben, wie das Wort (und), bestimmt die Inverse Document Frequency(idf) wie spezifisch ein Wort für einen gesamten Korpus ist. Das einzelne Dokument ist dafür demnach nicht mehr wichtig. Die Idee dabei ist, dass Wörter, die sehr selten vorkommen ein Dokument unter Umständen stärker beeinflussen können, als ein Wort, das so häufig vorkommt, dass es schon keine Aussagekraft oder semantische Relevanz mehr hat. Das Verfahren verwendet keine probabilistischen Elemente sondern ist eine rein statistisch Möglichkeit herauszufinden, wie wichtig ein Wort eines Dokuments für das Gesamtthema des Dokuments ist.

## Latent Semantic Analysis

Viele Verfahren der Topic-Modellierung gehen auf das sogenannte Latent Semantic Analysis (LSA) zurück, das zuerst 1990 erwähnt wurde. LSA baut auf dem Prinzip der Tf-idf auf, indem es die Tf-Matrix nutzt und adaptiert. Bei dem Verfahren geht es um das Finden von sogenannten Hauptkomponenten in Dokumenten. Grob können Hauptkomponenten als allgemeine Überthemen verstanden werden. So wäre Politik die Hauptkomponente, die von Wörtern wie Politiker, Bundestag und Regierung beschrieben wird. Mathematisch gesehen verbindet man mit der Hauptkomponente die Hauptkomponentenanalyse, die als statistische Methode versucht, viele Variablen mithilfe von Eigenvektoren und Kovarianzen in einer vereinfachten Form mit einem möglichst geringen Informationsverlust darzustellen. Dabei ist das Ziel die Tf-Matrix möglichst verkleinert darzustellen, die die Anzahl der Vorkommen für jedes Wort pro Dokument abbildet. Sie wird in Matrizen aus Eigenvektoren und Eigenwerten aufgespalten, wodurch es möglich ist, die wichtigen Hauptkomponenten aus der meist sehr großen Tf-matrix zu extrahieren. Diese Matrix wird als Produkt von drei Matrizen dargestellt, von denen die mittlere eine Diagonalmatrix darstellt. Die Werte auf der Diagonalen lassen daraus die Topics der Textmenge ablesen. LSA stellt sich als ein attraktives Verfahren heraus, da es Synonyme besser erkennen kann, als LDA und wird heutzutage unter anderem intensiv in dem Bereich des Digital Marketings genutzt.

## Probabilistic Latent Semantic Analysis

Im Gegensatz zu LSA, was Singulärwertzerlegung für die Topic-Modellierung nutzt, verfolgt Probabilistic Latent Semantic Analysis (PLSA) einen anderen Ansatz. Die Kernidee ist, ein Wahrscheinlichkeitsmodell zu finden, das ein Dokument erzeugt, dass zu der Tf Matrix passt. PLSA hat als erstes Modell dem Konzept der Topic-Modellierung Wahrscheinlichkeiten hinzugefügt. Dabei sind Das Dokument und Wort bekannt, das assoziierte Topic jedoch nicht. Der grundlegende Gedanke ist, dass ein Dokument generiert werden kann, indem zuerst das Dokument mit einer bestimmten Wahrscheinlichkeit gewählt wird, dann ein Topic und aus diesem Topic dann ein Wort - alles mit Wahrscheinlichkeiten beschrieben. Es lässt sich der Fall, dass Dokument  $d$  über Topic  $z$  handelt, mit einer Wahrscheinlichkeit von  $P(z | d)$  beschreiben. Dazu berechnet  $P(w | z)$ , wie wahrscheinlich es ist, dass das Wort  $w$  aus einem gegebenen Dokument  $d$  zu Topic  $z$  gehört. Somit führt PLSA die Gesamtwahrscheinlichkeit ein, dass ein Wort in einem gegebenen Dokument zu finden ist:  $P(D, W) = P(D) \sum_z P(Z | D)P(W | Z)$ . Die Gleichung verbindet die Wahrscheinlichkeit ein bestimmtes Dokument zu erhalten mit der eines Wortes in diesem bedingt durch die Topic-Verteilung in diesem Dokument.

## Latent Dirichlet Allocation

Basierend auf LSA und PLSA ergibt sich eine der meist verbreiteten Verfahren, das Latent Dirichlet Allocation (LDA). Dieses Verfahren ist sehr ähnlich zu PLSA, das durch zwei Dirichlet-Verteilungen ergänzt wurde. LDA liegt, genau wie PLSA, ein generierender Prozess zugrunde, der jedes Dokument basierend auf mehreren Wahrscheinlichkeitsverteilungen generieren kann. Dabei wird zuerst das Dokument  $d$  mit  $P(d)$  gewählt, für

dieses dann eine Topic  $z$  mit  $P(z | d)$  und anschließend die Wörter  $w$  mit  $P(w | z)$ . Grundsätzlich werden Dokumente bei LDA als die Dokument-Topic-Verteilung dargestellt, die die Ausprägungen verschiedener Topics in einem Dokument beschreibt. Für die Topics gibt es die Topic-Word-Verteilung, die die Wahrscheinlichkeit beschreibt, dass ein bestimmtes Wort in einer gewissen Regularität in einem Themenbereich vorkommt. Dabei geht man davon aus, dass ein Dokument eine Verteilung von Topics ist, während ein Topic als eine Verteilung über Wörter betrachtet wird. Der Unterschied zu PLSA liegt in zwei hinzugefügten Dirichlet Verteilungen, die die Zuordnung von Dokumenten zu Topics maßgeblich beeinflussen. Die Wahrscheinlichkeit, dass ein bestimmtes Dokument generiert wird, ist das Produkt der Wahrscheinlichkeiten der Dokument-Topic-Verteilung und Topic-Word-Verteilung mit den Wahrscheinlichkeiten der zwei hinzugefügten multinomialen Verteilungen. So werden erst die Topics zufällig ausgewählt, wie in der Dirichlet-Verteilung definiert, und aus diesen dann, mithilfe der zweiten Dirichlet-Verteilung, Wörter aus diesen Topics hergeleitet, wodurch das Enddokument entsteht. Die Herangehensweise ist in der Realität meistens umgekehrt, da selten Dokumente auf Basis bekannter Topics generiert werden müssen, sondern eher Topics von bekannten Dokumenten gefunden werden sollen.

### 2.3 Grundlagen der Latent Dirichlet Allocation

Latent Dirichlet Allocation ist ein grundlegendes und bekanntes Verfahren aus der maschinellen Verarbeitung natürlicher Sprachen. Begründet wird dies unter anderem auf der Komplexität der damals bestehenden Techniken der Textverarbeitung. So waren Clusteralgorithmen zu starr in ihrem Anwendungsumfeld, während Dimensionsreduktionen, wie die Hauptkomponentenanalyse von LSA Ergebnisse lieferte, die sehr schwer zu interpretieren waren. Das Prinzip der Topic-Modellierung basiert auf einer Menge an Dokumenten, die den Korpus darstellen. Dabei werden bei LDA alle Dokumente als Menge von Wörtern angenommen. Diese sind als sogenannter Bag of Words modelliert. Das bedeutet, dass die Wörter als Menge ohne Reihenfolge interpretiert werden und nur die Häufigkeit der Begriffe in dieser Menge relevant ist. LDA ist ein generatives Modell, das durch unterschiedliche Einflussgrößen, wie in Abbildung 2.1 dargestellt, Dokumente erstellen kann.

Bezüglich der Namensgebung, steht Latent für alles, was wir im Vorhinein nicht kennen. Im Fall LDA handelt es sich um die Themen, die in einem Dokument zu einem bestimmten Teil vertreten sind. Dirichlet beschreibt eine Verteilung von Verteilungen. Dies ist vergleichbar mit einem Würfel, bei dem regulierbar ist, wie gleichmäßig die Zahlen gewürfelt werden. Dabei ist der Würfel eine Verteilung und die Aufteilung der Gleichmäßigkeit auch. Beim Topic Modeling bedeutet Dirichlet eine Verteilung von Topics in Dokumenten und eine Verteilung von Wörtern in Topics. Die Dirichlet-Verteilungen sind die ausschlaggebende Veränderung im Vergleich zu PLSA. Durch sie kann die Stärke der Ausprägung der Topics in der Dokument-Topic-Verteilung und der Wörter in der Topic-Word-Verteilung bestimmt werden. Die Allocation weist mithilfe der errechneten Dirichlet-Verteilungen Topics Wörtern und Dokumenten Topics zu. Eine Besonderheit bei

der Topic-Modellierung mit LDA ist, dass die Anzahl der gesuchten Themen  $K$  vorgegeben werden muss. Oft ist diese vorher jedoch nicht bekannt und muss über Hilfsverfahren, wie der Perplexitätsberechnung ermittelt werden. Dabei wird gemessen, wie geeignet die das Modell für die Verarbeitung der gegebenen Daten ist. Die Funktionsweise von LDA ist über folgende graphische Abbildung beschrieben:

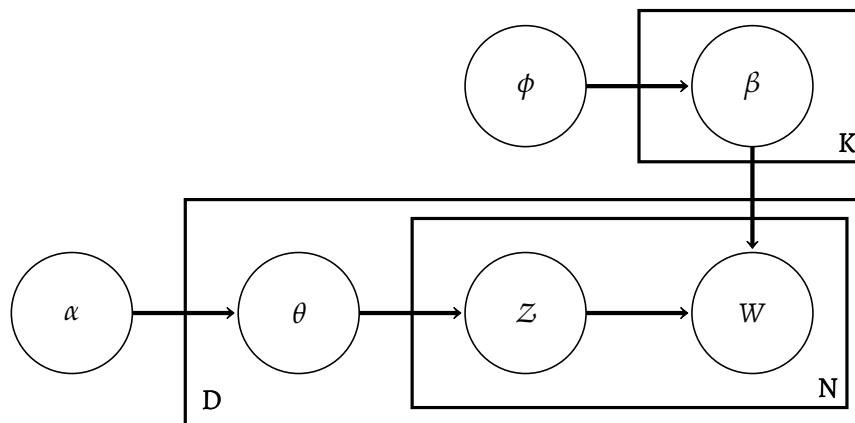


Abbildung 2.1: Graphische Darstellung von LDA

Dabei beschreibt  $W$  als einzige nicht verborgene Variable eines von  $N$  Wörtern des Dokuments. Das Wort ist semantisch einem Thema  $Z$  zugeordnet. Das Thema wiederum hängt von der Themen-Verteilung  $\theta$  des Dokuments ab, das als ein Element der  $|M|$  vorliegenden Dokumente betrachtet wird. Bei LDA werden zwei Verteilungen aus den Dokumenten  $d \in D$  und  $k \in K$  gelernt:

- die Dokument-Topic-Verteilung  $\theta_d$ , die angibt, mit welcher Wahrscheinlichkeit das Dokument zu jedem Themen gehört und
- die Topic-Wort-Verteilung  $\phi_k$ , die die Wahrscheinlichkeit berechnet, dass ein Wort einem Thema angehört.

$M = \text{LDA}(D)$  beschreibt ein LDA Modell, das auf dem Korpus  $D$  trainiert wurde. Das Modell und dessen Verteilungen kann durch die Parameter  $\alpha$  und  $\beta$  angepasst werden.  $\alpha$  bestimmt die Intensität der Dokument-Themen-Verteilung, während  $\beta$  die der Topic-Wort-Verteilung beeinflusst. Bei einem großen  $\alpha$  ist die Verteilung der Topics in einem Dokument ähnlicher. Bei LDA werden zwei widersprüchliche Bedingungen verfolgt, die von den beiden Parametern beeinflusst werden können:

1. Erstens strebt man für alle Wort eines bestimmten Dokuments so wenig zugeordnete Themen an, wie möglich.
2. Zweitens soll ein Thema über so wenig relevante Wörter wie möglich verfügen.

Die beiden Ziele stehen in einer Wechselbeziehung zueinander, da eine minimale Anzahl an vertretenen Topics in einem Dokument zu maximal vielen Wörtern in diesen Topics führt. Die minimale Anzahl an Topics wäre erreicht, wenn man alle Wörter eines Dokuments einem Thema zuweist. Dadurch verfügt das Topic jedoch über alle Wörter



des Dokuments.  $\alpha$  befindet sich in dem Bereich  $[0, 1]$  mit sinnvollen Werten zwischen  $[0.01, 0.1]$ , während  $\beta = 0.01$  durchschnittlich die besten Ergebnisse liefert. Große Werte führen zu einer Gleichverteilung, die wiederum eine Verschlechterung der Perplexität bedeutet. Somit bietet die Perplexität ein Mittel, um  $\alpha$  und  $\beta$  optimal für die individuelle Anwendung zu finden.

### Der generative Prozess

Bei der Erstellung eines Klassifikators gibt es zwei unterschiedliche Herangehensweisen: den deskriptiven und den generativen Ansatz. Bei der deskriptiven, oder auch beschreibenden Statistik geht es um die sinnvolle und übersichtliche Darstellung empirischer Daten durch zum Beispiel Tabellen oder Kennzahlen. Betrachtet man zwei Variablen, ein bekanntes  $X$  und eine gesuchte Variable  $Y$ , dann wird im deskriptiven Modell die bedingte Wahrscheinlichkeit  $P(Y | X)$  betrachtet. Im Gegensatz dazu ist bei generativen Modellen die Wahrscheinlichkeit von  $X$  und  $Y$  gemeinsam relevant, also  $P(X \cap Y)$ . Bei generativen Modellen besteht die Möglichkeit, Ausgabeinstanzen zu erstellen. Bei LDA handelt es sich um einen generativen Algorithmus. Das bedeutet nicht direkt, dass LDA bei der Ausführung aktiv neue Dokumente entwickelt, sondern stellt eine Erklärung dar, wie die Zusammenstellung der Topics und deren Wörter in einem spezifischen Fall entstanden sind. Es könnten theoretisch im Fall einer neuen Dokumentzuordnung zu Abteilungen für jede Abteilung zufällig Dokumente generiert werden. Diese würden zufällig gewählte Wörter basierend auf der Topic-Verteilung aneinanderreihen und mit hoher Wahrscheinlichkeit keinen Sinn ergeben. Trotzdem könnten diese Dokumente mit den Dokumenten der Abteilungen verglichen und auf Ähnlichkeit werden. Dann könnten die Parameter solange angepasst werden, bis das Ergebnis akzeptabel ist. Der reguläre Weg ist jedoch andersherum: Wir suchen nicht das Dokument zu einer gegebenen Dokument-Topic-Verteilung und einer Topic-Word-Verteilung, sondern haben das Dokument bereits und suchen die Dokument-Topic-Verteilung. Es wird ein Modell gelernt, dass den gegebenen Korpus am wahrscheinlichsten nach folgender Vorgehensweise generiert haben könnte. Dabei sei  $Dir(\alpha)$  eine Dirichletverteilung, was einen Parameter  $\alpha$  bzw.  $\beta$  übergeben bekommt und  $Multinom(\theta_d)$  eine Multinomialverteilung mit dem Parameter  $\theta$  bzw.  $\phi$ . Bei der Berechnung ergibt sich der Output durch die dann berechneten Wahrscheinlichkeitsverteilungen  $\theta_d$  und  $\phi_k$ .

1. Wähle ein  $\theta_d$  als  $Dir(\alpha)$
2. Wähle ein  $\phi_k$  als  $Dir(\beta)$
3. Für jedes Wort  $w$  and Stelle  $i = 1, \dots, N$  im Dokument  $d$ :
  - 3.1 Wähle ein Thema  $z_{d,i}$  als  $Multinom(\theta_d)$
  - 3.2 Wähle ein Wort  $w_{d,i}$  als  $Multinom(\phi_{z_{d,i}})$

Für das Modell kann die Gesamtwahrscheinlichkeit berechnet werden. Diese ergibt sich aus dem Produkt aus:

- der Wahrscheinlichkeit der Dokument-Topic-Verteilung zusammen mit der Dirichlet-Verteilung  $\alpha$  für jedes Dokument

- der Wahrscheinlichkeit der Topic-Word-Verteilung zusammen mit der Dirichlet-Verteilungen  $\beta$  für alle Topics
- der Wahrscheinlichkeit eines Topics zusammen mit der Dokument-Topic-Verteilung für alle Wörter in allen Dokumenten
- der Wahrscheinlichkeit eines Wortes zusammen mit der Topic-Word-Verteilung für alle Wörter in allen Dokumenten

$$P(w, z, \theta, \phi, \alpha, \beta) = \prod_{i=1}^K P(\phi_i, \beta) \cdot \prod_{j=1}^D P(\theta_j, \alpha) \cdot \prod_{r=1}^N P(z_{j,r} | \theta_j) \cdot P(w_{j,r} | \phi_{j,r}) \quad (2.1)$$

Abbildung 2.2: Die totale Wahrscheinlichkeit eines LDA Modells  $M$

Die Schwierigkeit des Algorithmus besteht in der Berechnung der  $\theta$ -Verteilung und entsprechenden Topics der gegebenen Dokumente für latente Variablen. Das heißt bekannt sind das Wort und die Dirichlet-Verteilungen  $\alpha$  und  $\beta$ . Auf Basis dieser gegebenen Informationen wird dann die Wahrscheinlichkeit für eine Topic errechnet, der das Wort angehört. Dazu wird die Dokument-Topic-Verteilung gesucht, die von den Dirichlet-Verteilungen beeinflusst wird. Dies lässt sich durch folgende Wahrscheinlichkeitsverteilung ausdrücken, ist jedoch nur approximierbar und nicht exakt zu bestimmen:

$$P(\theta, z | w, \alpha, \beta) = \frac{P(\theta, z, w | \alpha, \beta)}{P(w | \alpha, \beta)} \quad (2.2)$$

Abbildung 2.3: Die bedingte Wahrscheinlichkeit für die Topic und Dokument-Topic-Verteilung gegeben einem Wort und den Dirichlet-Verteilungen  $\alpha$  und  $\beta$

### Alpha und Beta

Die Dirichlet Verteilungen werden durch die beiden Parameter  $\alpha$  und  $\beta$  bestimmt. Diese formulieren die mathematische Bedeutung der beiden Ziele von LDA, die während der Ausführung versucht werden zu verfolgen:

1. Ein Dokument wird so wenigen Themen wie möglich zugewiesen ( $\alpha$ )
2. Jedes Thema hat so wenig relevante Wörter wie möglich ( $\beta$ )

Dabei kann 1. erreicht werden, wenn alle Worte eine Topic wären, was jedoch nicht mit 2. übereinstimmen würde. Für einen erfüllten 2. Satz gibt es nicht die minimale Anzahl an Topics. Die Funktionsweise von verschiedenen  $\alpha$ -Werten zeigen folgende Abbildungen:



## 2 Grundlagen

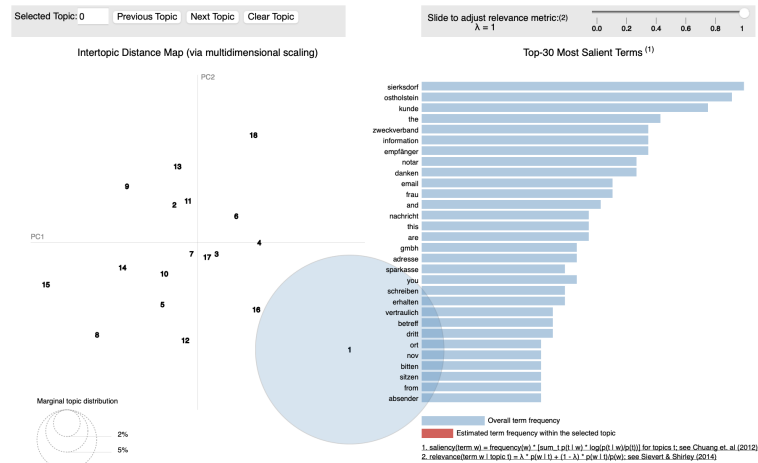


Abbildung 2.4: Je kleiner der  $\alpha$ -Wert wird, desto eindeutiger dominiert ein Topic das Dokument.

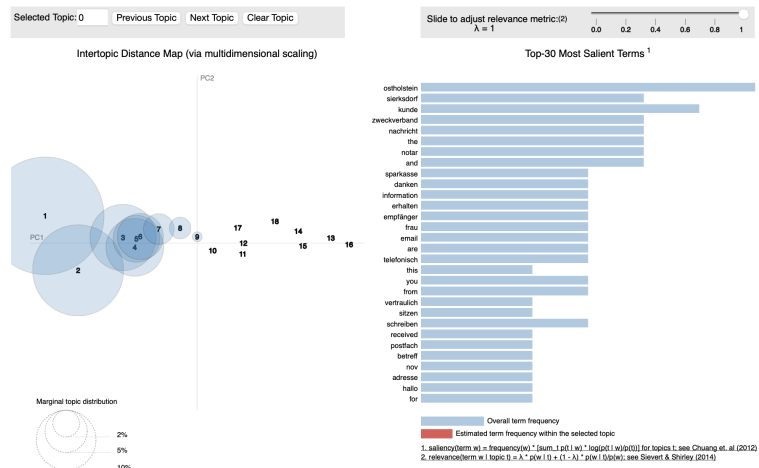


Abbildung 2.5: Je größer der  $\alpha$ -Wert wird, desto gleichmäßiger sind die Topics in dem Dokument beteiligt.

In der ersten Abbildung 2.4 führt ein kleiner  $\alpha = 0.01$  Wert zu einer sehr eindeutigen Dokument-Topic-Verteilung. Das erkennt man daran, dass es nur einen sehr großen blauen Kreis gibt. Jeder Kreis repräsentiert eine Topic und mit der Größe dessen Anteil am Dokument. Befinden sich zwei Kreise zum Teil auf der gleichen Position bedeutet das, dass sich die Wörter in den Topics zu einem gewissen Teil überschneiden. In Abbildung 2.4 sind bei den meisten Topics nur die Zahlen und keine Kreise zu erkennen, da die Topics einen sehr kleinen Anteil an dem Dokument haben. Dies ist durch  $\alpha$  bedingt. Bei der zweiten Abbildung 2.5 hingegen haben wir ein  $\alpha = 1$ ,

was eine gleichmäßigere Verteilung zur Folge hat. So sieht man, dass es viele Kreise gibt, die sich zum Teil auch überlappen. Die Größe der Kreise unterscheidet sich nicht so stark voneinander, wie bei Abbildung 2.4, somit sind die Topics bezüglich ihren Anteils an dem Dokument ähnlicher zueinander. Der Trade-off zwischen den beiden Zielen ist der Grund für das Funktionieren von LDA, denn dadurch werden Gruppen mit semantisch eng miteinander verbundenen Wörtern, die oft miteinander vorkommen, gefunden. Nach dem gleichen Prinzip funktioniert die Dirichlet-Verteilung der Topic-Word-Verteilung  $\beta$ .

# 3

## Konzept

Dieser Abschnitt wird die Vorgehensweise beschreiben, wie unter der Nutzung der Daten der ZVO Topic-Modellierung umgesetzt wird, um Aufschluss über die Qualität der Daten zu erlangen. Dafür werden zuerst die Daten, deren Format und deren Verarbeitung vorgestellt und dann die Herangehensweise geschildert und aufgeschlüsselt.

### 3.1 Daten

Die ZVO hat 133045 unterschiedlich große Dokumente zur Verfügung gestellt. Die Daten sind in einem Dataframe gespeichert und liegen in folgendem Format vor:

	filename	subject-message	Abt0	Abt1	Abt2	Abt3	...	Abt15	Abt16	Abt17
0	FILE0	content0	0	0	0	1	...	0	0	0
1	FILE1	content1	0	0	0	0	...	1	0	0
2	FILE2	content2	1	0	0	0	...	0	0	0
3	FILE3	content3	0	0	0	1	...	0	1	0
4	FILE4	content4	0	1	0	0	...	0	0	1
...	...	...	...	...	...	...	...	...	...	...
133044	FILE133044	content133044	0	0	1	0	...	0	0	0

Abbildung 3.1: Daten TODO...

Relevant für die Auswertung sind die subject-message, die die eigentliche Anfrage enthält, und die jeweilige Abteilung, der sie von der ZVO händisch zugeordnet wurde. Die Tabelle verfügt über eine Matrix mit 18 Abteilungen, von denen pro subject-message eine oder mehrere mit einer 1 versehen ist bzw. sind. Dies beschreibt die Abteilung bzw. Abteilungen, der bzw. denen diese Anfrage manuell zugeordnet wurde. Die Daten in subject-message sind bereits bereinigt, also liegen wie in Abbildung

## 3.2 vor:

## OUTPUT:

---

```
'wasser verbraucht amt deutschland ablesung zaehlen strom voll ort luebeck
  art straße messung verband nummer platz markieren wechsel lieferant
  stelle verbrauch kunde kunden anrede mann sommer beschwerde schrift
  allgemein kommunikation datenmanagement fern'
```

---

Abbildung 3.2: So könnte ein Datum der ZVO aussehen

Um die Einträge in eine computer-lesbare Form zu verwandeln, muss ein Dictionary erstellt werden, dass alle Wörter auf eine Anzahl ihrer Vorkommen abbildet. Dafür müssen die Wörter als alleinige Listeneinträge einlesbar sein, wie in Abbildung 3.3:

## OUTPUT split:

---

```
['wasser', 'verbraucht', 'amt', 'deutschland', 'ablesung', 'zaehlen',
 'strom', 'voll', 'ort', 'luebeck', 'art', 'straße', 'messung', 'verband',
 'nummer', 'platz', 'markieren', 'wechsel', 'lieferant', 'stelle',
 'verbrauch', 'kunde', 'kunden', 'anrede', 'mann', 'sommer', 'beschwerde',
 'schrift', 'allgemein', 'kommunikation', 'datenmanagement', 'fern']
```

---

Abbildung 3.3: Ein gesplittetes Datum vorbereitet für das Dictionary

## Datenreinigung

Bevor die Topic-Modellierung auf den Daten durchgeführt werden kann, müssen die Daten einem Prozess unterzogen werden. Dieser beginnt mit der Datenakquise, also der Sammlung bestimmter relevanter Daten. Im Falle der ZVO bedeutet dies, dass es genügend Kundenanfragen gibt, die verarbeitet werden können. Wenn diese Daten bestehen, werden sie auf die relevanten Wörter reduziert, aus denen eine bedeutsame Inferenz von Informationen möglich ist, sodass unter anderem die sogeanannten Stop-Words, also eine Menge von Verbindungswörtern entfernt werden. Es gibt vordefinierte Sammlungen von Stop-Words und es können eigene definiert werden. Das Entfernen von Stop-Words verhindert, dass sich das Topic-Modell an den falschen Wörtern aufhält, die keine aussagekräftige semantische Information in sich tragen. Nachdem die Stop-Words entfernt wurden, ergeben die meisten Sätze keinen inhaltlichen Sinn mehr, wie in Abbildung 3.2 zu erkennen. Dies ist jedoch nicht schlimm, da der nächste Schritt der Reinigung den Daten die Reihenfolge entzieht und das Bag of Words Modell erstellt. Dabei werden alle Wörter als eine Menge von unabhängigen Wörtern betrachtet, die sich nur durch ihre Vorkommensanzahl in dem Korpus unterscheiden. Dabei wird die Groß- und Kleinschreibung ebenfalls irrelevant, da es bei einer Menge von Wörtern ohne Reihenfolge auch keine Satzanfänge

mehr gibt. Des weiteren besteht die Möglichkeit alle konjugierten bzw. deklinierten Formen eines Wortes zum Infinitiv bzw. Nominativ zusammenzufassen. Dies wird als Lemmatisierung bezeichnet und wurde bei der ZVO nicht angewendet.

#### Feature Engineering

Wenn die Daten in der gewünschten Form vorliegen, beginnt der Schritt des Feature-engineerings. Für einen Computer sind Wörter nicht so leicht zu verarbeiten, wie Zahlen, weshalb in diesem Schritt eine Quantisierung der Wörter und Überführung dieser in eine zahlenbasierte Form vorgenommen wird. Dabei sind die Features in Form der Spalten des Dataframes bereits passend gegliedert. Für die Darstellung der Dokumente als Menge an Zahlenwerten wird ein Dictionary erstellt, das die Wörter wie in Abbildung 3.3 erhält und die Anzahl der Wörter ohne Duplikate in einer Liste, das Dictionary, auflistet. Diese Auflistung der Wörter zusammen mit ihrer Anzahl und laufenden Indexnummer kann als Input für ein LDA Modell verwendet werden. Die Worte werden intern als ID dargestellt, die für einen Menschen nicht zu verstehen ist. Aus diese Grund wird im Algorithmus der `id2wordParameter` eingeführt, der die Übersetzung zwischen Zahl und Datum darstellt.

## 3.2 Anwendungsfall ZVO

Bei der ZVO sollen jährlich händisch aufgenommene Anfragen in Zukunft maschinell klassifiziert werden. Dafür wird im Rahmen des M4KK ein Kundenanfrage-Klassifikator erstellt. Für die korrekte Klassifikation ist die Qualität der vorliegenden Daten von hoher Bedeutung. Die Arbeit verfolgt das Ziel, die Qualität der Daten zu analysieren und semantische Gruppierungen zu analysieren. Dafür wird ein LDA Modell generiert, das als Datengrundlage alle verfügbaren ZVO Daten verwendet. Das Vorgehen ist wie folgt in Abbildung 3.4 dargestellt:

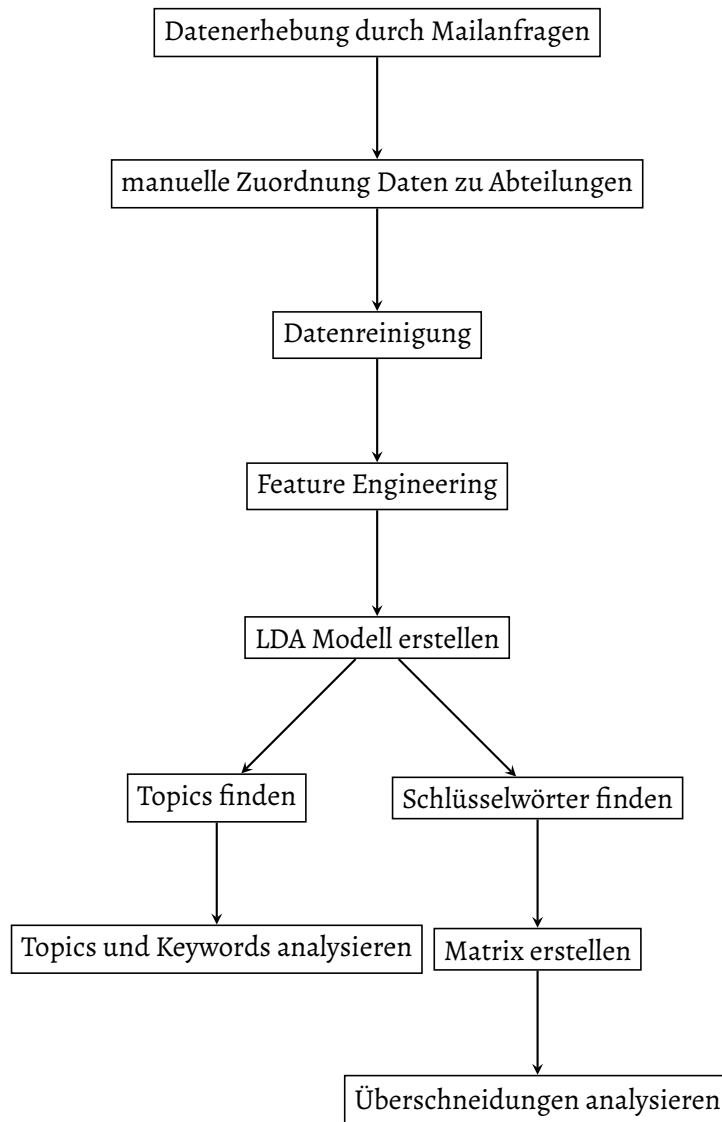


Abbildung 3.4: Prozessschritte von der Datenerhebung bis zur Analyse. Die Arbeit verfolgt zwei Ziele: Die Topicanalyse und Zuordnung von Topic und Label.

Nach der Erhebung der Daten können die einzelnen Anfragen manuell in die vorgegebenen Abteilungsgruppen eingeteilt werden. Dort wird die Datenreinigung, wie in Abschnitt 3.2 beschrieben vorgenommen. Sind die Daten bereinigt, kann das Feature Engineering beginnen, wonach die Daten für den Computer verständlich formatiert sind. Die Erstellung des LDA Modells wird vorgenommen, sobald der Korpus erstellt ist. Die Modellerstellung ist durch den in Abbildung 3.5 gegebenen Pseudocode beschrieben.

Ist das Modell generiert und die Topics auslesbar, können die Daten evaluiert werden. Dazu werden zwei Ansätze verfolgt. Zum Einen werden Topics und zugehörigen

```

data ←
for d in Anzahl Dokumente do
    data ← data + str(d)
end for
Teile data in einzelne items einer Liste auf
Erstelle ein Dictionary aus der Liste
Wandle ID aus Dictionary in Wörter um
Erstelle den Korpus
Erstelle das Modell
Gibt die Topic-Wort-Verteilungen für alle Topics aus

```

### ABBILDUNG 3.5

Pseudoalgorithmus für die Prozesskette

Schlüsselwörter untersucht, um u.a. Aufschluss über mögliche Verbesserungspotentiale bei der Datenreinigung oder dem Feature Engineering festzustellen. Als zweites werden die vom Modell erfassten Gruppen betrachtet und den gegebenen manuell klassifizierten Abteilungen zugeordnet. Die folgenden Punkte beschreiben den Prozess im Detail:

#### 1. Gruppen und Wörter finden

Ein LDA Modell besteht aus zwei Verteilungen, die die zugrundeliegenden Daten semantisch darstellbar machen: die Dokument-Topic-Verteilung und die Topic-Wort-Verteilung. Als Ausgabe des Modells ist also zu erkennen, welche Topics die Dokumentmenge durchschnittlich hauptsächlich beschreiben und welche Wörter in den Topics jeweils dominant vorkommen. Das Modell kann Topics nicht inhaltlich benennen, sondern nur die Verteilungen darstellen. Somit ist nicht eindeutig, welches Topic welche Abteilung der ZVO darstellt. Dafür betrachten wir die Topic-Wort-Verteilungen und schließen von dieser auf die Qualität der Daten. Ist über die dominanten Wörter in einem Topic zu erkennen, welche Abteilung dieser repräsentiert, scheint das Modell und die Daten gut genug zu sein, um die Daten zu klassifizieren. Sollte die Abteilung nicht an den Wörtern ablesbar sein, sind die Daten nicht optimal für eine Klassifikation geeignet. Dabei ist auch relevant, wie oft die gleichen Wörter in den jeweiligen Topics vorkommen und ob Stop-Words Manipulation eine Option der Qualitätssteigerung darstellt.

#### 2. Zuordnung Abteilung zu Topic

Das LDA Modell erstellt aus den Daten 18 Topics. Diese Topics sollten im optimalen Fall sehr ähnlich zu den händisch klassifizierten Abteilungen sein, wenn die vordefinierten Abteilungen eine akkurate Repräsentation der Topics in den Daten darstelle. Ist dies nicht der Fall, kann man auf eine schlechte Klassifikation schließen. Dies kann durch eine schlechte Qualität der Daten als auch durch eine ineffiziente manuelle Einteilung der Topics bedingt sein. Die Fähigkeit, Topics auf Abteilungen zu mappen, gibt Aufschluss über die Qualität der Daten. Für die Zuordnung werden zwei Matrizen verwendet: gruppen\_LABEL

und `gruppen_LDA`. Die erste sortiert alle Dokumentindizes als Listenelemente in die jeweilige Zeile der Matrix, sodass der Index eines händisch in Abteilung 3 eingeordneten Dokumentes in `gruppen_LABEL[4]` zu finden ist. Die Matrix `gruppen_LDA` beinhaltet alle Indizes der Dokumente, die vom Modell klassifiziert wurden, in gleicher Struktur. Dafür wird für jedes Dokument, das den Korpus ausmacht, eine dokumentseigene Dokument-Topic-Verteilung errechnet. Das Topic, für das das Dokument am wahrscheinlichsten ist, bestimmt, welcher Teilliste der Dokumentindex angehängt wird. Beide Matrizen verfügen nun über die Indizes der Dokumente in den jeweiligen Topics bzw. Abteilungen und können anhand der einzigartigen Indizes auf Überschneidungen geprüft werden. Die Anzahl der Überschneidungen werden in einer Matrix gespeichert, die jedes Element von `gruppen_LDA` auf jedes Element von `gruppen_LABEL` abbildet und deren Überschneidung zählt. Eine optimale Zuordnung von Topic auf Abteilung ist möglich, wenn jede Zeile ein Maximum in einer Spalte hat, die nicht auch das Maximum einer anderen Zeile enthält. Ist dies jedoch nicht der Fall, sind die Daten in der aktuellen Form nicht optimal für die Klassifizierung.



# 4

## Implementierung

Dieser Abschnitt befasst sich mit der Implementierung des im Abschnitt 3 beschriebenen Konzepts. Dabei wird die Implementierung durch Python Code erklärt und die genaue Vorgehensweise dargestellt. Die Ergebnisse werden gezeigt, jedoch noch nicht analysiert.

### 4.1 Topic Modeling Methode

Zur Untersuchung der Qualität der ZVO-Daten wird in dieser Arbeit die LDA Methode verwendet. Dabei wird nur der Text als bekannt angenommen. Die Topics sind zu Beginn nicht bekannt. Als grundlegendes Topic Modellierungsverfahren findet es Verwendung in einem breiten Anwendungsspektrum. Durch die Bekanntheit von LDA sind bereits viele Pakete und Bibliotheken in Programmierumgebungen vorzufinden und einfach zu implementieren. Seit LDAs Veröffentlichung in 2000 wurde eine umfassende und detailreiche Dokumentation entwickelt, die neben vielen Forenbeiträgen, die Arbeit mit LDA stark erleichtern. Zusätzlich hat LDA in diesem Anwendungsfall den Vorteil, dass es nicht wie zum Beispiel LSA direkt Dokumentähnlichkeiten ausgibt, sondern das Ergebnis in Form einer Matrix darstellt, die Wörter auf Dokumente abbildet. Damit ergeben sich als Werte der Matrix die Topics, denen die Wörter jeweils angehören. Bei LDA Modellen ist die Anzahl der Topics ein individueller Input, durch den sich das Ergebnis schwerwiegend verändern kann. Die Werte in der Matrix können somit von 0 bis zu der individuellen Anzahl der Topics reichen. Die optimale Anzahl an Topics ist grundsätzlich ein nicht einfaches Problem bei Anwendungen. Im Fall der ZVO werden als Anzahl der Topics 18 gewählt, da dies die Anzahl der bereits erstellten Abteilungen ist.

### 4.2 Toolauswahl

Bekannte Frameworks zur Topic-Modellierung implementieren grundsätzlich ähnliche Algorithmen. Betrachtet wurden in dieser Arbeit Mallet, Gensim und Sci-kit

Learn. Wichtige Schritte sind die Vorbereitung der Daten, die Implementierung, die Auswertung und die Visualisierung. Als Bibliothek wird in dieser Arbeit Gensim verwendet, die für die Verarbeitung von unstrukturierten Daten und Anwendung von unüberwachten Algorithmen bekannt ist. Algorithmen, wie word2vec, LSI oder LDA entdecken automatisch Strukturen durch das Prüfen von gemeinsam auftretenden Mustern im Korpus der Trainingsdaten. Viele bekannte Algorithmen sind bereits in diesem Framework implementiert, was die Umsetzung der Topic-Modellierung vereinfacht. Gensim erlangte in der Vergangenheit Bekanntheit durch seine hochoptimierten Implementierungen bekannter Algorithmen und der Schnelligkeit und Verlässlichkeit, mit der diese ausgeführt werden. Außerdem wird Gensim in Python verfasst, was sich sehr gut für Probleme im Bereich des Data Science eignet.

### 4.3 Umsetzung Konzept

Wie in Abschnitt 3.2 beschrieben verfolgt die Arbeit zwei Ansätze. Im Folgenden wird die Umsetzung dieser dargelegt:

1. Alle Dokumente werden für die Erstellung eines Korpus genutzt. Der Korpus generiert eine Topic-Verteilung für die Gesamtheit aller Dokumente. Dabei werden zuerst alle Anfragedaten in einen String zusammengefügt, der als Grundlage für das Wörterbuch und den Korpus dient. Um diesen in ein Dictionary, also eine nummerierte Auflistung aller Wörter und dessen Anzahl, zu verwandeln, muss der String in eine Liste mit voneinander getrennten Items gesplittet werden. Hier wird ein Bag of Words Prinzip verfolgt, die Reihenfolge ist somit irrelevant für das Ergebnis des Modells. Aus der Liste wird dann das Dictionary erstellt. Durch den Aufruf des LDA Modells wird aus dem Bag of Words mithilfe des Dictionary eine vorgegebene Anzahl an Themen aus der Wortenge modelliert, basierend auf häufig zusammen auftretenden Wörtern. Dadurch ergibt sich neben einer Verteilung der Topics in einem Modell die Verteilung der Wörter, die ein Topic besonders beeinflussen. Ist das Modell generiert, können die Topics ausgelesen werden. Der Programmcode für die beschriebenen Schritte ist in Abbildung 4.1 dargestellt.

---

```

INPUT:
data = ''

# Zusammenfügen aller Dokumente in einen großen String
for x in range(0,106000):
    data += df.at[x, 'subject-message']

# Aufteilen des Strings in viele einzelne Wörter
list = data.split()

# Erstellen eines Dictionarys
dictionary = corpora.Dictionary([list])
temp = dictionary[0]
id2word = dictionary.id2token

# Erstellen des Korpus
corpus = [dictionary.doc2bow(text) for text in [list]]

# Erstellen des LDA Modells
lda = LdaModel(corpus, num_topics=18, id2word = id2word)

#Ausgeben aller Topics inklusive Schlüsselwörtern
pprint(lda.print_topics())

```

---

Abbildung 4.1: Der Input zum Aufbau eines LDA Modells

Der Input-Code generiert aus allen Daten ein Korpus, der 18 Gruppen in den Daten findet. Jede Topic-Word-Verteilungen hat die Eigenschaft, alle Wörter des Korpus aufzulisten, auch wenn das Topic von gewissen Wörtern keinen Gebrauch macht. Die Topics, die von dem Modell generiert wurden, sind durch die 10 häufigsten Schlüsselwörter und ihren Wahrscheinlichkeiten zum Auftreten definiert. Der Output des in Abbildung 4.1 definierten Algorithmus ist in Abbildung 4.2 gelistet.

---

```

OUTPUT:
[(0,
  '0.012*"ostholstein" + 0.011*"nachricht" + 0.010*"sierksdorf" + '
  '0.009*"zweckverband" + 0.008*"betreff" + 0.008*"danken" + 0.007*"email" + '
  '0.007*"hra" + 0.007*"datum" + 0.007*"hyperlink"'),
 (1,
  '0.014*"ostholstein" + 0.011*"nachricht" + 0.010*"zweckverband" + '
  '0.009*"sierksdorf" + 0.008*"sitzen" + 0.008*"danken" + 0.007*"wagrienring"
  ,
  '+ 0.007*"lübeck" + 0.007*"betreff" + 0.006*"the"'),
 (2,
  '0.014*"sierksdorf" + 0.013*"zweckverband" + 0.011*"nachricht" + '
  '0.010*"ostholstein" + 0.008*"danken" + 0.008*"betreff" + 0.007*"sitzen" + '
  '0.006*"the" + 0.006*"homepage" + 0.006*"frau"'),

```

---

---

```

(3,
'0.014*"sierksdorf" + 0.011*"ostholstein" + 0.011*"nachricht" + '
'0.010*"zweckverband" + 0.010*"betreff" + 0.008*"the" + 0.007*"frau" + '
'0.006*"danken" + 0.006*"denken" + 0.006*"lübeck"'),
(4,
'0.012*"zweckverband" + 0.011*"frau" + 0.011*"sierksdorf" + '
'0.011*"ostholstein" + 0.009*"nachricht" + 0.008*"betreff" + 0.007*"the" + '
'0.007*"danken" + 0.006*"lübeck" + 0.006*"öffentlich"'),
(5,
'0.015*"zweckverband" + 0.011*"sierksdorf" + 0.011*"ostholstein" + '
'0.009*"the" + 0.008*"betreff" + 0.007*"lübeck" + 0.007*"danken" + '
'0.007*"nachricht" + 0.007*"hyperlink" + 0.006*"sitzen"'),
(6,
'0.013*"nachricht" + 0.012*"ostholstein" + 0.012*"zweckverband" + '
'0.010*"betreff" + 0.009*"sierksdorf" + 0.007*"danken" + 0.006*"the" + '
'0.006*"lübeck" + 0.006*"frau" + 0.005*"hyperlink"'),
(7,
'0.012*"ostholstein" + 0.012*"sierksdorf" + 0.011*"zweckverband" + '
'0.010*"nachricht" + 0.008*"danken" + 0.008*"the" + 0.007*"hra" + '
'0.007*"email" + 0.007*"betreff" + 0.006*"lübeck"'),
(8,
'0.012*"ostholstein" + 0.009*"sierksdorf" + 0.008*"nachricht" + '
'0.008*"danken" + 0.008*"betreff" + 0.008*"frau" + 0.008*"zweckverband" + '
'0.008*"hyperlink" + 0.007*"email" + 0.007*"homepage"'),
(9,
'0.013*"sierksdorf" + 0.012*"zweckverband" + 0.011*"ostholstein" + '
'0.009*"nachricht" + 0.009*"danken" + 0.008*"frau" + 0.007*"the" + '
'0.007*"lübeck" + 0.006*"hyperlink" + 0.006*"sitzen"'),
(10,
'0.011*"zweckverband" + 0.011*"nachricht" + 0.010*"sierksdorf" + '
'0.010*"betreff" + 0.009*"ostholstein" + 0.009*"lübeck" + 0.007*"the" + '
'0.007*"sitzen" + 0.007*"danken" + 0.007*"hyperlink"'),
(11,
'0.013*"sierksdorf" + 0.009*"ostholstein" + 0.009*"zweckverband" + '
'0.008*"betreff" + 0.008*"nachricht" + 0.007*"danken" + 0.007*"hyperlink" +
',
'0.007*"lübeck" + 0.006*"email" + 0.006*"datum"'),

```

---

---

```

(12,
  '0.013*"ostholstein" + 0.012*"sierksdorf" + 0.009*"nachricht" + '
  '0.009*"zweckverband" + 0.008*"betreff" + 0.008*"danken" + 0.007*"email" + '
  '0.006*"frau" + 0.006*"lübeck" + 0.006*"wagrienring" '),
(13,
  '0.013*"sierksdorf" + 0.010*"the" + 0.010*"zweckverband" + '
  '0.009*"ostholstein" + 0.009*"nachricht" + 0.008*"danken" + 0.007*"lübeck"
  + '
  '0.007*"betreff" + 0.007*"frau" + 0.007*"öffentlich" '),
(14,
  '0.012*"zweckverband" + 0.011*"danken" + 0.011*"sierksdorf" + '
  '0.010*"ostholstein" + 0.010*"betreff" + 0.008*"nachricht" + '
  '0.007*"hyperlink" + 0.007*"gmbh" + 0.006*"lübeck" + 0.006*"kundennummer" '),
(15,
  '0.014*"ostholstein" + 0.011*"sierksdorf" + 0.008*"zweckverband" + '
  '0.008*"danken" + 0.008*"frau" + 0.007*"betreff" + 0.007*"the" + '
  '0.007*"nachricht" + 0.006*"amtsgericht" + 0.006*"sitzen" '),
(16,
  '0.014*"zweckverband" + 0.013*"ostholstein" + 0.011*"nachricht" + '
  '0.009*"danken" + 0.008*"sierksdorf" + 0.007*"hyperlink" + 0.007*"lübeck" +
  ,
  '0.006*"sitzen" + 0.006*"the" + 0.006*"betreff" '),
(17,
  '0.012*"ostholstein" + 0.011*"danken" + 0.011*"sierksdorf" +
    0.009*"betreff" '
  '+ 0.009*"zweckverband" + 0.008*"nachricht" + 0.007*"the" + 0.007*"frau" + '
  '0.006*"hyperlink" + 0.006*"sitzen" ')]

```

---

Abbildung 4.2: Die Topics und Schlüsselwörter

Das LDA hat erfolgreich auf allen verfügbaren Daten ein Modell gebaut, das die Daten in Topics [0, 17] eingeteilt hat. Jedes Topic hat dominante Wörter, die zusammen mit ihren jeweiligen Auftretenswahrscheinlichkeiten aufgelistet sind. Dabei sind dies nur die Wörter mit der höchsten Vorkommenswahrscheinlichkeit. Jedes Topic wird von allen verfügbaren Wörtern im Dictionary dargestellt, nur zu einem unterschiedlichen Anteil. Die Wahrscheinlichkeiten für das Auftreten von Wörtern liegt im Intervall [1.5%, 0%]. Dabei sind viele Wörter in mehreren Topics als Schlüsselwörter zu finden.

2. Die Qualität der Daten und des Klassifikators kann alternativ über die Zuordnung der Topics zu den Abteilungen untersucht werden. Die optimalen Daten werden eine perfekte Zuordnung schaffen, ohne beim Modell eine Verwirrung festzustellen. Dafür wird in dieser Arbeit untersucht, wie gut sich die Topics auf die Abteilungen zuordnen lassen. Der wohl naivste Ansatz ist, Topic  $x$  auf Abteilung  $x$  für alle  $x \in [0, 17]$  abzubilden. Dafür ist die Anzahl der Dokumente pro Topic bzw. Abteilung relevant. Dabei wird nur die Anzahl der zugeordneten Dokumente überprüft, indem für jedes auf Basis des Korpus eine Dokument-Topic-Verteilung generiert wird. Bei die-

ser wird vom Algorithmus die Topic, für die die höchste Wahrscheinlichkeit errechnet wurde, für das Dokument ausgewählt. Um welche Dokumente es sich jedoch bei dem Vergleich jeweils tatsächlich handelt ist zu diesem Punkt noch irrelevant. Überschneidungen werden noch nicht betrachtet. In Abbildung 4.3 ist die Anzahl der Dokumente in den Topics dargestellt:

Topic	Counts LDA
T0	8605
T1	3959
T2	5974
T3	1714
T4	5823
T5	2291
T6	23558
T7	12721
T8	4207
T9	1100
T10	10021
T11	3243
T12	14556
T13	2697
T14	29456
T15	1162
T16	575
T17	1400

Abbildung 4.3: Anzahl Dokumente in den Topics

Dasselbe ergibt sich für die Abteilungen und die Anzahl derer Dokumente. Dafür wird in dem von der ZVO bereitgestellten Datensatz die Zuordnung für jedes Dokument abgefragt. Dadurch kann das Dokument akkurat der Topic zugewiesen werden, wie es händisch getan wurde. Dies ist in Abbildung 4.4 zu erkennen.

Abteilung	Counts Label
L0	9770
L1	21061
L2	14245
L3	235
L4	1307
L5	251
L6	4009
L7	3610
L8	9410
L9	23533
L10	20676
L11	43
L12	2866
L13	9616
L14	8144
L15	12126
L16	4748
L17	2700

Abbildung 4.4: Anzahl Dokumente in den Abteilungen

Bei diesem Ansatz muss bedacht werden, dass ein Dokument von der ZVO in mehrere Abteilungen eingeteilt werden kann, während der Algorithmus in diesem Fall das Dokument immer nur einem Topic zuordnen. Da die Mehrheit der Dokument jedoch nur einer Abteilung zugewiesen ist, entsteht hier nur eine kleine Tendenz, die die Daten der ZVO in Abbildung 4.4 etwas mehr Dokumente zuordnet, als in Abbildung 4.3.

In Abbildung 4.4 sind die Abteilungen über Labels von 0 bis 17 aufgelistet. Dabei dient der Begriff Label der Adressierung einer Abteilung. So stellt Label 0 die Abteilung 0 der ZVO dar. Im Anwendungsfall steht jedes Label damit für eine reale Abteilung. Die Zuordnung der Labels zu Abteilungen ist in Tabelle 4.5 dargestellt.

Die Anzahl der Dokumente eines Topics kann nun mit der einer Abteilung verglichen werden, um die grundsätzliche Kompatibilität zwischen Topics und Abteilungen prüfen zu können. Dies gibt jedoch nur eine grobe Richtung an, da die Dokument selbst nicht geprüft werden. Das bedeutet, dass die verglichenen Anzahlen nicht prüfen, ob es sich sowohl in der Topic, als auch in der Abteilung auch wirklich um das gleiche Dokument handelt. Um die Dokumente mit in den Vergleich einzubeziehen und damit die Zuordnung von Topic auf Abteilung zu verbessern, wird eine weitere Methode betrachtet. Die Zuordnung der Topics zu Abteilungen kann approximiert werden, indem gezählt wird, in wie vielen Dokumenten sich die Topics und Abteilungen jeweils überschneiden. Genau wie bei der Gruppeneinteilung wird für diese Überschneidung jedes Dokument erneut auf dem Korpus geprüft, um dessen Topic-Wort-Verteilung zu erstellen. Die Dokumente werden anhand ihrer einzigartigen Zeilen-ID adressiert. Die IDs werden der Topic zugeordnet, die die höchste

#### 4 Implementierung

Label	Abteilung
L0	Ablesung
L1	Allgemeiner Schriftverkehr
L2	BM-Beschwerden
L3	KA-Kommunaler Ausfall
L4	MaKo-Klärfälle
L5	ZSB-Aktionen
L6	ZSB-BAV
L7	ZSB-Zählerdatenmanagement
L8	ZSC-Forderungsmanagement
L9	ZSK-An/Abmeldungen
L10	ZSK-Bank
L11	ZSK-Interessenten
L12	ZSK-Korrekturrechnungen
L13	ZSK-Schwierige Fälle
L14	ZSK-Stammdaten/Postrückläufer L15
ZSK-Standard Abfall L16	ZSK-Standard Verbrauch
L17	ZSK-Widersprüche Abfall

Abbildung 4.5: Die Zuordnung von Label zu Abteilung

Wahrscheinlichkeit in der Verteilung des Dokuments erreicht. Die Zuordnung findet in einer großen Liste statt, die über 18 Unterlisten verfügt, in die die IDs jeweils hinzugefügt werden. Die Überschneidungen werden in einer Matrix dargestellt, die für jedes Topic des Modells die Überschneidungen zu jeder Abteilung auflistet. Somit kann für jede Zeile die passende Zuordnung gefunden werden. Dabei stellt eine Zeile jeweils die Dokumente dar, die aufsummiert die Zahl einer Zeile in Abbildung 4.3 ergeben.



## 4 Implementierung

	LO	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11	L12	L13	L14	L15	L16	L17
T0	555	977	<b>2150</b>	21	66	19	211	251	910	960	902	2	210	596	302	758	318	250
T1	512	580	<b>650</b>	3	36	7	139	101	370	453	570	0	66	378	91	196	123	53
T2	508	755	263	9	67	21	254	145	282	<b>1498</b>	1207	1	119	300	479	527	177	130
T3	108	179	98	2	15	3	33	57	236	<b>432</b>	218	0	34	55	153	198	74	27
T4	493	867	442	5	63	8	134	180	451	<b>1111</b>	1033	8	73	348	446	489	191	89
T5	91	467	98	5	25	2	60	75	162	<b>480</b>	309	1	55	127	169	351	113	47
T6	1263	3993	1738	28	127	48	811	587	1384	4139	<b>4730</b>	6	573	2096	1620	2080	680	495
T7	<b>3376</b>	1298	870	17	524	16	221	338	662	1904	1740	9	185	381	664	891	537	133
T8	310	585	216	10	27	5	67	114	203	771	<b>1013</b>	0	81	323	264	440	205	73
T9	58	126	47	0	3	1	17	16	131	174	<b>400</b>	0	15	55	63	69	59	17
T10	286	<b>1610</b>	1733	12	51	27	284	299	858	1567	1331	4	239	893	674	673	304	248
T11	194	409	377	8	27	3	88	122	180	<b>678</b>	470	0	80	173	167	395	142	118
T12	486	2355	2495	16	80	24	564	327	1005	<b>2672</b>	1513	3	311	1254	886	1377	579	327
T13	151	393	198	5	23	4	64	83	160	414	<b>639</b>	1	68	192	123	275	151	68
T14	1008	6084	2681	86	153	61	990	820	2223	<b>5730</b>	3832	8	701	2327	1860	3093	993	552
T15	270	95	36	3	10	1	29	32	46	228	<b>284</b>	0	19	31	61	81	23	21
T16	81	82	25	1	3	0	18	38	27	<b>105</b>	90	0	14	29	37	76	24	15
T17	20	206	128	4	7	1	25	25	120	<b>217</b>	395	0	23	58	85	157	55	37

Abbildung 4.6: Anzahl der Überschneidungen zwischen LDA Topics (vert.) und ZVO Labels (horiz.) mit den maximalen Werten pro Zeile in **fett** markiert.

Diese Matrix bildet die Überschneidungen aller 133044 Dokumenten ab. Dabei sind die 18 LDA Topics auf der vertikalen Achse und die 18 LABEL Abteilungen auf der horizontalen Achse aufgetragen. Bezüglich der Zuordnung ist der naivste Ansatz, jeder Zeile (also LDA Topic) die Spalte (also Label Abteilung) mit der maximalen Überschneidung zuzuordnen. Dabei ist das Ziel die Anzahl der Gesamtüberschneidungen zu maximieren.

# 5

## Analyse

Die Implementierung des LDA Modells wurde im vorausgegangenen Teil vorgenommen. In diesem Abschnitt werden die Ausgaben analysiert und aus verschiedenen Perspektiven auf Erkenntnisse geprüft. Wie in Abschnitt 3 und 4 wird auch dieser Abschnitt in die zwei Teilbereiche dieser Arbeit unterteilt.

### 1. Gruppen und Wörter finden

Das LDA Modell hat 18 Gruppen aus den Daten generiert und diese mit einer Verteilung von Schlüsselwörtern beschrieben. Als Anwendungsziel in Bezug auf die ZVO sollen die Topics des Modells in Zukunft die Abteilungen darstellen. Betrachtet man dafür die Wörter der Topics, um die Topics semantisch den Abteilungen zuzuordnen, wird dies nicht gelingen. Das liegt daran, dass sich die dominanten Wörter in den Topics zu stark überschneiden. Zum Beispiel gehört ostholstein in allen Topics zu den häufigsten 5 Wörtern. In der folgenden Darstellung sind die häufigsten 5 Wörter aufgelistet und mit den 18 Topics auf Vorkommen geprüft:

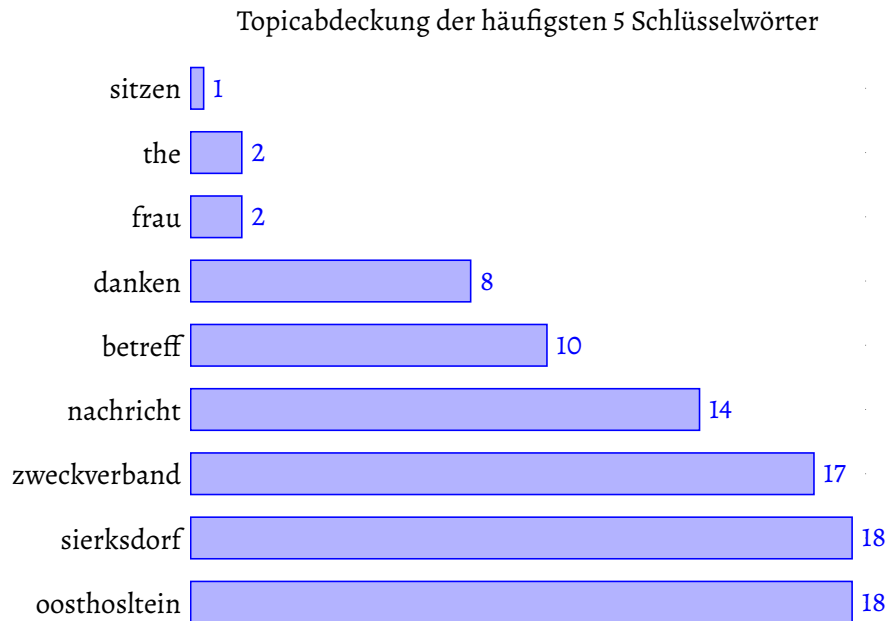
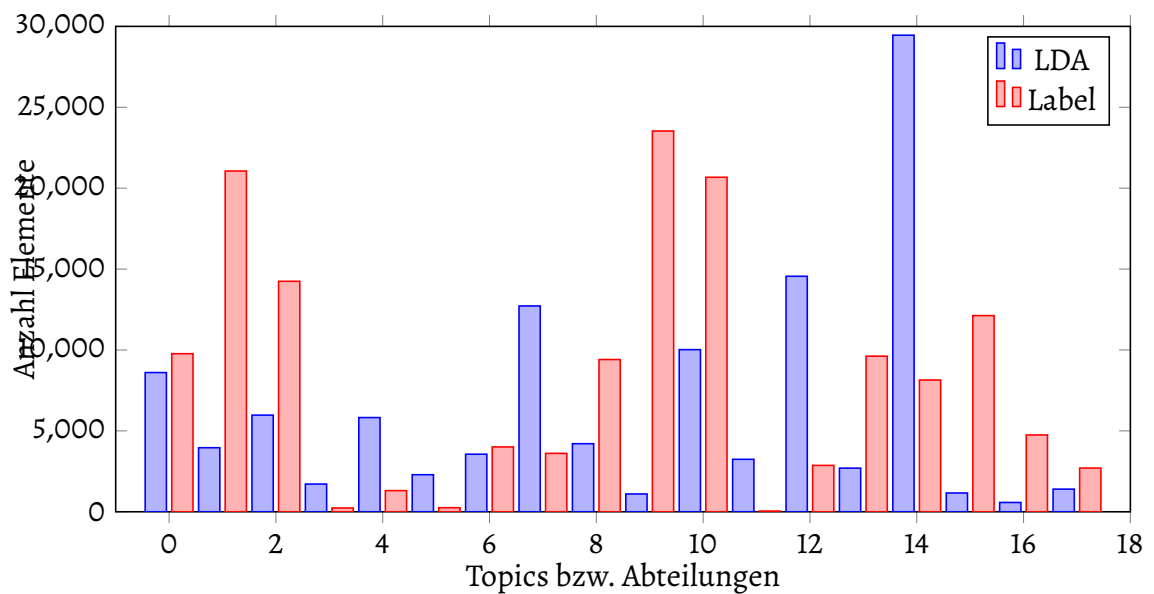


Abbildung 5.1: Topicabdeckung der häufigsten 5 Wörter aller Topics

In Abbildung 5.1 sind die häufigsten 5 Wörter aller Topics auf der Y-Achse aufgelistet. Die X-Achse beschreibt die Anzahl der Topics, die das bestimmte Wort als eines der häufigsten 5 besitzen. Hätten alle 18 Topics Schlüsselwörter, die das Topic eindeutig beschreiben, würden sich 90 verschiedene Wörter auf der Y-Achse ergeben. Hier sind jedoch nur 9 Wörter aufgezählt, deren Anzahl sich auf 90 addiert. Die Topics teilen sich also auf 90 möglichen Schlüsselwörtern nur 9 Stück. Das bedeutet, dass sich die Topics semantisch sehr schwach bis garnicht von einander unterscheiden lassen. Dies ist auf die Qualität der Daten und dessen Wörter zurückzuführen. Die Topics lassen sich nicht stark genug von einander abgrenzen, wenn in jedem Topic die selben Wörter zu einem hohen Anteil vorkommen. Somit kann die Qualität der Daten und damit die des Klassifikators erhöht werden, indem die Stoppwörter erweitert werden. Wenn man beispielsweise das Wort oostholstein betrachtet, ist dies nicht aussagekräftig für die Abteilung, da sich alle Abteilungen in Ostholstein befinden. Die Wahl der zu ergänzenden Stoppwörter hängt also mit von der semantischen Bedeutung des Wortes ab. Dazu muss auf die Anzahl der Wörter insgesamt geachtet werden und die stark vertretenen Worte dieser Art entfernt werden. Zuletzt ist bedeutend, wie stark die semantische Bedeutung des jeweiligen Wortes für die bestimmte Abteilung ist. Ein Wort, das eine ganz bestimmte Abteilung beschreibt, wie ablesen die Abteilung Ablesung, darf dies natürlich nicht entfernt werden.

## 2. Zuordnung Abteilung zu Topic

Wie in der Konzeption beschrieben ist der naivste Ansatz, um die Topics als Abteilungen zu interpretieren, den Topics der Reihenfolge nach die Abteilungen zuzuordnen. Also würde Topic 0 Abteilung 0 zugeteilt bekommen.



Das Säulendiagramm in Abbildung 2 bildet jeweils für alle  $x \in [0, 17]$  Abteilung  $x$  auf Topic  $x$  ab und stellt die Anzahl in diesen Topics bzw. Abteilungen gegenüber. Dies zeigt, dass die Topics nicht optimal auf die Abteilungen abgebildet sind. Die Anzahl der enthaltenen Dokumente sollte ähnlicher sein. Betrachtet man zum Beispiel Topic 14, kann diese nicht Abteilung 14 darstellen, da es sich um einen Unterschied von 21312 Dokumenten, also 72%, handelt.

Die Abbildung ?? zeigt, dass die Topics unabhängig von ihrem Index jedem Topic zugeordnet werden könnten. Um die korrekte Zuordnung zu finden, müssen alternative Ansätze gefunden werden. In den Abbildungen 4.3 und 4.4 ist die Anzahl der Dokumente in den jeweiligen Topics und Abteilungen aufgelistet. Um eine bessere Übersicht über die Daten zu bekommen, sind die Listen in den folgenden Abbildungen nach der Größe sortiert und mit einem weiteren Parameter dargestellt, der als Vergleichswert dienen soll. Dieser Parameter stellt den prozentualen Anteil der Topics im Korpus dar:

## 5 Analyse

Topics	Counts LDA	Anteil[%]
14	29456	22.13
6	23558	17.7
12	14556	10.94
7	12721	9.56
10	10021	7.53
0	8605	6.47
2	5974	4.49
4	5823	4,38
8	4207	3.16
1	3959	2.98
11	3243	2.44
13	2697	2.03
5	2291	1.72
3	1714	1.29
17	1400	1.05
15	1162	0.87
9	1100	0.83
16	575	0.43

Abbildung 5.2: Anzahl Dokumente in Topics mit relativem Anteil

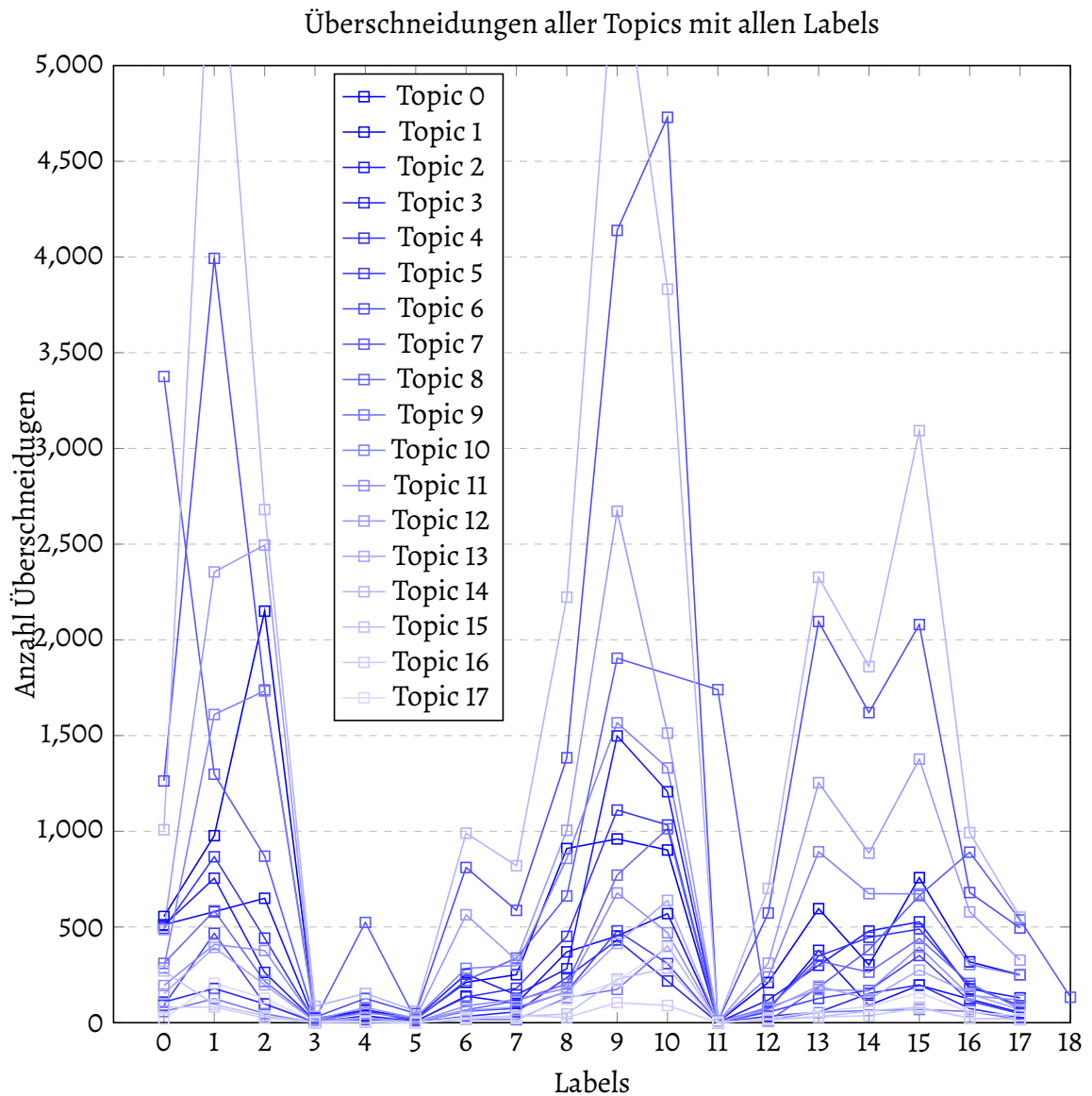
Abteilung	Counts Label	Anteil[%]
9	23533	15.86
1	21061	14.20
10	20676	13.94
2	14245	9.6
15	12126	8.17
0	9770	6.59
13	9616	6.48
8	9410	6.34
14	8144	5.49
16	4748	3.20
6	4009	2.7
7	3610	2.43
12	2866	1.93
17	2700	1.82
4	1307	0.88
5	251	0.17
3	235	0.16
11	43	0.03

Abbildung 5.3: Anzahl Dokumente in Abteilungen mit relativem Anteil

Nun wird durch die quantitative Darstellung deutlich, dass sich nicht alle Topics eindeutig einer Abteilung zuordnen lassen. Dafür sind die Anzahlen in Abbildung 5.2

und Abbildung 5.3 zu unterschiedlich. Das Ergebnis zeigt, dass die Topic-Verteilung bei LDA das 14. Thema stärker erkennt, als die ZVO die Abteilung mit den meisten zugeordneten Dokumenten. Diese unterschieden sich durch 5923 Dokumente. Die Betrachtung der unterschiedlichen Anteile dient somit nicht gut als Zuordnungsmethode, kann jedoch Aufschluss darüber geben, dass die Daten von der ZVO gleichmäßiger zugeteilt wurden, als sie von dem Topic-Modell erkannt werden. Dies bedeutet, dass die Wörter in den Dokumenten in ihrer Bedeutung und syntaktischen Umgebung zu ähnlich sind, als dass sie sich optimal in die von der ZVO vorgegebenen Abteilungen einteilen lassen. Dies könnte auch an bestimmten Wörtern liegen, die in vielen Abteilungen vorkommen, wie zu, Beispiel Ostholstein oder Zweckverband. Die Wörter können dafür sorgen, dass Dokumente in die gleiche Topic sortiert werden, die sich eigentlich durch andere Wörter semantisch stärker voneinander unterscheiden und manuell anderen Abteilungen untergeordnet wurden.

Die reine Anzahl der Dokumente ist nicht aussagekräftig genug, um eine Qualität der Zuordnung umfassend zu untersuchen. Dafür werden die Überschneidungen zwischen Topic und Abteilung betrachtet. Für die Zuordnung der Topics auf Abteilungen wurde in Abschnitt 4 in Abbildung 4.6 eine Matrix erstellt, die alle Überschneidungen gezählt hat. Diese wurden dann für alle Topics und Abteilungen in der Matrix als absolute Werte aufgetragen. Abbildung ?? stellt die Ergebnisse der Matrix in einem Graphen dar, auf dem die X-Achse die Labels, also Abteilungen, aufrägt und die Y-Achse für die Anzahl der überschneidenden Dokumente steht. Wie viele Dokumente sich in jedem Topic und Abteilung überschneiden ist durch die 18 Graphen dargestellt, die jeweils den Wert in einer Funktion aufgetragen haben. Aus Abbildung 2 ist ein globaler Trend zu erkennen, der bei den Abteilungen 1, 2 und 9, 10 sehr viele Überschneidungen aufweist.



Die Erkenntnis kann auf die hohe Anzahl der Dokumente in diesen Abteilungen zurückgeführt werden. Eine alternative Erklärung könnte sein, dass die Dokumente in den Abteilungen ein sehr breites semantisches Spektrum abdecken, womit sich viele Topics identifizieren können. Wählte man für jede Topic die Abteilung mit der maximalen Überschneidung, bekommt man die Liste in Abbildung 5.4, die jedem Index, der die Topics repräsentiert, die Abteilungen als Werte zuordnet.

---

```
maxmatrix = [2,2,9,9,9,9,9,10,0,10,10,1,9,9,10,9,10,9,9]
```

---

Abbildung 5.4: Liste der zugeordneten Topics und Abteilungen bei maximaler Überschneidung pro Topic

Für diese Zuordnung prüfen wir nun die Anzahl der Dokumente in den jeweils gewählten Topics bzw. Abteilungen. Dabei ist die reine Schnittmenge nicht der alleinige Parameter. Dafür werden die Anteile aus Abbildung 4.3 und Abbildung 4.4 zum Vergleich herangezogen. Eine Zuordnung ist nur dann gut, wenn die Überschneidungsmenge möglichst nah an der Gesamtmenge der zugeordneten Dokument in der Topic bzw. Abteilung ist. Eine vergleichsmäßig hohe Schnittmenge heißt nicht automatisch eine korrekte Zuteilung, wenn die Abteilung zum Beispiel manuell deutlich mehr Daten zugeordnet hat.

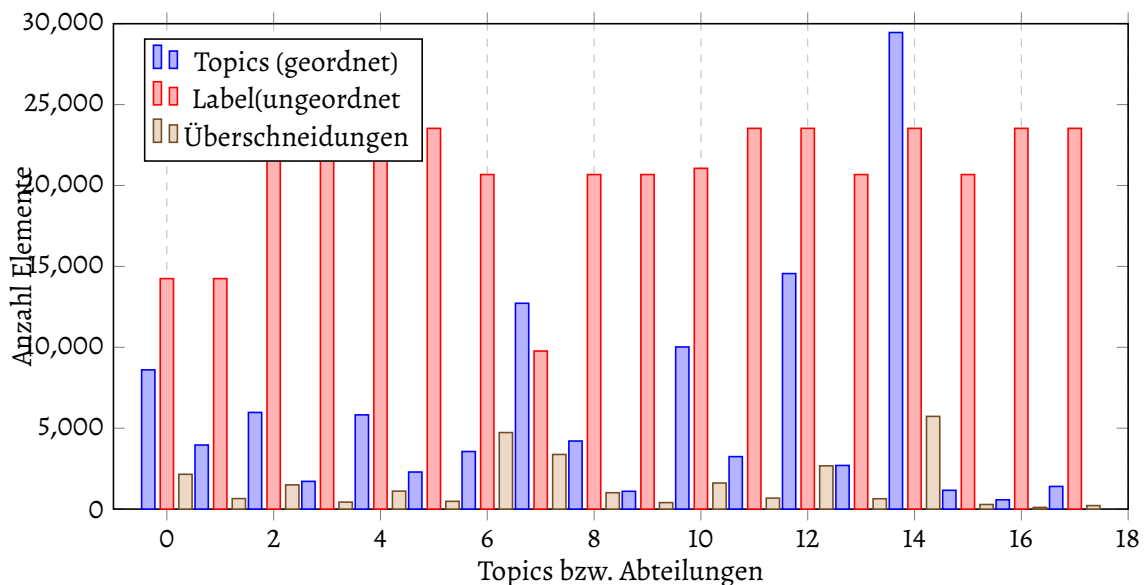


Abbildung 5.5: Die aus Abbildung 5.4 zeigt die gewählten Zuordnungen. Diese werden in dieser Abbildung mit den absoluten Dokumentmengen der Topics bzw. Abteilungen verglichen.

### HIER DIE NEUE CHART HIN MIT ÜBERSCHNEIDUNGEN IM VGL ZU BETRAG DER ABTEILUNG BZW TOPIC

Wie zu erkennen ist, sind nur die Zahlen 0, 1, 2, 9, 10 in der Liste vertreten. Eine optimale Zuordnung wäre jedoch erst erreicht, wenn alle Zahlen von 1 bis 18 ohne Duplikate in der Liste in einer beliebigen Reihenfolge vorkommen. Dies liegt daran, dass eine Topic nur genau eine Abteilung darstellen darf. In der Liste wird jedoch deutlich, dass viele Themen des LDA Modells mit dem neunten Thema der ZVO kompatibel wären. Dabei werden 9/18 Topics Abteilung 9, 1/18 Topic Abteilung 1, 5/18 Topics Abteilung 10 und 2/18 Topics Abteilung 2 zugeordnet. Hier fällt auf, dass diesen Abteilungen die meisten Dokumente von der ZVO händisch zugeordnet wurden. Somit können sie sich auch mit vielen Dokumenten aus den LDA Topics überschneiden. Dies lässt zusätzlich vermuten, dass zum Beispiel die Abteilung 9, die laut Abbildung 4.4 die meisten Dokumente enthält, ein zu breites inhaltliches Spektrum abdeckt und in noch weitere Unterthemen unterteilt werden könnte. Dies wird dadurch begründet, dass das Topic-Modell viele der Dokumente aus Abteilung 9 in verschie-



dene Topics eingeteilt hat.

Die Zuteilung anhand der maximalen absoluten Überschneidungen ist nicht injektiv. Für die Verarbeitung der Daten ist die injektive Zuordnung jedoch fundamental wichtig, aber schwierig zu erreichen. Für die Beurteilung der Matrix führen wir einen weiteren Parameter ein, die durchschnittliche Überschneidung. Dadurch kann die Kompatibilität der Abteilungen im Bezug auf das Matching mit Topics besser analysiert werden. Vorallem bei Abteilungen, die eine sehr unausgeglichene Überschneidungsmengen haben, da sie deutlich mehr Dokumente als andere enthalten, bietet der Durchschnitt eine alternative Sichtweise. Für die durchschnittliche Überschneidung wird zuerst die Summe aller Dokumente einer Abteilung errechnet, indem alle Werte einer Spalte addiert werden. Daraufhin entsteht eine neue Matrix, in der jeder Wert jeweils durch die Summe seiner Spalte dividiert wird. Dabei wird die Anzahl an Dokumenten in jeder Abteilung irrelevant für das Endergebnis. Somit beschreibt jeder Wert, wie groß der Anteil der aus dieser Topic überschneidenden Dokumente bezogen auf die gesamte Abteilung ist in %:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	5.68	4.64	15.09	8.94	5.05	7.57	5.26	6.95	9.67	4.08	4.36	4.65	7.33	6.2	3.71	6.25	6.7	9.26
1	5.24	2.75	4.56	1.28	2.75	2.79	3.47	2.8	3.93	1.92	2.76	0	2.3	3.93	1.12	1.62	2.59	1.96
2	5.2	3.58	1.85	3.83	5.13	8.37	6.34	4.02	3	6.37	5.84	2.33	4.15	3.12	5.88	4.35	3.73	4.81
3	1.11	0.85	0.69	0.85	1.15	1.2	0.82	1.58	2.51	1.84	1.05	0	1.19	0.57	1.88	1.63	1.56	1
4	5.05	4.12	3.1	2.13	4.82	3.19	3.34	4.99	4.79	4.72	5	18.6	2.55	3.62	5.48	4.03	4.02	3.3
5	0.93	2.22	0.69	2.13	1.91	0.8	1.5	2.08	1.72	2.04	1.49	2.33	1.92	1.32	2.08	2.89	2.38	1.74
6	12.93	18.96	12.2	11.91	9.72	19.12	20.23	16.26	14.71	17.59	22.88	13.95	19.99	21.8	19.89	17.15	14.32	18.33
7	34.55	6.16	6.11	7.23	40.09	6.37	5.51	9.36	7.04	8.09	8.42	20.93	6.45	3.96	8.15	7.35	11.31	4.93
8	3.17	2.78	1.52	4.26	2.07	1.99	1.67	3.16	2.16	3.28	4.9	0	2.83	3.36	3.24	3.63	4.32	2.7
9	0.59	0.6	0.33	0	0.23	0.4	0.42	0.44	1.39	0.74	1.93	0	0.52	0.57	0.77	0.57	1.24	0.63
10	2.93	7.64	12.17	5.11	3.9	10.76	7.08	8.28	9.12	6.66	6.44	9.3	8.34	9.29	8.28	5.55	6.4	9.19
11	1.99	1.94	2.65	3.4	2.07	1.2	2.2	3.38	1.91	2.88	2.27	0	2.79	1.8	2.05	3.26	2.99	4.37
12	4.97	11.18	17.51	6.81	6.12	9.56	14.07	9.06	10.68	11.35	7.32	6.98	10.85	13.04	10.88	11.36	12.19	12.11
13	1.55	1.87	1.39	2.13	1.76	1.59	1.6	2.3	1.7	1.76	3.09	2.33	2.37	2	1.51	2.27	3.18	2.52
14	10.32	28.89	18.82	36.6	11.71	24.3	24.69	22.71	23.62	24.35	18.53	18.6	24.46	24.2	22.84	25.51	20.91	20.44
15	2.76	0.45	0.25	1.28	0.77	0.4	0.72	0.89	0.49	0.97	1.37	0	0.66	0.32	0.75	0.67	0.48	0.78
16	0.83	0.39	0.18	0.43	0.23	0	0.45	1.05	0.29	0.45	0.44	0	0.49	0.3	0.45	0.63	0.51	0.56
17	0.2	0.98	0.9	1.7	0.54	0.4	0.62	0.69	1.28	0.92	1.91	0	0.8	0.6	1.04	1.29	1.16	1.37
$\Sigma$	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100

Abbildung 5.6: Durchschnittliche Überschneidungen aller Topics mit Labels

Wählt man aus jeder Zeile das maximale Element aus und notiert die assoziierten Abteilungen in einer Liste, dann ergibt sich das folgende Ergebnis. Der Unterschied bei der Darstellung ist nun, dass die Zahlen pro Zeile nicht mehr als absolutes Maß betrachtet werden, sondern als relative Häufigkeit in der jeweiligen Abteilung. Somit macht es keinen Unterschied, dass die Abteilungen verschieden viele Dokumente von der ZVO zugeordnet bekommen haben.

---

```
average_row = [2,0,5,8,11,15,10,4,10,10,2,17,2,16,3,0,7,10]
```

---

Abbildung 5.7: Zuordnung zwischen LDA und Label

Man erkennt bereits eine Verbesserung zu Abbildung 4.6, da in der Zuteilung mehr Abteilungen zugeordnet wurden. Insgesamt sind 12/18 in der Liste vertreten. Wie bei der ersten Liste, bei der 0, 1, 2, 9, 10 mehrfach zugewiesen wurden, sind hier 2 und 10 drei mal zugewiesen und 0 zwei mal. Somit unterstreicht das Ergebnis die Problematik der Zuteilung der ZVO in Bezug auf diese Abteilungen.

# 6

## Zusammenfassung und Ausblick

In dieser Arbeit wurde das Ziel verfolgt, die Qualität des Klassifikators zu analysieren und die Gruppierung der Abteilungen der ZVO zu untersuchen. Dabei beschäftigt sich der eine Teil der Arbeit mit der reinen Topic Modellierung und dessen Analyse. Bei dieser wurde die Erkenntnis gewonnen, dass die Stoppwörter noch weiter ausgebaut werden müssen, um die Qualität zu verbessern. Zu diesen Stoppwörtern gehören unter anderem die folgenden: ostholstein, sierksdorf, zweckverband, nachricht, betreff, danken, the und sitzen. Im zweiten Teil der Arbeit wurden die Abteilungen untersucht, indem versucht wurde, diesen die Topics zuzuordnen. Aufgefallen ist, dass die Zuteilung der Dokument sehr ungleich ist und Abteilungen wie L9 oder L10 zu viele Dokumente enthalten und in eindeutigeren Unterthemen aufgeteilt werden sollten, um die Klassifikation zu verbessern.

...