

Product Safety and Quality Engineering
Manuel Löwer und Nadine Schlüter (Hrsg.)

1

Webbasierte Applikation eines NLP-Modells zur automatisierten Auswertung nutzergenerierter Inhalte für die Produktentwicklung

Franz Wieck



PRODUKT
SICHERHEIT
QUALITÄT



BERGISCHE
UNIVERSITÄT
WUPPERTAL



BERGISCHE
UNIVERSITÄT
WUPPERTAL

**Webbasierte Applikation eines NLP-Modells zur
automatisierten Auswertung nutzergenerierter In-
halte für die Produktentwicklung**

Dissertation
zur Erlangung eines Doktorgrades

in der
Fakultät für Maschinenbau und Sicherheitstechnik
der
Bergischen Universität Wuppertal

vorgelegt von

Franz Wieck

Wuppertal 2022

Product Safety and Quality Engineering – Band 1

Franz Wieck

Webbasierte Applikation eines NLP-Modells zur automatisierten Auswertung nutzergenerierter Inhalte für die Produktentwicklung

Herausgeber: Manuel Löwer und Nadine Schlüter,
Bergische Universität Wuppertal,
Fachgebiet Produktsicherheit und Qualität,
Gaußstraße 20, 42119 Wuppertal

Umschlaggestaltung: Franz Wieck

Zugl.: Wuppertal, Univ., Diss. 2022

DOI: <https://doi.org/10.25926/BUW/0-2>

URN: <https://nbn-resolving.org/urn:nbn:de:hbz:468-2-21>

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.dnb.de> abrufbar.

Dieses Werk steht, soweit nicht anders angegeben, unter der Creative Commons-Lizenz CC BY-NC 4.0

<https://creativecommons.org/licenses/by-nc/4.0/deed.de>

Kurzfassung

Große Menge an Daten im Internet stellen ein besonderes Informationspotential für die nutzerzentrierte Produktentwicklung dar, insbesondere Kundenrezensionen auf Online-Handelsplattformen bieten aufgrund des direkten Produktbezugs einen besonderen Nutzen für die Produktentwicklung. Für eine Auswertung dieser Kundenbewertungen bedarf es Kenntnissen aus der Linguistik, der künstlichen Intelligenz und der Produktentwicklung.

In dieser Arbeit werden nutzergenerierte Inhalte mit Hilfe von künstlicher Intelligenz (Natural Language Processing) ausgewertet und für die Produktentwicklung aufbereitet, um eine datenbasierte Entscheidungsgrundlage für Produktinnovationen und -verbesserungen zu schaffen.

Die wichtigsten Ergebnisse dieser Arbeit sind: i) eine Methode zur Aufbereitung nutzergenerierte Texte, ii) eine Logik zur Auswertung gelabelter Textdaten und iii) ein kontextspezifisches und anforderungsgerechtes Visualisierungskonzept. Aus den Ergebnissen der Fallstudie lässt sich schließen, dass sowohl das webbasierte Tool, die entwickelte Methode als auch die Logik fähig sind, nutzergenerierte Inhalte für die Produktentwicklung sinnvoll und nutzbringend aufzubereiten.

Für das Forschungsfeld stellt die Entwicklung des Tools und die Methode erstmalig ein gesamtheitliches Rahmenwerk für die Textdatenauswertung mithilfe von NLP für die Produktentwicklung dar.

Für die Produktentwicklung stellt diese Arbeit eine sehr individualisierte und änderbare Software und Methodenbasis zur Verfügung, um unstrukturierte Textdaten aufzubereiten. Das Gesamtergebnis ist, dass das hier entwickelte Tool das Kundenverständnis deutlich verbessert und damit das Informationsmanagement im Unternehmen nachhaltig stärkt und die Wettbewerbsfähigkeit des Unternehmens auch in dynamischen Märkten wie heute sichert.

Schlagwörter: NLP, nutzergenerierte Inhalte, Produktentwicklung, Django

Abstract

Large amounts of data on the internet represent a special information potential for user-centred product development; in particular, customer reviews on online trading platforms offer a special benefit for product development due to the direct product reference. An evaluation of these customer reviews requires knowledge from linguistics, artificial intelligence and product development.

In this thesis, user-generated content is evaluated with the help of artificial intelligence (natural language processing) and processed for product development in order to create a data-based decision-making basis for product innovations and improvements.

The most important results of this work are: i) a method for processing user-generated texts, ii) a logic for evaluating labelled text data, and iii) a context-specific and requirement-oriented visualisation concept. From the results of the case study it can be concluded that both the web-based tool, the method and the logic are capable of preparing user-generated content for product development in a meaningful and useful way.

For the research field, the development of the tool and the method represents the first holistic framework for text data evaluation using NLP for product development.

For product development, this work provides a highly individualised and modifiable software and method basis for processing unstructured text data. The overall result is that the tool developed here significantly improves customer understanding and thus sustainably strengthens information management in the company and ensures the company's competitiveness even in dynamic markets like today.

Keywords: NLP, user-generated content, product development, Django

Inhaltsverzeichnis

1	Einleitung.....	1
1.1	Problemstellung.....	1
1.2	Aufbau der Arbeit.....	3
2	Ausgangssituation und Begriffsdefinitionen.....	7
2.1	Der Produktentwicklungsprozess.....	7
2.1.1	Ingenieurwissenschaftliche Betrachtungsweise.....	8
2.1.2	Management-basierte Perspektive.....	10
2.1.3	Moderne Entwicklungsprozesse.....	11
2.1.4	Bewertung der Perspektiven in Bezug auf den Nutzerfokus.....	12
2.2	Produktkategorien.....	13
2.3	Die Integration des Nutzers in den Produktentwicklungsprozess.....	14
2.4	Datenbasierte Ansätze zur Verarbeitung von Nutzerinformationen für die Produktentwicklung.....	16
2.5	Nutzergenerierte Inhalte im Internet.....	17
2.6	Formen und Datenbanken nutzergenerierter Inhalte.....	18
2.7	Zusammenfassung der beschriebenen Ausgangssituation.....	19
2.8	Begriffsdefinitionen.....	20
3	Stand der Technik.....	25
3.1	Natural Language Processing (NLP).....	26
3.1.1	Trends und Paradigmen.....	26
3.1.2	Methoden, Prinzipien und Funktionsweisen.....	29
3.1.3	Bewertungskriterien.....	36
3.2	Methoden zur Nutzerintegration.....	38
3.3	Online Bewertungen als Forschungsgegenstand.....	41
3.4	Zusammenfassung, Bewertung und Abgrenzung zum Stand der Technik.....	44
4	Handlungsbedarf und Zielsetzung.....	47
4.1	Handlungsbedarf.....	47
4.2	Zielsetzung.....	49
5	Entwicklung des Modells.....	53
5.1	Entwicklung einer Methode zur Beschriftung von nutzergenerierten Textdaten.....	53

5.2	Entwicklung einer Logik zur Auswertung gelabelter Textdaten	57
5.2.1	Anforderung an die Textanalyse und Auswertlogik	57
5.2.2	Integration der Analyseziele in den Aufgabenbereich der Produktentwicklung ...	58
5.2.3	Stimmungsanalyse	59
5.2.4	Verwendete Textverarbeitungsalgorithmen	60
5.3	Grobkonzept für die Visualisierung.....	61
5.3.1	Anforderung an das Visualisierungskonzept	61
5.3.2	Visualisierungskonzept.....	61
6	Webbasierte Umsetzung des Modells	63
6.1	Anforderungen an die Implementierung	63
6.2	Software-Entwicklungsumgebung und verwendetes Framework	64
6.3	Webbasierte Applikation – Aufbau, GUI und Funktionsweise	66
6.3.1	Textdaten hochladen.....	68
6.3.2	Texte labeln.....	68
6.3.3	Label bearbeiten	69
6.3.4	Datenauswertung.....	70
6.3.5	Datenmanagement und Datenbankstruktur.....	73
6.4	Umsetzung der Applikation.....	77
6.4.1	Models.....	80
6.4.2	Templates	80
6.4.3	Views	84
6.5	Zwischenfazit.....	85
7	Validierung anhand einer realen Fallstudie.....	87
7.1	Datensatz: Rezensionen von <i>Amazon.de</i>	87
7.2	Label erstellen	87
7.3	Rezensionen labeln.....	89
7.4	Textanalyse	91
7.5	Auswertung und Visualisierung	94
8	Ergebnisse und Diskussionen.....	101
8.1	Aufbereitung von nutzergenerierten Inhalten für die Produktentwicklung	101
8.2	Algorithmus-basierte Auswertung gelabelter Textdaten	102
8.3	Praxisnahe Implementierung und praktische Implikationen.....	103
9	Fazit und Ausblick.....	105

Literaturverzeichnis	109
Anhang.....	117
A.1 Übersicht existierender Textkorpora.....	117
A.2 Nomenklatur zur Bezeichnung von Wortarten	118
A.3 Ablaufschema der Änderung eines Datensatzes	118
A.4 Ausschnitte Software-Code.....	119
A.4.1 Django-Modelle	119
A.4.2 Django-Views.....	121
A.5 Label-Auswertung.....	126

Abbildungsverzeichnis

Abbildung 1: Schematischer Aufbau der Zusammenhänge und betrachteten Themengebiete für die Auswertung nutzergenerierter Inhalte für die Produktentwicklung im Rahmen dieser Arbeit	4
Abbildung 2: Zeitlicher Verlauf der Anzahl wissenschaftlichen Veröffentlichungen im Bereich Management und Ingenieurwesen zum Thema Produktentwicklungsprozess (Citation Report von WebOfScience.com).....	7
Abbildung 3: Ingenieursperspektive auf die Produktentwicklung (Pahl und Beitz 1977).....	9
Abbildung 4: Phasen der Produktentwicklung aus Management Perspektive nach O'Connor und DeMartino (2006).....	10
Abbildung 5: Die verschiedenen Arten der Kundenintegration nach Schweitzer et al. (2020) (Quelle: Eigene Darstellung).....	15
Abbildung 6: Nutzerinformationen im Produktentwicklungskontext: Informationsgrundlage, Analyseziele und datenbasierte Methoden (eigene Darstellung).....	16
Abbildung 7: Top 20 Webseiten in Deutschland nach der Anzahl der Unique Visitors im Juni 2021 (in Millionen) (Statista 2021c).....	17
Abbildung 8: Verwendeter Ansatz zur Verarbeitung von Kundeninformationen (basierend auf Abbildung 6).....	20
Abbildung 9: Zwei Beispiele für Produktarchitekturen (angelehnt an Ulrich 1995, S. 6)	21
Abbildung 10: Hierarchischer Aufbau von Rezensionen auf Online-Handelsplattformen vom einzelnen Rezensions-Text bis zur gesamten Portfolio-Rezensions-Sammlung	23
Abbildung 11: Zeitlicher Verlauf und Paradigmenwechsel von NLP (Cambria und White 2014, S. 51).....	27
Abbildung 12: Hauptworte und deren Abkürzungen nach STTS (Stuttgart-Tübingen-TagSet) (nach Schiller et al. 1999)	30
Abbildung 13: Graphendarstellung für das POS-Tagging eines einfachen Teilsatzes (HU BERLIN).....	30
Abbildung 14: Worthäufigkeitstabelle - Ein Zwischenschritt bei der Vektorisierung von Texten	34
Abbildung 15: Darstellungsform eines ausgewerteten Datensatzes mit der Methode Bag-of-Words	35
Abbildung 16: Die Sigmoid Funktion bildet die Grundlage der Einteilung bei der logistischen Regression	36
Abbildung 17: Gegenüberstellung der betrachteten wissenschaftlichen Ansätze und Einordnung der vorliegenden Arbeit	46

Abbildung 18: Einordnung der formulierten Teilziele (Sterne 1-5) in den Gesamtprozess (Kreise 1-6) für die Auswertung von nutzergenerierten Inhalten.....	51
Abbildung 19: Charakterisierung von Textpassagen anhand von Formulierungen und Inhalten und deren Merkmalen und Ausprägungen.....	54
Abbildung 20: Prozessbeschreibung der entwickelten Methode zur Beschriftung von Textdaten.....	55
Abbildung 21: Schematischer Ablauf der Zuordnung der Wortgruppen der Rezensionen zu Labeln.....	56
Abbildung 22: Anforderungsprofil für die Textanalyse (Anwendungsfall auf die Übersicht von Abbildung 6 angewendet)	58
Abbildung 23: Integration der Analyseziele in den Produktentwicklungsprozess.....	59
Abbildung 24: Schematischer Aufbau der Textanalyse der Wortgruppen	60
Abbildung 25: Geeignete Kennzahlen (Einzelwerte – Anzahl) der Elemente für eine tabellarische Darstellung im Dashboard	62
Abbildung 26: Dashboard-Dummy der Visualisierungskonzepte (links: zeitlicher Verlauf mit Graph, mitte: zeitlicher Verlauf Balkendiagramm, rechts oben: tabellarische Form, rechts unten: Listendarstellung).....	62
Abbildung 27: Verwendete Entwicklungsumgebung PyCharm der Firma JetBrains.....	65
Abbildung 28: Windows PowerShell - Befehle zum Starten des lokalen Servers für die Webapplikation	65
Abbildung 29: Bestandteile und Verknüpfungen der webbasierten Applikation des Modells..	67
Abbildung 30: Graphische Oberfläche des Web-Tools zum Labeln der Rezensionen (Home-View).....	69
Abbildung 31: GUI zur Bestimmen des neuen Labels und einer Farbe	70
Abbildung 32: Überblick der verschiedenen Bereiche der Datenauswertung	71
Abbildung 33: Aufbau und Zusammensetzung der graphischen Auswertung.....	72
Abbildung 34: Datenbankstruktur im Backend des Tools.....	73
Abbildung 35: Struktur der Benutzerinformationen im Backend der Webapplikation.....	74
Abbildung 36: Abfrageformular für die Registrierung (links) und Login-Formular (rechts)	75
Abbildung 37: Datenbankstruktur der Rezension, Label und gelabelten Rezensionen.	76
Abbildung 38: Exemplarischer Datensatz der Datenbank LabeledReview (links) und Profilansicht eines Users (rechts)	77
Abbildung 39: Django's Systemarchitektur MVT basierend auf dem Model-View-Control-Ansatz (angelehnt an George 2020, S. 17).....	79
Abbildung 40: Schematischer Aufbau einer Webseite (Wireframe) (angelehnt an George 2020, S. 23).....	81

Abbildung 41: Visualisierung des Analytics-Templates ohne Integration (links) und mit Integration (rechts) in das Django-Projekt in einem Webbrowser.....	83
Abbildung 42: Beispielhafte Darstellung für den Zusammenhang zwischen Model-View-Template bei der Webapplikation	83
Abbildung 43: Auswahl der Produktmerkmale eines Stechbeitel-Sets der Firma Kirschen.....	88
Abbildung 44: Unterscheidung zwischen Gesamtnutzung und Anzahl Rezensionen	90
Abbildung 45: Details einer gelabelten Rezension	91
Abbildung 46: Vergleich der Ergebnisse der Textanalyse in Abhängigkeit von der Sprache (als integrierte Funktion des Tools).....	92
Abbildung 47: Vergleich der Ergebnisse deutscher und englischer Implementierung.....	93
Abbildung 48: Auswertung - Anzahl nützlicher Rezensionen und gelabelter Textpassagen (Toolausschnitt)	94
Abbildung 49: Auswertung - Anzahl verwendeter Label, Stimmung und Subjektivität.....	95
Abbildung 50: Auswertung der Label mittels Gruppenbildung.....	96
Abbildung 51: Verschiedene Auswertungen aus dem Diagramm-Bereich (links: Anzahl Rezensionen pro Jahr; mitte: Stimmungs- und Subjektivitätswerte für die Gesamtrezensionen; rechts: Stimmungs- und Subjektivitätswerte für das Label #Klinge	98
Abbildung 52: Auswertung der Stimmungs- und Subjektivitätsverteilung für verschiedene Label	99
Abbildung 53: Auswertung - Textdetail-Bereich für die vergebenen Label.....	100
Abbildung 54: STTS-Tags gemäß TIGER-Annotationsschemata (HU BERLIN)	118
Abbildung 55: Änderungsprozedur für eine gelabelte Textpassage	118
Abbildung 56: Auswertung der Stimmungs- und Subjektivitätsverteilung.....	127

Tabellenverzeichnis

Tabelle 1:	Vergleich der Perspektiven und Bewertung in Hinblick auf den Nutzerfokus... 12
Tabelle 2:	Unterscheidung zwischen Nutzende und Kundschaft in Abhängigkeit der Intention 22
Tabelle 3:	Ausschnitt verschiedener Textkorpora für die NLP-Bibliothek NLTK (Bird et al. 2009) (Eine vollständige Übersicht ist dem Anhang A.1 zu entnehmen)..... 30
Tabelle 4:	Übersicht Methodenkatalog zur Kundenintegration und Potentialanalyse zur Informationsanreicherung der Methoden auf Basis NLP extrahierter nutzergenerierten Inhalten..... 40
Tabelle 5:	Übersicht der analysierten Publikationen..... 41
Tabelle 6:	Kennwerte, praktische Arbeitsanweisungen und theoretische Zielwerte für die Zuordnung der Label..... 56
Tabelle 7:	Übersicht der NLP-Funktionen des webbasierten Tools..... 60
Tabelle 8:	Exemplarischer Aufbau und Bestandteile des verwendeten Datensatzes 87
Tabelle 9:	Auswahl der relevanten Lebensdauerphasen für das Produktbeispiel 89
Tabelle 10:	Detailuntersuchung der Veränderung der Ergebnisse der Textanalyse in Abhängigkeit von der Sprache 92
Tabelle 11:	Erweiterte Auswertung - Anzahl verwendeter Label und Häufigkeitsverteilung prozentual..... 97
Tabelle 12:	Übersicht verschiedener Textkorpora für die NLP-Bibliothek NLTK (Bird et al. 2009)..... 117
Tabelle 13:	Erweiterte Auswertung - Anzahl verwendeter Label und Häufigkeitsverteilung 126

1 Einleitung

Ingenieur:innen konzeptionieren, berechnen und konstruieren um technische Fragestellungen oder Herausforderungen zu lösen. Die Komplexität der Maßnahmen und die Zusammenhänge in der Produktentwicklung verlangen eine (intelligente) Unterstützung der Entwicklungsteams in Form von Prozesswissen, um den Projekterfolg maßgeblich positiv zu beeinflussen (Ramesh und Tiwana 1999, S. 233). Software hilft ihnen dabei. Früher waren vorgefertigte und komplexe Computerprogramme, wie CAD-, Produktdatenmanagement-, Simulation- oder CAM-Systeme den großen Firmen vorbehalten. Heute müssen bestehende Softwarelösungen verfügbar und anpassbar auf die speziellen technischen Fragestellungen sein, um richtige Entscheidungen in der Produktentwicklung zu treffen. Insbesondere die Entwicklung neuer Produkte ist eine sehr komplexe und wissensintensive Tätigkeit. Für Unternehmen, die Produkte in einem Umfeld entwickeln, das durch kürzere Produktlebenszyklen, dynamische Märkte und komplexe Prozesse gekennzeichnet ist, ist das Wissensmanagement eine nachhaltige Strategie zur Erhaltung des Wettbewerbsvorteils (Madhavan und Grover 1998; Göpfert 1998; Bender und Gericke 2021; Feldhusen und Grote 2013). Die Herausforderung besteht darin, das Wissensmanagement in das Unternehmen zu integrieren, so dass wissenszentrierte Aktivitäten im Entwicklungsteam und im gesamten Unternehmen möglich sind und gefördert werden (Madhavan und Grover 1998, S. 1). Bei der Entwicklung von marktfähigen Produkten (Innovationen) müssen technisches Know-how und Prozesswissen mit Informationen über die Kunden verknüpft werden, damit das Produkt den Bedürfnissen der Kunden entspricht (Su et al. 2006, S. 784). Zusammenfassend lässt sich sagen, dass die aktuelle Entwicklung von Produktinnovationen gekennzeichnet ist durch i) einen verkürzten Produktlebenszyklus, ii) ein verbessertes Verständnis des Wissensmanagements über Kundeninformationen und iii) eine umfassendere Anwendung datenbasierter Methoden für die Produktentwicklung (Zhan et al. 2019). Diese Randbedingungen stellen Ingenieur:innen und Designer:innen in der Produktentwicklung vor Herausforderungen, insbesondere bei kleinen und mittleren Unternehmen (KMU), weil die zur Verfügung stehenden Ressourcen nicht ausreichen, um die erforderlichen Maßnahmen zu realisieren (Cantamessa et al. 2020). Welche weiteren Probleme in diesen drei Bereichen auftreten wird im Folgenden näher erläutert.

1.1 Problemstellung

Immer kürzere Produktlebenszyklen, die Notwendigkeit, sich auf neue Märkte und Kundensegmente einzustellen, um weiterhin innovative Produkte zu entwickeln und der zunehmende Trend zum Outsourcing, verändern die Art der Technologieentwicklung in Firmen (Narula 2004). Netzwerkaktivitäten und strategische Allianzen werden genutzt, um die komplexen (multitechnologischen) Produkte

zu entwickeln (ebd.). Diesem Trend zu begegnen, sollten die Entwicklungsteams befähigt werden, sich selbst Wissen anzueignen, um neue Aufgaben zielgerichtet und sicher zu erfüllen (Madhavan und Grover 1998). Die Einbeziehung von Wissen über den sich dynamisch verändernden Markt und das damit verbundene Wissensmanagement ist eine komplexe Aufgabe, insbesondere für Informationen über die Kunden oder nutzende Person des Produktes (Kim und Wilemon 2009; Pavanelli Stefanovitz und Lopes de Sousa Jabbour 2021).

Die aktuellen Randbedingungen der Produktentwicklung erfordern ein verbessertes Verständnis des Wissensmanagements über die Kundeninformationen. In Wissenschaft und Praxis besteht Einigkeit darüber, dass Kundeninformationen wichtig sind. Die Möglichkeiten, Einflussgrößen und Schwierigkeiten werden auf unterschiedlichsten Ebenen untersucht. Ob es um die Risikobewertung der Kundenintegration (Song et al. 2013), den Einfluss nutzerzentrierten Designs (Veryzer und Borja de Mozota 2005) oder die Integration von Kunden aus Sicht des Managements (Bartl et al. 2012) geht, bei allen Untersuchungen steht die Kundschaft oder die Kundeninformationen als Forschungsgegenstand im Fokus. Damit dieses Wissensmanagement von Kundeninformationen den KMU für die Produktentwicklung zur Verfügung gestellt werden kann, genügt es nicht, die Informationen zu sammeln. Vielmehr müssen die Informationen weiterverarbeitet werden. Die datenbasierten Methoden für die Produktentwicklung sind in den letzten Jahrzehnten stark weiterentwickelt worden und stellen für KMU aktuell die größte Barriere dar.

Eine sehr passende und viel referenzierte Beschreibung für datenbasierte Prozessketten, beschreiben Chen et al. (2014) zur Erläuterung der Wertschöpfungskette bei Big Data (große Datenmengen) Anwendungen. Die vier Phasen sind: Datenerzeugung, Datenerfassung, Datenspeicherung und Datenanalyse (ebd.). Die Herausforderung bei Big Data besteht darin, dass durch die vervielfachte Datenerzeugung, eine so große Datenmenge entsteht, die die aktuellen Methoden nicht verarbeiten können. Das bedeutet, dass die Datenspeicherung mit üblichen Computersystemen oder Datenbankstrukturen nicht mehr abbildbar ist und dadurch die Analyse dieser Daten andersartig, nämlich nicht auf Basis von klassisch gespeicherten Daten, erfolgen muss. Obschon diese Herausforderungen direkt auf die Produktentwicklungstätigkeiten übertragbar sind, mangelt es an einem Verständnis dafür, wie Unternehmen Big Data nutzen können, um Produktinnovationen zu entwickeln (Johnson et al. 2017). Wenngleich dies mehrere Vorteile mit sich brächte: Verkürzung der Markteinführungszeit, Verbesserung der Produktakzeptanz bei den Kunden, Senkung der Kosten und besseres Verständnis für die Kundschaft (Zhan et al. 2018; Tan und Zhan 2017). Auch die Überlegenheit der Big-Data-Analytik gegenüber bestehenden Marketing-Methoden auf den Erfolg von neuen Produkten wurde exemplarisch nachgewiesen (Aljumah et al. 2021).

Die große Menge an Daten verhindert, dass Ingenieur:innen und Designer:innen ihre Aufgaben derzeit ohne externe Hilfe effizient und effektiv erfüllen können. Etablierte Methoden wie Interviews, Kano-Modell, House-of-Quality und Fragebögen, aber auch neuere Methoden wie Lead-User oder

Customer Journey stoßen durch die Schnelllebigkeit der Produkte und die zur Verfügung stehende Datenmenge an ihre Grenzen. Eine besonders große Diskrepanz zwischen der Menge an verfügbaren Informationen und der fehlenden Methode zur Analyse (für die Produktentwicklung) besteht bei nutzergenerierten Inhalten auf Online-Handelsplattformen. Online-Handelsplattformen bieten aus mehreren Gründen ein großes Potenzial als Datenbasis bzw. "Ort-der-Suche". Zum einen ist aufgrund der immer größer werdenden Bedeutung des Online-Handels davon auszugehen, dass es kaum noch Produkte gibt, die nicht auf diesen Plattformen angeboten werden. Zum anderen bieten aktuelle Online-Handelsplattformen die Möglichkeit, dass kaufende Personen Bewertungen und Fragen zu den angebotenen Produkten verfassen können und diese Informationen frei zugänglich sind. Dieser nutzergenerierte Inhalt bietet das Potenzial, um Kundenwünsche, Trends und Schwachstellen zu erfassen und abzuleiten. In dieser Arbeit wird die Herausforderung dargestellt, wie Unternehmen datenbasierte Methoden für die Produktentwicklung nutzen, an die Konzepte von Big Data angelehnt, ohne die Anforderungen an die Datenspeicherung von Big Data zu erfüllen.

Eine vielversprechende Möglichkeit die in unstrukturierter Textform vorliegenden Informationen automatisiert auszuwerten, stellen Algorithmus-basierte Analyseverfahren aus dem Bereich der Künstlichen Intelligenz, insbesondere des sogenannten Maschinellen Lernens (machine learning) dar. Für eine Auswertung dieser Kundenbewertungen sind Kenntnisse aus der Linguistik, der künstlichen Intelligenz, der Produktentwicklung und der Interaktion zwischen Nutzer und Produkt erforderlich (Abbildung 1). Im Rahmen dieser wissenschaftlichen Arbeit wird der *Betrachtungsgegenstand* (nutzergenerierte Inhalte) mithilfe von Natural Language Processing (NLP)¹ (*Methode*) ausgewertet und für die Produktentwicklung (*Perspektive*) aufbereitet, um eine datenbasierte Entscheidungsgrundlage für aktuelle oder zukünftige Produkte zu bieten. Die einzelnen Elemente und deren Zusammenhänge werden mit dem Ziel beschrieben, diskutiert und erörtert, ein Text-Mining-Modell zur automatisierten Auswertung von nutzergenerierten Inhalten aus Online-Handelsplattformen für die Produktentwicklung zu entwickeln.

1.2 Aufbau der Arbeit

Den schematischen Aufbau dieser Arbeit und die Zusammenhänge der betrachteten Themengebiete für die Auswertung nutzergenerierter Inhalte stellt Abbildung 1 dar. Dabei sind diese vier Bestandteile von besonderer Bedeutung: **Konzept**, **Analyse**, **Synthese** und **Datenbasierte Entscheidung**. Wie diese Begriffe für diese Arbeit interpretiert werden und welchen wissenschaftlichen Stellenwert sie einnehmen wird im Folgenden erläutert.

¹ Im Folgenden wird der geläufigere englische Begriff Natural Language Processing (NLP) anstatt der deutschen Formulierung verwendet.

Im Rahmen dieser Arbeit wird für die Ingenieurwissenschaften ein vollständig neuartiges **Konzept** für die Nutzung nutzergenerierter Inhalte entwickelt. Dieser Teil der Arbeit ist als Grundlagenforschung zu verstehen. Aufbauend auf den Erkenntnissen werden bestehenden Software-Algorithmen aus dem Bereich von NLP verwendet.

Die **Analyse**, stellt für die Ingenieurwissenschaft den geringsten Neuheitsgrad dar, weil NLP in anderen Disziplinen (KI, Linguistik und Informatik) entwickelt worden ist. NLP wird als bestehende Methode für die Analyse verwendet, ohne die Funktionsweise zu verbessern oder im Kern zu verändern. Vielmehr wird NLP als Mittel zur Lösung der technischen Probleme eingesetzt.

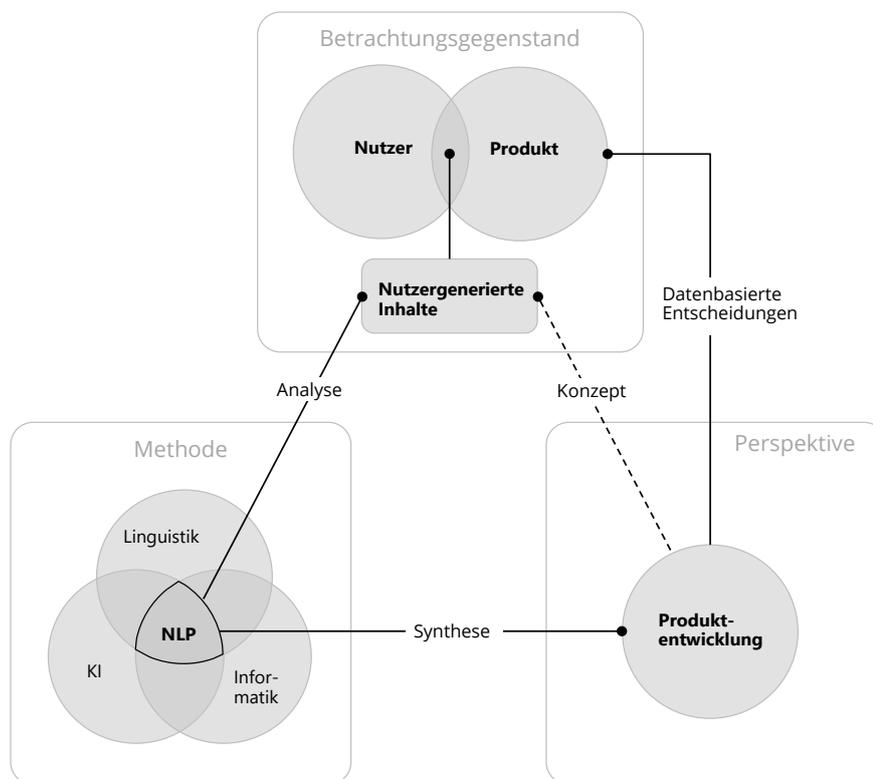


Abbildung 1: Schematischer Aufbau der Zusammenhänge und betrachteten Themengebiete für die Auswertung nutzergenerierter Inhalte für die Produktentwicklung im Rahmen dieser Arbeit

Die **Synthese** ordnet sich vom Innovationsgrad her betrachtet zwischen den beiden vorherigen Teilen dieser Arbeit ein, weil in dieser Arbeit nicht genügend verschiedenartige Datenquellen, Produktarten oder andere Unterscheidungsmerkmale untersucht und verglichen werden, um Regeln oder Gesetzmäßigkeiten abzuleiten. Vielmehr wird nur eine Produktart und eine Informationsquelle verwendet, um das Potential von NLP für die Produktentwicklung aufzuzeigen.

Der letzte Teil – die **Datenbasierte Entscheidungsgrundlage** für die Produktentwicklung – stellt den Praxisbezug des vorher wissenschaftlich hergeleiteten und gegenüber anderen wissenschaftlichen Veröffentlichungen eingeordneten, abgegrenzten oder widersprochenen Konzepten dar. Der Teil der

Arbeit ist sehr wichtig, weil die Ingenieurwissenschaft sich neben der wissenschaftlichen Erkenntnis auch am Praxisbezug der technischen Lösung misst.

Die Arbeit ist wie folgt aufgebaut: Kapitel 2 präsentiert die Ausgangssituation in der Produktentwicklung und beschreibt die notwendigen Grundlagen und Definitionen; Kapitel 3 diskutiert den Stand der Technik und Forschung zur Analyse von Online-Rezensionen und NLP-Algorithmen zur Textauswertung; Kapitel 4 zeigt den Handlungsbedarf und die Zielsetzung auf; Kapitel 5 beschreibt die Vorgehensweise zur Entwicklung der Modelllogik; Kapitel 6 beschreibt die prototypische Implementierung dieser Logik in einem Softwarecode; Kapitel 7 präsentiert die lauffähige Software anhand eines Fallbeispiels; Kapitel 8 diskutiert Schwachstellen und Einschränkungen und Kapitel 9 fasst die Ergebnisse zusammen und gibt einen Ausblick die Möglichkeiten der Optimierung und Erweiterung der entwickelten Applikation.

2 Ausgangssituation und Begriffsdefinitionen

In diesem Kapitel werden die Ausgangssituation in der Produktentwicklung und die Randbedingungen für die Entwicklung des Text-Mining-Modells erläutert. Zu diesem Zweck werden die wichtigen Perspektiven auf die Produktentwicklung beschrieben und deren Bezug zur Nutzerintegration bewertet. Diese Gegenüberstellung verdeutlicht die Relevanz der Arbeit für die technische Produktentwicklung. Anschließend wird die Nutzerintegration in Abhängigkeit der Produktkategorie dargestellt, indem die verschiedenen Produktkategorien (aus dem Marketing-Bereich) diskutiert werden. Anschließend wird die Produktkategorie ausgewählt, die für die Arbeit relevant ist. Danach werden etablierte Maßnahmen zur Nutzerintegration in den Produktentwicklungsprozess (aus dem Management-Bereich) aufgelistet und die Begebenheit nutzergenerierter Inhalte im Internet kurz erläutert. Auf Basis dieser Beschreibung wird der Bedarf nach softwarebasierten Lösungen für die Produktentwicklung hergeleitet. Abschließend werden wichtige Begriffsdefinitionen vorgestellt.

2.1 Der Produktentwicklungsprozess

Wie werden Produkte entwickelt? Diese Frage wird seit den 90er Jahren in der Wissenschaft kontrovers diskutiert. Dass dieser Trend immer noch existent ist, zeigt die steigende Anzahl der wissenschaftlichen Veröffentlichungen aus der einschlägigen Literatur aus den Management- und Ingenieurwissenschaften (vgl. Abbildung 2).

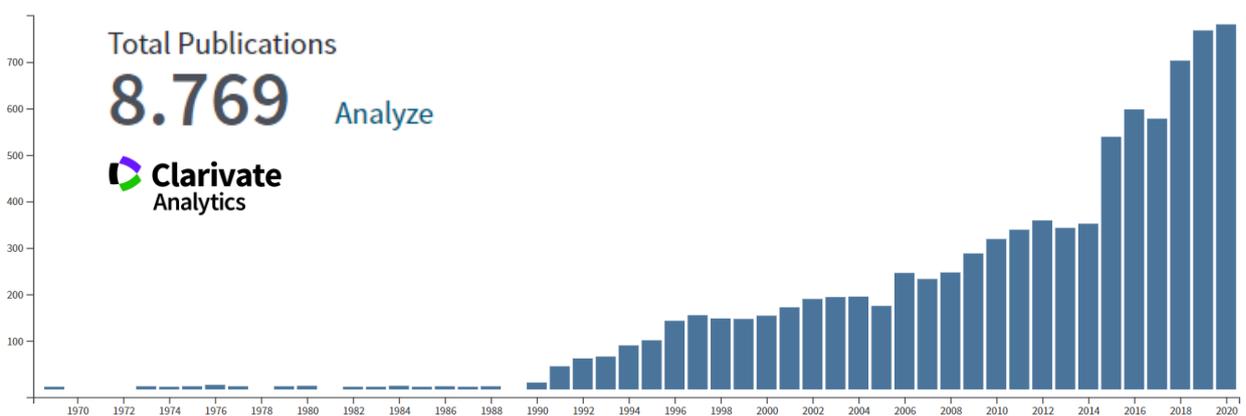


Abbildung 2: Zeitlicher Verlauf der Anzahl wissenschaftlichen Veröffentlichungen im Bereich Management und Ingenieurwesen zum Thema Produktentwicklungsprozess (*Citation Report* von WebOfScience.com)

In der ingenieurwissenschaftlichen Literatur finden sich viele Phasenmodelle, um die Tätigkeiten und Abfolgen für eine erfolgreiche Produktentwicklung zu beschreiben. Neben dieser Perspektive existieren noch zwei weitere, die ursprünglich aus der Softwareentwicklung und dem Projektmanagement entstandene, die agile Perspektive und zuletzt die unternehmensbasierte Management Perspektive. Warum ist es notwendig diese drei Perspektiven auf die Produktentwicklung im Rahmen dieser Arbeit

zu beschreiben? Die Frage scheint auf den ersten Blick berechtigt. Die Begründung für einen multip-len Ansatz und die daraus entstehenden Vorteile sollen deshalb jetzt näher erläutert werden.

Das am weitesten entwickelte Wissen zu den Wirkzusammenhängen, Parametern und Einflüssen in der Produktentwicklung ist in der aktuellen Management-Literatur zu finden. Das begründet sich durch die in dieser Domäne viel stärker vertretenen und etablierten, standardisierten, quantitativen Forschungsmethoden. Somit ist es sinnvoll, die Erkenntnisse dieser Arbeit zu synthetisieren und in eine ingenieurtechnische Perspektive zu transferieren. Der Transfer ist notwendig, da die vorliegende Arbeit Ingenieur:innen in der Produktentwicklung unterstützen soll. Für diesen Transfer muss einerseits der Produktentwicklungsprozess aus der Ingenieurperspektive betrachtet werden und andererseits mittels einer Abbildungs- oder Mapping-Logik die Management-Bestandteile in die Ingenieurs-Logik überführt werden. Die letzte – aus dem Software- und Projektmanagementbereich stammende – ist die agile Produktentwicklungsperspektive. Diese Vorgehensweise und Darstellung von Vorgängen, Prozessen und Abläufen im Entwicklungsprozess ist durch Iterationen also ständige Wiederholungen oder Verbesserungen geprägt. Dieses Vorgehen stellt aktuell in der Theorie und Praxis die modernste Vorgehensweise dar, um innovative Produkte zu entwickeln. Um diesen Trend in der vorliegenden Arbeit zu folgen, werden die Erkenntnisse aus der klassischen ingenieurwissenschaftlichen Perspektive (Wasserfall-Ansatz) in eine moderne – agile – Perspektive überführt.

Im folgenden Abschnitt werden die drei Perspektiven beschrieben und anschließend der Nutzerfokus bewertet.

2.1.1 Ingenieurwissenschaftliche Betrachtungsweise

Ingenieur:innen entwickeln und konstruieren technische Produkte und Systeme. Die Ingenieur-tätigkeit an sich und die wissenschaftliche Untersuchung hat in Deutschland eine sehr lange Tradition. Dadurch existieren vor allem in der deutschen wissenschaftlichen Literatur viele verschiedene Vorgehensmodelle. Einen sehr ausführlichen Überblick in chronologischer Reihenfolge geben Pahl et al. (2007, S. 40–42). Da die Entwicklung der Vorgehensmodelle nach dieser Zeit oftmals nicht praktisch-technisch, sondern vielmehr hochschul- oder lehrstuhlpolitisch motiviert waren, existieren in den neueren Methoden keine signifikanten Unterschiede. Daher wird im Rahmen dieser Arbeit repräsentativ das Modell nach Pahl/Beitz vorgestellt. In dem Kapitel *Der Produktentwicklungsprozess* schreiben die Autoren sinngemäß: Das hier entwickelte Modell ist für eine branchenunabhängige Konstruktionspraxis entwickelt worden, um ein allgemein anwendbares methodisches Vorgehen zu beschreiben unter Berücksichtigung der bekannten Methoden von Pahl et al. (2007, S. 204). Der Prozess wurde in der ersten Auflage von Konstruktionslehre von 1977 erstmalig von den beiden Autoren Gerhard Pahl und Wolfgang Beitz beschrieben (Pahl und Beitz 1977).

Dieser Produktentwicklungsprozess besteht aus mehreren Phasen (vgl. Abbildung 3). Die einzelnen Phasen werden nacheinander durchlaufen. Durch Iterationen ist ein „zurückspringen“ in frühere Phasen möglich. Die einzelnen Phasen sind: Klären der Aufgabe, Konzipieren, Entwerfen und Ausarbeiten. Denen übergeordnet existieren noch zwei Optimierungsphasen: Die Prinzip- und Gestaltoptimierung.

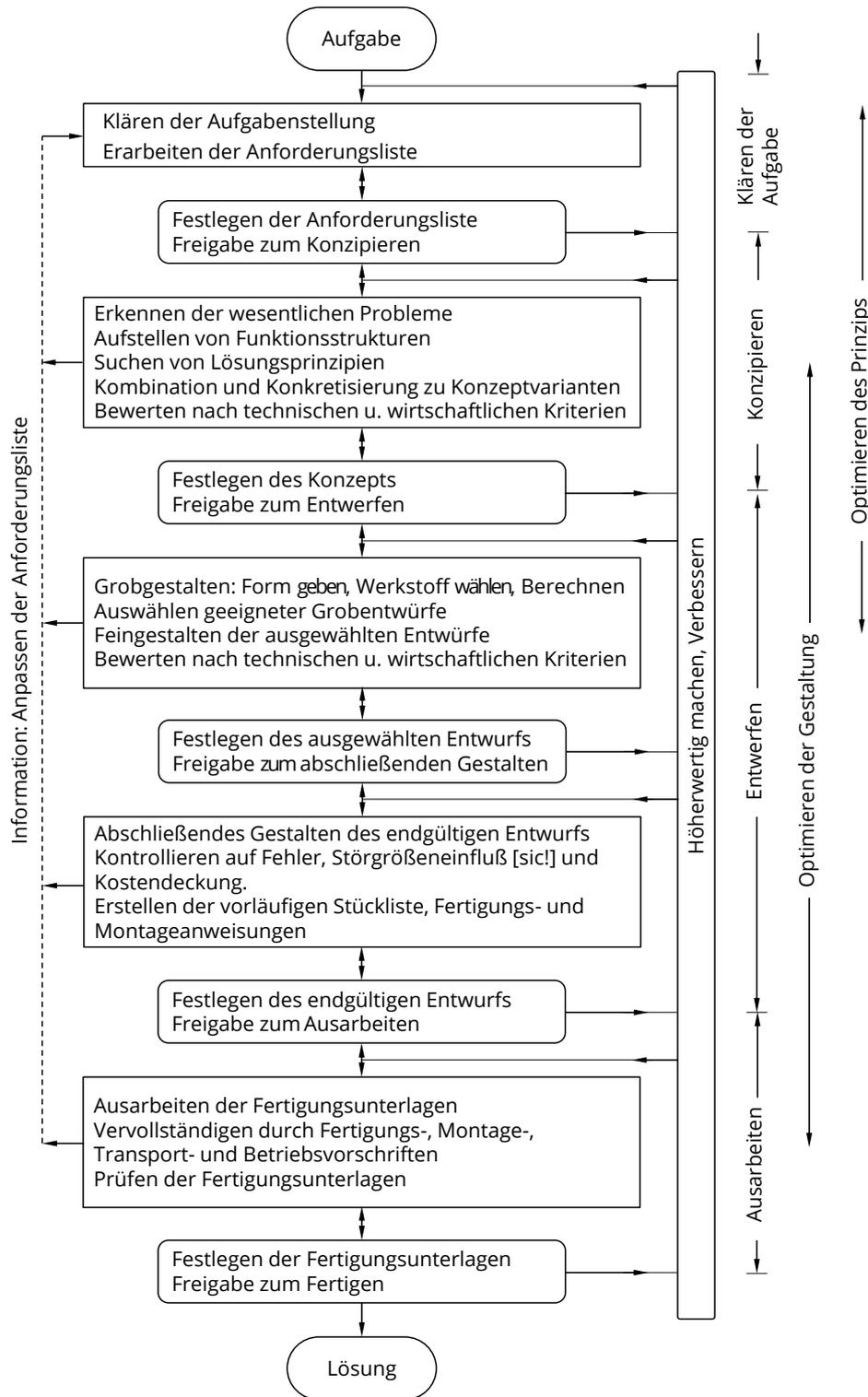


Abbildung 3: Ingenieursperspektive auf die Produktentwicklung (Pahl und Beitz 1977)

2.1.2 Management-basierte Perspektive

Für das Management sind die exakten Tätigkeiten der Ingenieur:innen während der Produktentwicklung nicht im Detail notwendig zu beschreiben. Vielmehr wird diese Betrachtungsweise in Bezug auf den Geschäftsprozess bewertet. Ein sehr etabliertes und vielfach verwendetes Framework entwickelten O'Connor und DeMartino (2006). Im folgenden Abschnitt wird dieses Framework beschrieben. Die Beschreibungen und Erklärungen sind aus (ebd.) entnommen.

Das Framework besteht aus drei aufeinanderfolgenden Phasen: **Discovery** (Entdeckung), **Incubation** (Inkubation) und **Accelaration** (Beschleunigung) (vgl. Abbildung 4). Da die deutschen Begriffe weniger geläufig sind, werden im weiteren Verlauf die englischen Begriffe verwendet.

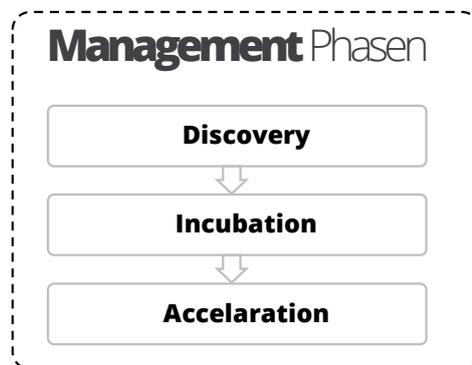


Abbildung 4: Phasen der Produktentwicklung aus Management Perspektive nach O'Connor und DeMartino (2006)

Discovery-Phase

Diese Phase stellt den Beginn der Produktentwicklung dar. Sie ist durch Maßnahmen geprägt, die Chancen und Möglichkeiten für neuartige Produkte generieren oder suchen. Typische Entdeckungsaktivitäten während dieser Phase sind: Erfindungen, interne Forschungen, Ideen- und Chancensuche, Partizipation an Offenen Innovationen (Open Innovation), Lizenzierung von Technologien oder Kapitalbeteiligung an kleineren Firmen oder Startups (Equity Investments). In der Praxis sind Firmen gleichzeitig in mehreren dieser Felder gleichzeitig aktiv.

Incubation-Phase

In dieser Phase wird aus einer Chance eine Geschäftsmodell-Idee. Dieser Transfer findet auf Basis der Ideenphase hin zu einer frühen technischen Marktexperimentierungs-Phase statt. Der Erfolg dieser Phase wird insbesondere durch die Experimentier- und Lernfähigkeit potenzieller Kunden und die Anpassung der Technologie oder der Unternehmensstrategie beeinflusst. In dieser Phase existiert die höchste Ausfallrate. Die Incubation-Phase oder auch Reifephase schließt mit einem funktionierenden und von kaufinteressierten Personen validierten Prototyp ab.

Accelaration-Phase

Ziel dieser Phase ist der Aufbau eines operativen Geschäfts. Die größte Herausforderung besteht darin, dass die Wertschöpfung aktuell nur in einer „geschützten“ Umgebung stattfinden kann – und noch

nicht alleinig (frei am Markt) überlebensfähig ist. Umsatzprognosen für das Geschäftsmodell ermöglichen Wachstumsgrenzen und Fehlentwicklungen zu identifizieren. Dafür werden die ersten Kundeninformationen genutzt, um zwischen potentiellen und qualifizierten Kunden zu unterscheiden. Eine entscheidende Besonderheit dieser Phase besteht darin, dass der Erfolg der (finanziellen) Investitionen und der Aufbau des Geschäfts am Umsatzwachstum und nicht an der Profitabilität bemessen wird.

2.1.3 Moderne Entwicklungsprozesse

Es existiert eine Vielzahl an Veröffentlichungen, Frameworks und Fallstudien zu diesem Thema. Dieser Abschnitt soll keine ganzheitliche Beschreibung moderner Entwicklungsprozesse und der damit zusammenhängenden Forschungsfragen in ihrer Gesamtheit liefern. Vielmehr soll ein erhöhtes Bewusstsein dafür aufgebaut werden, innovative Maßnahmen und neuartige Methoden für die Produktentwicklung in einem modernen Arbeitsumfeld zu berücksichtigen, um eine praxisnahe Implementierung zu ermöglichen. Auf Basis der am häufigsten zitierten veröffentlichten Literaturübersicht zu dem Suchbegriff: *agile development* (Agile Entwicklung) werden die synthetisierten Erkenntnisse hier beschrieben.

Moderne Produktentwicklung ist geprägt durch eine iterative, inkrementelle Veränderung am Produkt basierend auf empirischen Daten. Larman und Basili (2003) geben einen sehr umfangreichen Überblick über die historische Entwicklung von Software-Entwicklungs-Praktiken, die durch iteratives und inkrementelles Vorgehen bestimmt sind. Diese Analyse zeigt, dass sich schon in den 1950er Jahren eine Arbeitsweise bei der Softwareentwicklung etablierte. Kern der Entwicklungsarbeit war und ist, die Veränderungen am Produkt durch schnelle Feedback-Schleifen mit Kunden zu erproben, um frühzeitig Schwachstellen und Verbesserungen zu identifizieren. Diese Vorgehensweise ist dem klassischen Ingenieurs-Entwicklungsprozess völlig gegensätzlich. Die *schweren Entwicklungsprozesse* der Ingenieur:innen sind für die Vorgehensweisen bei der Entwicklung eines Fahrzeugs noch immer akzeptiert – für dynamischere Märkte, schnell ändernde Kundenbedürfnisse oder bei erhöhtem Konkurrenzdruck müssen *leichtere Prozesse* genutzt werden, um die Handlungen der Produktentwickler zu koordinieren. Diese Prozesse werden auch als agile Prozesse bezeichnet. Es existiert eine Vielzahl an Frameworks: Scrum, Extreme Programming (XP), Kanban, Lean Startup etc. Die systematische Literaturrecherche über empirische Untersuchungen zu agilen Softwareentwicklungsprojekte analysierte 36 Studien (von möglichen 1996) in Hinblick auf vier Themenfelder: Einführung und Anwendung, menschliche und soziale Faktoren, Wahrnehmungen zu agilen Methoden und vergleichende Studien (Dybå und Dingsøyr 2008, S. 833). Die Autoren kommen zu folgenden Ergebnissen bezüglich agiler Entwicklungsmethoden (Auswahl): Agile Methoden können einen positiven Einfluss auf die Arbeit

in der Produktentwicklung haben. Eine hybride Variante, bei der klassische Methoden mit agilen Maßnahmen kombiniert werden, ist oft hilfreich. Für kleine Teams sind die Methoden geeigneter als für große Projekte. Die Zusammenarbeit mit Kunden wird verbessert. (Dybå und Dingsøyr 2008, S. 850)

Für die praktische Umsetzung moderner Entwicklungsprozesse wird vielmals das Agile Manifest als theoretische Grundlage genutzt (Agile manifesto 2001).

Die zwölf universellen Prinzipien für eine agilere Arbeitsweise aus dem *Agilen Manifest* bilden die Grundlage für das bekannteste agile Framework Scrum (Schwaber und Beedle 2002). Das Framework besteht aus Verantwortlichkeiten oder Rollen, Stakeholdern, Sprints, Ereignissen und verschiedenen Artefakten (Gegenstände im übertragenen – nicht physischem - Sinne) (Schwaber und Sutherland 2011).

2.1.4 Bewertung der Perspektiven in Bezug auf den Nutzerfokus

Die drei vorgestellten Perspektiven auf den Produktentwicklungsprozess und die damit verbundenen Tätigkeiten zeigen deutliche Unterschiede. Neben vielen anderen Aspekten, stellt die Rolle des Nutzers ein besonders kritisches Unterscheidungsmerkmal dar. In der ingenieurwissenschaftlichen Perspektive wird der Nutzer nur in den frühen Phasen der Produktentwicklung (Anforderungsermittlung in der Phase *Klären der Aufgabe*) stark berücksichtigt. Eine größere Rolle nimmt der Nutzer in der Management-basierten Perspektive ein. Bei ihr ist der Nutzer in allen drei Phasen ein sehr wichtiger Faktor um den (ökonomischen) Erfolg des Produktes zu maximieren. Den höchsten Stellenwert – mit großem Abstand – wird der nutzenden Person in der agilen Produktentwicklung zugesprochen. Vielmehr ist die Kundschaft hier zentraler Gegenstand, welche als ultimativer und allmächtige Mitspielende während der gesamten Produktentwicklung agiert. Tabelle 1 stellt die Perspektiven und die Bewertung des Nutzerfokus vereinfacht gegenüber.

Tabelle 1: Vergleich der Perspektiven und Bewertung in Hinblick auf den Nutzerfokus

Perspektive	Produktentwicklungsphasen oder Tätigkeiten			Nutzerinteraktion
Ingenieurs-basierte	Klären der Aufgabe	Konzipieren	Entwerfen und Ausarbeiten	
Management-basierte	Discovery	Incubation	Accelaration	
Agile	Iterieren , Inkrementell, Empirisch			

Die Auswertung zeigt, dass die agile Perspektive die größte Nutzerinteraktion aufweist und dass bei der Ingenieurs-basierten Perspektive mit dem Nutzer am wenigsten interagiert wird. Wenn aber die technische Umsetzung durch die Ingenieur:innen den größten Einfluss auf die Produktgestaltung hat, dann besteht hier ein besonders großer Konflikt. Wie stark dieser Konflikt ist, ist abhängig von der Produktkategorie. Im folgenden Kapitel werden die verschiedenen Produktkategorien in Bezug auf die kaufende Person und den Kaufentscheidungsprozess beschrieben.

2.2 Produktkategorien

Was ist ein Produkt? Für diese Arbeit wird angenommen, dass ein Produkt sowohl eine Software, eine Dienstleistung, ein klassisches physisches Produkt oder eine Kombination dieser drei Kategorien sein kann. Außerdem ist es notwendig, dass Produkte auf der Grundlage der Nutzer differenziert werden. Es existieren zwei Arten von Nutzern: *Verbraucher* und *Unternehmen*. Geläufiger sind die Begriffe: *B2C-Produkte* (Business-to-Customer für Verbraucherprodukte) und *B2B-Produkte* (Business-to-Business für Geschäftsprodukte).

Welche B2C-Produkte gibt es und wie unterscheiden sie sich voneinander? Eine etablierte und praxiserprobte Kategorisierung aus dem Marketingbereich unterteilt Verbraucherprodukte in vier Gruppen: **Convenience**, **Shopping**, **Speciality** und **Unsought** Produkte. Diese werden im Folgenden ausführlicher beschrieben und mit Beispielen versehen. Da diese englischen Begriffe oft auch im Deutschen verwendet werden, wird hier keine deutsche Übersetzung angegeben.

Convenience Produkte sind günstige Konsumgüter. Im Vergleich zu den anderen 3 Gruppen werden diese am häufigsten gekauft. Beispiele sind: Waschmittel, Lebensmittel, Zeitschriften etc.

Shopping Produkte sind teurere Konsumgüter, die weniger oft, aber mit mehr Aufwand beim Kaufprozess (Informationsbeschaffung) gekauft werden. Fahrräder, Möbel und Computer sind klassische Vertreter dieser Produktgruppe.

Speciality Produkte sind hochpreisig und haben besondere Produkteigenschaften. Der Aufwand zum Erwerb dieser Produkte verlangt besondere Anstrengungen. Limitierte Sneaker, Designerkleidung und Luxusuhren sind Beispiele dafür. Oftmals werden diese Produkte weniger stark verglichen.

Unsought Produkte sind Konsumgüter, von denen die Kundschaft entweder nicht weiß oder nicht weiß, dass er sie braucht. Typische Beispiele für diese Produktgruppe sind Hochwasserversicherungen, Verlobungsringe und Bestattungsdienstleistungen.

Die Klassifizierung basiert auf den Merkmalen: Produkteigenschaften, Preis und Kaufverhalten der Kundschaft. Von besonderem Interesse ist die Informationsbeschaffung während des Kaufprozesses. Während dieser Phase unterscheiden Korgaonkar et al. (2006) drei Arten von Konsumgütern: **Suchprodukt**, **Erfahrungsprodukt** und **Vertrauensprodukt**. Diese Produktkategorien werden in der Veröffentlichung (ebd.) näher beschrieben:

Suchprodukte sind Produkte bei denen die Produkteigenschaften vor dem Kauf festgestellt werden können. Und das alleine durch die bloße Beschreibung entschieden werden kann, ob das Produkt gekauft wird oder nicht.

Erfahrungsprodukte sind Produkte bei denen ohne persönliche Erfahrung keine vollständige Information über wichtige Merkmale des Produktes bekannt sind. Im Gegensatz zu den Suchprodukten ist bei den Erfahrungsprodukten eine tatsächliche Erfahrung (vor dem Kauf) erforderlich, um sich von

ihren Produkteigenschaften² zu überzeugen. Je nach Aufwand der Informationsbeschaffung existieren zwei Klassen von Erfahrungsprodukten. Dabei ist bei den Erfahrungsprodukten der 2. Klasse die Informationsbeschaffung kostspieliger oder schwieriger als der Erwerb der tatsächlichen Produkterfahrung.

Vertrauensprodukte sind Produkte bei denen die Bewertung der Produktmerkmale weder vor noch nach dem Gebrauch des Produktes möglich sind. Medizinische Beratung oder auch Anti-Falten-Cremes stellen typische Beispiele für diese Produktgruppe dar.

Auf der Grundlage der vorgestellten Definitionen folgen für die weitere Arbeit diese Einschränkungen. Es werden nur **B2C-Produkte**, insbesondere **Shopping-Produkte** die unter die Kategorie der **Erfahrungsprodukte** fallen berücksichtigt. Der Grund für diese Entscheidung war, dass das Thema (*nutzergenerierte Informationen*) sowohl für den Ersteller (*Nutzer*) als auch für den Leser (*potenziell kaufbereite Person*) relevant ist und daher ein größeres Potenzial für nützliche Informationen hat.

2.3 Die Integration des Nutzers in den Produktentwicklungsprozess

Welche Rollen spielen Nutzer in der Produktentwicklung? Übereinstimmung besteht darüber, dass Nutzer eine externe Informationsquelle darstellen. Die Wirkung auf den Produktentwicklungsprozess, das Zusammenspiel zwischen Kunden und Firma und die dafür notwendigen Voraussetzungen werden in der aktuellen wissenschaftlichen Literatur kontrovers diskutiert. Eine sehr umfängliche und in der Forschungsgemeinschaft etablierte Sichtweise der unterschiedlichen Arten der Kundenintegration beschreiben Schweitzer et al. (2020). Aufbauend auf einer systematischen Literaturrecherche analysieren die Autoren 153 empirische Untersuchungen und synthetisieren sechs verschiedene Arten („Perspektiven“) der Kundenintegration (vgl. Abbildung 5).

Hauptsächlich bestimmen zwei Faktoren die Kundenintegration: Erstens die Art der Informationen und zweitens die Weiterverarbeitung innerhalb des Unternehmens (Schweitzer et al. 2020, S. 822). Die zwei Faktoren lassen sich durch die folgenden Fragen ebenfalls erörtern:

Was soll mittels der Kundschaft herausgefunden werden?

Wie werden die gesuchten Informationen im Unternehmen weiterverarbeitet?

Auf Basis dieser zwei Faktoren wird hier die Einteilung in Kunden- und Unternehmensbasiert eingeführt. Die nachfolgenden Absätze orientieren sich an der Arbeit von Schweitzer et al. (2020).

Da der Fokus dieser Arbeit in der Untersuchung der Nutzbarmachung der Kundeninformationen liegt und nicht in der Untersuchung der organisatorischen oder unternehmensinternen Voraussetzungen, wird im Weiteren nur die Kategorie *Kundenbasiert* näher betrachtet.

² In der ursprünglichen Beschreibung wird hier nur der Begriff Qualität verwendet. Nach Ansicht des Autors der vorliegenden Arbeit kann die Aussage um den Begriff der Produkteigenschaften erweitert werden und wurde daher ersetzt.



Abbildung 5: Die verschiedenen Arten der Kundenintegration nach Schweitzer et al. (2020) (Quelle: Eigene Darstellung)

Wie kann die Kundschaft in den Entwicklungsprozess eingebunden werden? Neben der *direkten* und *indirekten Kundeneinbindung* gibt es die Möglichkeit der *kundenbasierten Ideenbewertung*. Bei der *direkten Kundeneinbindung* werden beispielweise Ideenwettbewerbe, Lead User Workshops, Kreativitätssessions oder informelle Austausche mit Kunden durchgeführt. Existiert kein direkter Austausch zwischen der Firma und der Kundschaft, so wird das als *indirekte Kundeneinbindung* bezeichnet. Das Beobachten aktueller Kunden und die Untersuchung von Nutzungsschwierigkeiten mit bestehenden Produkten sind typische Beispiele dafür. Bei der dritten Kategorie bewertet die nutzende Person potenzielle Konzeptideen.

Der positive oder negative Einfluss dieser drei Möglichkeiten der Kundenintegration wurde in der Studie in Bezug auf die Produktentwicklungsphasen (Explorations-, Inkubations- und Wachstumsphase), der Nutzerarten (normale, potentielle, aktuelle, domainspezifische Nutzer) und des Produktinnovationsgrades (bahnbrechend oder technologisch neu) analysiert. Das Ergebnis zeigt ein differenziertes Bild. Es zeigt sich jedoch, dass die Kundenintegration in den meisten Fällen einen positiven Einfluss auf die Produktentwicklung hat.

Die Frage, welche Maßnahmen der *kundenbasierten* Integration zielführend für die Produktentwicklung sind, können Schweitzer et al. (2020) nicht abschließend beantworten – dennoch können sie mit dem systemischen Literaturreview einen positiven Trend für die Kundenintegration im Entwicklungsprozess nachweisen.

Die Herausforderung besteht darin, die kundenbasierten Maßnahmen in den Produktentwicklungsprozess zu integrieren. Verschärft wird diese Herausforderung durch externe Faktoren wie den immer kürzeren Lebenszyklus von Produkten, die sich rasch ändernden Kundenbedürfnisse und nicht zuletzt den durch die zunehmende Globalisierung der Märkte immer stärker werdenden Innovationsdruck auf das Unternehmen. Da diese Faktoren in Zukunft noch mehr an Einfluss gewinnen werden, müssen die Unternehmen neue und bessere Wege finden, um Informationen von den Kunden zu erhalten. Eine ständig wachsende Quelle für solche Informationen ist das Internet, insbesondere Social-Media-Plattformen. Das nächste Kapitel beschreibt die aktuelle Situation von nutzergenerierten Inhalten im Internet und zeigt das Potenzial für die Produktentwicklung auf, um nutzergenerierte Inhalte aus dem Internet nutzbar zu machen.

2.4 Datenbasierte Ansätze zur Verarbeitung von Nutzerinformationen für die Produktentwicklung

Es existieren viele Ansätze zur Aufbereitung von Nutzerinformationen für die Produktentwicklung. Eine sehr umfassende (wenn auch nicht erschöpfende) Liste der wichtigsten datenbasierten Evaluierungsmethoden und Analyseziele verschiedener (Nutzer-)Informationen findet sich bei Zhan et al. (2019). Das folgende Kapitel basiert auf dieser Veröffentlichung und fasst die wichtigsten Erkenntnisse für vorliegende Arbeit zusammen.

Als **Informationsgrundlage** dienen Informationen und Wissen über Kundenbedürfnisse, Kaufverhalten und Produkteigenschaften (vgl. Abbildung 6). Die Informationen über die Produktmerkmale wurden näher untersucht. Diese können als mögliche Gestaltungspräferenzen, Produktkombinationen oder Produktvarianten, Produktanforderungen und neue Produktkonzepte weiter detailliert beschrieben werden. Die Frage, wie die Informationen bewertet werden und ob sie für die Produktentwicklung nützlich sind, bleibt unbeantwortet. Darüber hinaus werden die Informationen in Bezug auf die Kundschaft charakterisiert. Die Informationen können „vom“, „für“ oder „über“ die Person sein. Oftmals wird das Analyseziel als Bewertungsgrundlage für die Informationsgrundlage genutzt.

Die Informationsarten werden untersucht um folgenden Erkenntnisse (**Analyseziele**) für die Produktentwicklung zu gewinnen: i) Neuentwicklungen oder Verbesserungen von Produkten, Produktgenerationen, -linien, -familien, -plattformen oder Herstellungsprozessen und ii) um Abhängigkeiten zwischen Kundschaft, Produkt und Herstellungsprozess zu identifizieren. Das Analyseergebnis kann in verschiedenen Phasen des Entwicklungsprozesses nützliche Informationen liefern.

Informationsgrundlagen	Analyseziele	Datenbasierte Methoden
Kundenbedürfnisse	Neuentwicklungen	Assoziationsregeln Neuronale Netze Clustering Conjoint-Analyse Genetische Algorithmen Entscheidungsbäume Fuzzy-Entscheidungsbäume Text Mining Themenanalyse Szenarioansätze
Kaufverhalten	Verbesserungen - Produkt - Produktgeneration, -linie, -plattform - Herstellungsverfahren	<i>Kombinationen aus</i> Assoziationsregeln & Neuronale Netze Assoziationsregeln & Clustering Conjoint-Analyse & Genetische Algorithmen
Produkteigenschaften - Gestaltungspräferenzen - Produktvarianten - Produktanforderungen - neue Produktkonzepte	Abhängigkeiten und Wirkzusammenhänge identifizieren - Kunde, Produkt, Herstellungsverfahren	

Abbildung 6: Nutzerinformationen im Produktentwicklungskontext: Informationsgrundlage, Analyseziele und datenbasierte Methoden (eigene Darstellung)

Für die Auswertung werden die folgenden **datenbasierten Methoden** verwendet: Assoziationsregeln, Neuronale Netze, Clustering, Conjoint-Analyse, Genetische Algorithmen, Entscheidungsbäume, Fuzzy-Entscheidungsbäume, Text Mining, Themenanalyse und Szenarioansätze und die Kombinationen Assoziationsregeln und Neuronale Netze, Assoziationsregeln und Clustering und Conjoint-Analyse und Genetische Algorithmen.

Zhan et al. (2019) kommen zu dem Ergebnis, dass durch die verschiedenen Ansätze das stillschweigende zerstreute Kundenwissen extrahiert, gesammelt und in Bezug auf die Produkteigenschaften analysiert werden kann und somit ein Wettbewerbsvorteil geschaffen wird. Auch die vorliegende Arbeit folgt dieser Argumentation. Aber im Gegensatz zum vorgeschlagenen Datenerhebungsverfahren der Autoren (webbasierte Umfrage und Fragebogen) wird in vorliegender Arbeit ein anderer Ansatz zur Datenerhebung verwendet. Diese Arbeit nutzt als Datengrundlage bereits vorhandene nutzergenerierte Inhalte im Internet.

2.5 Nutzergenerierte Inhalte im Internet

Es existieren aktuell schätzungsweise 4,66 Milliarden aktive Internetnutzer weltweit - 59,5 Prozent der Weltbevölkerung (Statista 2021a). Mit der Möglichkeit des Web 2.0 können diese Internetnutzer Informationen konsumieren, teilen und erzeugen. Infolgedessen wird das Internet zu einer immer größeren und wichtigeren Informationsquelle. Wie groß diese Informationsquelle werden kann, hat Seagate für das Jahr 2025 berechnet. In der Analyse wurde vorausgesagt, dass das Volumen der jährlich weltweit erzeugten digitalen Daten 175 Zettabyte³ erreichen wird (Statista 2021b). Die Top 20 Webseiten in Deutschland stellt Abbildung 7 dar.

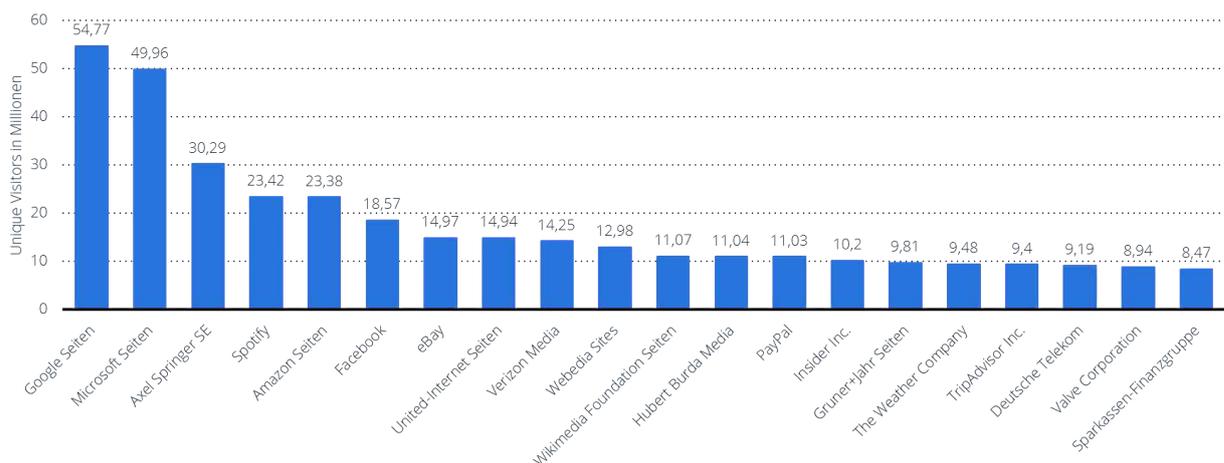


Abbildung 7: Top 20 Webseiten in Deutschland nach der Anzahl der Unique Visitors im Juni 2021 (in Millionen) (Statista 2021c)

³ 1 Zettabyte = 1 Billion Gigabyte = 1.000.000.000.000 GB

Neben den Webseiten von Google und Microsoft, die eher zum Konsumieren genutzt werden, können Webseiten wie Facebook, Amazon und eBay auch zum Teilen und Erzeugen von Daten verwendet werden. Daraus lässt sich schließen, dass von den ständig wachsenden Datenmengen in der Zukunft auch ein erheblicher Teil von den Nutzern des Internets erzeugt werden wird.

Was sind nutzergenerierte Inhalte (user generated content – UGC)? Eine einheitliche Definition ist in der aktuellen Literatur nicht zu finden. Für die vorliegende Untersuchung wird die Beschreibung von Naab und Sehl (2017) verwendet. Die beiden Autoren charakterisieren UGC durch drei Kriterien (Naab und Sehl 2017, S. 1258):

UGC weist ein gewisses Maß an Eigenleistung auf,

UGC muss veröffentlicht sein und

UGC wird nicht in einem beruflichen oder professionelleren Kontext erstellt.

Anhand dieser Kriterien ist es nun möglich Inhalte in UGC und Nicht-UGC zu klassifizieren. Wird die immer größer werdende Professionalisierung auf Social-Media-Plattformen wie Instagram berücksichtigt, so wird in naher Zukunft – wenn nicht schon heute – das dritte Kriterium immer schwächer. Welche verschiedenen Arten und Formen es von UGC gibt und wo diese veröffentlicht und erzeugt werden, ist aktuell nur in Teilen beschrieben und auch die Abgrenzungen sind oftmals nicht ganz konsistent. Einigkeit besteht über die Formen: Fotos, Videos, Ratings, Reviews und Blogs (Mendes-Filho und Tan 2008). Parallel dazu existieren Online-Community-Arten: Foren, unternehmensinterne Netzwerke, soziale Netzwerke, Open-Source und Kreativitäts-Communitys, Wikis, Frage-Antworten-Websites. Erste Erkenntnisse und Untersuchungen wurden in dem Feld schon durchgeführt und kamen zu dem Ergebnis, dass die Formen und deren Anwendung stark Community-spezifisch sind (Malinen 2015, S. 231).

2.6 Formen und Datenbanken nutzergenerierter Inhalte

Übereinkunft besteht darin, dass schon heute die große Anzahl der Daten nicht mehr händisch organisiert werden kann. Softwarebasierte Auswertung großer digitaler Datenmengen stellt eine adäquate Lösungsstrategie dar, um aus den vorhandenen Daten Informationen zu extrahieren. Aufbauend auf den Arten der verschiedenen nutzergenerierten Inhalte⁴, müssen die Softwarelösungen entweder **Bilder**, **Videos** oder **Sprache** verarbeiten. Dabei lässt sich Sprache auch nochmal in geschriebene und gesprochene Sprache unterteilen.

Für eine softwarebasierte Auswertung von **Videos** bieten Youtube-Videos eine fast unbegrenzte Datengrundlage. So schreibt Susan Wojcicki (CEO von Youtube), dass auf Youtube.com jede Minute 500h Videomaterial hochgeladen werden. Zusätzlich erweitern Videos von TikTok oder Instagram

⁴ Nutzerverhalten stellt ebenfalls eine Art von „nutzergenerierten Inhalten“ dar, wird aber hier nicht weiter berücksichtigt, da es nicht in die Definition von UGC fällt.

potenzielles Analysematerial. Neben Videos existieren sowohl bei der Suchmaschine Google, als auch bei den Social Media Plattformen wie Facebook, Instagram, Pinterest, Flickr etc. eine riesige Menge an zu analysierenden **Bildern**. Um diese Daten (Videos und Bilder) nutzbar zu machen, sind spezielle Softwarebibliotheken und –lösungen notwendig. Camastra und Vinciarelli (2015) geben einen sehr umfangreichen und gut beschriebenen Überblick aktueller Softwarelösungen für Audio-, Video- und Bildanalysen.

Die letzte Kategorie, **Sprache**, insbesondere Schriftsprache, stellt ebenfalls einen riesigen Datenpool dar. Vor allem durch Chat- und Kommentarfunktionen auf Online-Plattformen bieten nutzergenerierte Texte eine sehr vielseitige Analysemöglichkeit. Twitter, Whatsapp, Reddit, Booking.com aber auch Amazon, eBay oder GoogleMaps bieten kontextsensitive, hochwertige Informationen über Nutzer(meinungen). Die softwarebasierte Auswertung der Texte geschieht mithilfe Natural Language Processing (NLP) Algorithmen. Es existieren eine Vielzahl an Grundlagenbüchern und Einführungslektüre zu diesem Thema (Jurafsky und Martin 2020a; Bird et al. 2009; Vajjala et al. 2020).

2.7 Zusammenfassung der beschriebenen Ausgangssituation

Ingenieur:innen stehen heute vor der Herausforderung, die kürzeren Produktlebenszyklen und die sich schnell verändernden Märkte ressourcenschonend in der Produktentwicklung zu berücksichtigen. Konkret bedeutet dies, dass in kürzerer Zeit ein besseres Verständnis für die Kundschaft entwickelt werden muss, um die Entwicklungsaktivitäten und -entscheidungen an diese Anforderungen anzupassen. Dies ist auch der Grund für den Trend, dass sich der Entwicklungsprozess von einem phasenorientierten zu einem kundenorientierten Ansatz entwickelt. Kundenintegration kann manchmal durch direkte oder indirekte Kundenbeteiligung oder durch kundenbasierte Ideenbewertung erreicht werden. Darüber hinaus bieten datenbasierte Ansätze zur Verarbeitung von Nutzerinformationen ein enormes Potenzial für die Entwicklung eines verbesserten Kundenverständnisses. Obwohl die notwendigen Informationen im Internet verfügbar sind, ist die Menge an Informationen und die Erfassung und Auswertung mit Hilfe der derzeit etablierten Werkzeuge aus dem Entwicklungsprozess nicht möglich. Ein besseres Kundenverständnis kann durch die Analyse von Informationen über Kundenbedürfnisse, Kaufverhalten und Produkteigenschaften erreicht werden. Diese Informationen können dann zur Entwicklung neuer Produkte oder zur Verbesserung von Produkten genutzt werden, die auf diese Merkmale ausgerichtet sind.

In Hinblick auf die Aufgaben der Produktentwicklung, den verschiedenen Fachbereichen, den Herausforderungen und den immer größer werdenden Datenmengen ist eine Softwarelösung unabdingbar. Kommentare und Bewertungen im Internet zu Produkten oder Dienstleistungen für den B2C-Bereich, insbesondere Shopping-Produkte in der Kategorie Erfahrungsprodukte, bieten ein großes Potential mithilfe von NLP nutzergenerierte Inhalte auszuwerten und zusätzliche Kundeninformationen zu erhalten und diese für die Produktentwicklung zu nutzen. Mit Hilfe eines geeigneten

Lösungsansatzes können die Herausforderungen in der Produktentwicklung durch dieses Potential zwar nicht vollständig gelöst werden, aber der Ansatz bietet die Möglichkeit in den frühen Phasen der Produktentwicklung, insbesondere bei der Beschaffung und Synthese von Informationen signifikant zu unterstützen. Basierend auf der Auflistung der datenbasierten Ansätzen zur Verarbeitung von Nutzerinformationen (vgl. Abbildung 6) folgt für diese Untersuchung nachstehender reduzierter Betrachtungsgegenstand (vgl. Abbildung 8).

Informationsgrundlagen	Analyseziele	Datenbasierte Methoden
Kundenbedürfnisse	Verbesserungen - Produkt	Text Mining / NLP
Kaufverhalten (implizit)	Abhängigkeiten identifizieren - Kunde & Produkt	
Produkteigenschaften - Produkthanforderungen		

Abbildung 8: Verwendeter Ansatz zur Verarbeitung von Kundeninformationen (basierend auf Abbildung 6)

Als Informationsgrundlage dienen Kundenrezensionen auf Online-Handelsplattformen (Kaufverhalten), die die Kundenbedürfnisse beinhalten. Das Hauptziel ist die Analyse der Produkteigenschaften und -anforderungen, das Produkt zu verbessern und die Abhängigkeiten zwischen der Kundschaft und dem Produkt zu ermitteln. Die dafür gewählte Methode ist Text Mining bzw. NLP.

Abschließend für die Beschreibung der Ausgangssituation werden im nächsten Abschnitt notwendige Begriffsdefinitionen eingeführt, um daran anschließend den Stand der Technik und Wissenschaft für vorliegende Arbeit zu beschreiben.

2.8 Begriffsdefinitionen

Produktarchitektur und Produktdatenmodell

Im Laufe des Produktentwicklungsprozesses werden eine Vielzahl an Informationen zum Produkt generiert. Simulationsergebnisse, Stücklisten, Kostenkalkulationen oder ganze Fabrikplanungen können Teil einer Produktentwicklung sein. Alle diese Informationen abbilden zu wollen, stellt die aktuelle Wissenschaft und Technik vor (noch) ungelöste Herausforderungen. Model-based Systems Engineering (MBSE) stellt eine sehr vielversprechende Möglichkeit dar, um diese heterogenen Daten miteinander rechnerverarbeitend zu verknüpfen.

Im Rahmen dieser Untersuchung wird das Produktdatenmodell als eine existierende und allwissende Datenbasis angenommen. Diese Annahme ignoriert die Komplexität und die Heterogenität der Daten zugunsten einer für diese Arbeit einfach zu bearbeitenden Wissensquelle. Teile dieser Wissensquelle

werden konkretisiert und detailliert im Laufe dieser Arbeit beschrieben, insbesondere die Produktarchitektur und die nutzergenerierten Inhalte.

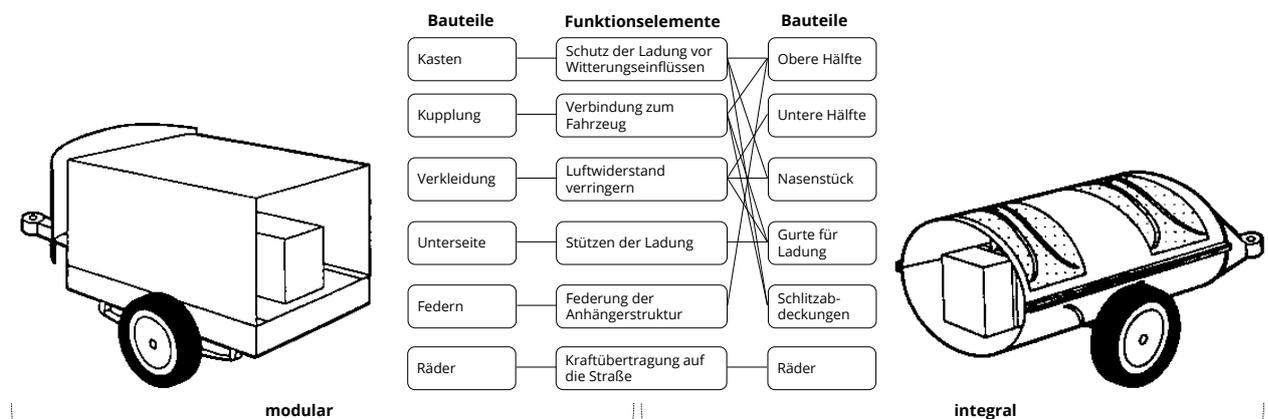


Abbildung 9: Zwei Beispiele für Produktarchitekturen (angelehnt an Ulrich 1995, S. 6)

Die Produktarchitektur (PA) stellt einen Teil dieses Produktdatenmodells dar (vgl. Abbildung 9). Sie ist ein Visualisierungsschema, das die Funktion eines Produktes den physischen Komponenten zuordnet. Ulrich (1995) beschreibt in seinem Aufsatz *The role of product architecture* die Bedeutung dieses Schemas für die verschiedenen Disziplinen, darunter Softwaretechnik, Designtheorie, Management und Produktentwicklungsmanagement. In Bezug zu den Beschreibungen von Pahl und Beitz (1977), insbesondere der Funktionsstruktur eines Produktes, erweitert Ulrich die Produktarchitektur um die Bauteile (physischen Komponenten) und deren Topologie. Diese Beschreibung bietet die Möglichkeit auf einfache Art und Weise die Komplexität der heutigen Produkte unter Berücksichtigung der wichtigen Merkmale (Funktion und Bauteile) lösungs- und domänenneutral darzustellen. Eine sehr abstrahierte Form zweier PA stellt Abbildung 9 dar. Diese Darstellungsform besteht aus der Funktionsstruktur, Produktstruktur und den Verbindungen zueinander.

In dieser Untersuchung wird die Produktarchitektur in reduzierter Form verwendet, um die Produktmerkmale zu beschreiben und zueinander in Beziehung zu setzen. Damit soll ein Rahmen geschaffen werden, in dem nicht nur Ingenieur:innen Informationen erfassen und verarbeiten können; vielmehr stellt das Verfahren der Abstraktion des Produktes in dieser Form auch eine Möglichkeit dar, Informationen domänenübergreifend und ohne großen Aufwand darzustellen.

Nutzende vs. Kundschaft

Nutzende und Kundschaft sind nicht identisch. Obschon es möglich ist, dass eine Person sowohl Nutzende als auch zur Kundschaft gehört. Die beiden Begriffe unterscheiden sich durch ihren Kontext (Tabelle 2). Ist die Person im Nutzungskontext, so wird diese Person als Nutzende bezeichnet. Ist die Person jedoch im Kaufkontext, so wird sie als Teil der Kundschaft gesehen.

Das folgende Beispiel soll den Unterschied zwischen den Begriffen mit einem fiktiven Szenario für ein Konsumgüterprodukt verdeutlichen. Das Szenario beschreibt eine private Person, die eine Kaffeemaschine für den Heimgebrauch benötigt.

Tabelle 2: Unterscheidung zwischen Nutzende und Kundschaft in Abhängigkeit der Intention

Unterscheidung Nutzende und Kundschaft		
	Kaufen	Nutzen
Kundschaft	X	
Nutzende		X

Mit dem Ziel dieses Produkt zu kaufen, wird sie im Fachhandel verschiedene Produkte vergleichen und eine Kaufentscheidung treffen. In dieser Phase ist die Person in der Kundenrolle. Gleichzeitig interagiert die Person mit dem Produkt – fasst es an, drückt Knöpfe. Es ist sogar denkbar, dass eine reale Produktnutzung während des Kaufprozesses stattfindet (z.B. bei Kaffeemaschinenhersteller eine typische Verkaufsstrategie). Das Ziel der Person bleibt der Kauf des Produktes. Sie wird somit als Teil der Kundschaft definiert. Nachdem die Kaffeemaschine gekauft wurde und im Haushalt verwendet wird, stellt jede Person, die dort einen Kaffee zubereitet, einen Nutzenden dar. Gleich ob es die kaufende Person des Produktes war oder nicht.

Das Beispiel soll verdeutlichen, dass sich insbesondere bei Konsumgütern die beiden Rollen oft überschneiden und dass die Intention der Person ihre Rolle definiert. Im Gegensatz dazu stehen B2B (Business-to-Business) Geschäftsbeziehungen. Bei diesen Geschäftsbeziehungen werden die Produkte an Unternehmen (und nicht an Privatpersonen) verkauft. Hier sind die Kaufenden selten auch gleichzeitig die Nutzenden.

Diese Forschungsarbeit betrachtet ausschließlich **Konsumgüter** und verwendet die Begriffe entsprechend der **Zielabsicht der Personen** in der betrachteten Phase. Der Schwerpunkt der Arbeit liegt auf der **Nutzungsabsicht**, da die Interaktion mit dem Produkt aus technischer Sicht relevant ist. Der Fokus auf der Kaufphase liegt in der Verantwortung anderer Disziplinen.

Produktrezensionen

Der Aufbau von Kundenrezensionen ist bei vielen Online-Plattformen ähnlich. So bestehen Rezensionen oder Reviews (diese Begriffe werden hier synonym verwendet) aus: Autor:in, Titel, Inhalt, Anzahl vergebener Sterne, Ort und Zeit der Erstellung und assoziiertes Produkt. Zum besseren Verständnis der späteren Text-Analyse, werden noch weitere Bestandteile von Rezensionen auf Online-Plattformen eingeführt (vgl. Abbildung 10). Dieses Hierarchie-Modell besteht aus vier Ebenen: (1) **Rezension**, (2) **Produkt**-Rezensions-Sammlung, (3) **Kategorie**-Rezensions-Sammlung und (4) **Portfolio**-Rezensions-Sammlung. Die Verbindungen zwischen diesen Ebenen basiert auf dem Aufbau von Online-Handels-Plattformen. Zusätzlich besteht jeder einzelne Rezensionstext aus Worten und diese Worte bilden Wortgruppen⁵.

⁵ Das ist eine grobe Verallgemeinerung und stellt keine ausführliche sprachwissenschaftliche Definition der Textbaufbauten dar. Für den weiteren Verlauf der Analyse werden die grammatikalischen und semantischen Wortkombinationen nicht näher betrachtet.

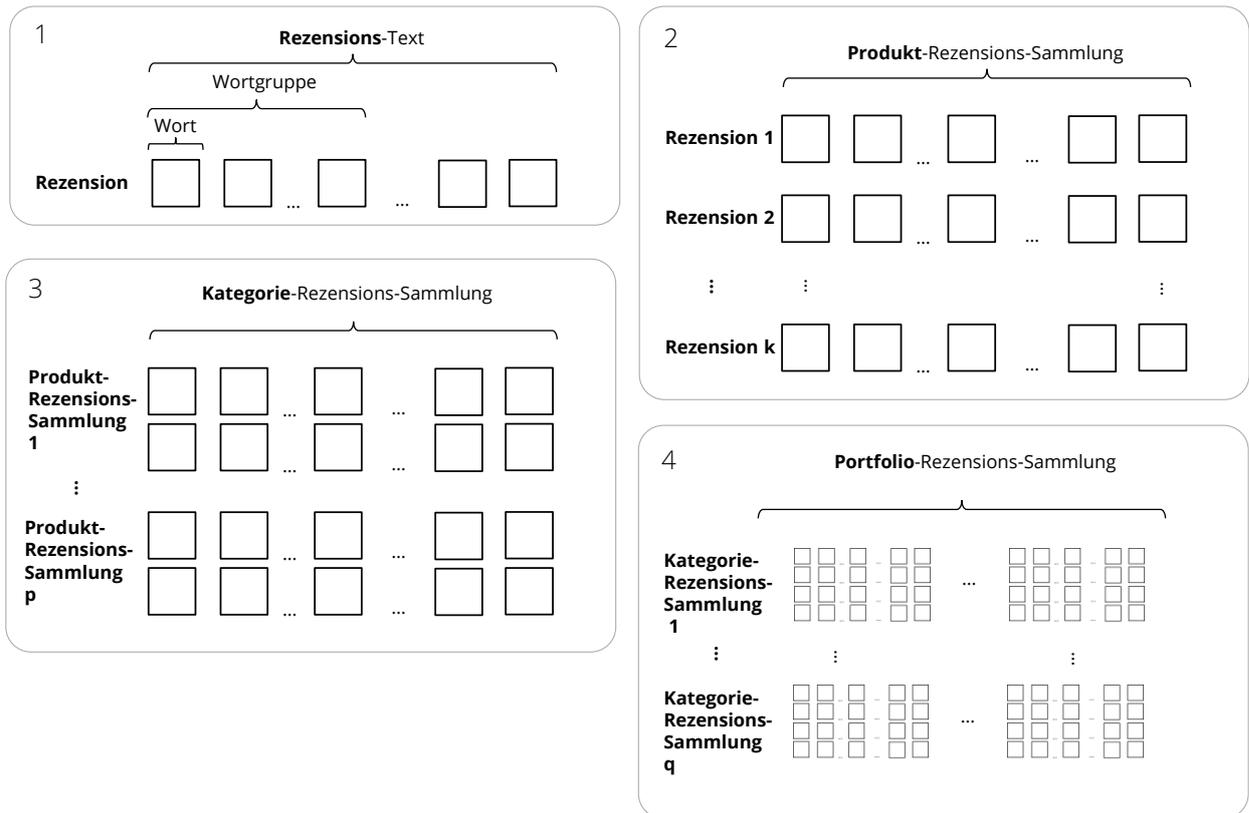


Abbildung 10: Hierarchischer Aufbau von Rezensionen auf Online-Handelsplattformen vom einzelnen Rezensions-Text bis zur gesamten Portfolio-Rezensions-Sammlung

Unter der Annahme, dass eine **Rezension** immer genau nur einem **Produkt** zugeordnet ist, ergeben sich die folgenden Verbindungen zwischen den Ebenen. Alle Rezensionen für ein Produkt bilden die **Produkt-Rezensions-Sammlung**. Alle **Produkt-Rezensions-Sammlungen** für vergleichbare Produkte bilden die **Kategorie-Rezensions-Sammlung** und die Sammlung aller **Kategorie-Rezensions-Sammlungen** bilden die **Portfolio-Rezensions-Sammlungen**.

In dieser Arbeit werden nur die Ebenen 1 und 2 betrachtet. Somit werden die einzelnen Bestandteile der Rezensionstexte im Detail analysiert und anschließend alle Rezensionen für das vorliegende Produkt ausgewertet und zueinander in Beziehung gesetzt. Dieses Vorgehen ermöglicht keine Aussagen über die Produktkategorie (Ebene 3) oder das gesamte Portfolio (Ebene 4). Es bildet die theoretische Grundlage um zukünftig auch auf den übergeordneten Ebenen datenbasierte Analysen durchzuführen.

Schlagworte und #Tags

Schlagworte (tag) oder Deskriptoren werden genutzt um Objekte zu beschreiben. Der Vorgang der Zuordnung eines Schlagworts zu dem Objekt wird als Verschlagwortung oder Indexierung bezeichnet. Typischerweise entstammen die Schlagworte einem kontrollierten Vokabular und werden nach

festen Regeln vergeben. Beispielweise bei der Indexierung von Literatur in einer Bibliothek oder der Verschlagwortung von Bildern bei einer Fotoagentur. Existieren weder ein kontrolliertes Vokabular, noch feste Vergaberegeln spricht man vom freien Verschlagworten. Dieses Vorgehen ist im Internet in sozialen Netzwerken ein übliches Vorgehen. Die dabei entstehende Heterogenität stellt dabei die größte Schwachstelle für die nachfolgende Nutzung (z.B. Suche) dar.

Eine Vielzahl wissenschaftlicher Studien untersucht die Verwendung von Schlagwörtern. So konnte beispielsweise eine Korrelation zwischen dem verfügbaren Vokabular und der Häufigkeitsverteilung am Beispiel der Plattform Open-Street-Maps nachgewiesen werden (Hukal 2018).

Für die hier vorliegende Arbeit werden Schlagworte aus einem *nicht kontrollierten Vokabular* verwendet um Textabschnitte zu beschreiben. Wiewohl die Schlagworte nicht aus einem kontrollierten Vokabular verwendet wurden, werden die Schlagworte mit einer Methode (systematische Vorgehensweise) abgeleitet. Dieser Ansatz ist universell anwendbar und ermöglicht die Herleitung vergleichbarer Schlagwörter für ähnliche Produkte. Dadurch wird die notwendige Flexibilität gewährleistet, um verschiedene Produktgruppen mithilfe dieses Ansatzes zu Verschlagworten. Gleichzeitig wird nur ein begrenztes Vokabular erzeugt. Dieses begrenzte Vokabular vereinfacht die Vergleichbarkeit zukünftiger Datensätze und ermöglicht dadurch eine bessere Grundlage für zukünftige Auswertungen.

3 Stand der Technik

In diesem Kapitel wird der aktuelle Stand der Technik für die Auswertung nutzergenerierter Inhalte für die Produktentwicklung beschrieben. In Abbildung 1 wurden die verschiedenen Domänen und deren Zusammenhänge schematisch dargestellt. Entsprechend der angrenzenden Domänen (Informatik, Produktentwicklung, Marketing, Linguistik etc.) konzentriert sich der Stand der Technik auf die Beschreibung der Überschneidung von NLP, Nutzerintegration in der Produktentwicklung und nutzergenerierten Inhalten.

Die mit dieser Arbeit verfolgte Zielsetzung, nämlich die Auswertung von nutzergenerierten Inhalten, fällt auch in den Bereich der Wissenssoziologie. Insbesondere für die Produktentwicklung ins Wissensmanagement. Eine sehr etablierte wissenschaftliche Sichtweise auf das Wissensmanagement gliedert sich in die Phasen: *Schaffung von Wissen, Speicherung und Abruf von Wissen, Wissenstransfer und Wissensanwendung* (Alavi und Leidner 2001). Bereits 2001 betonten die Autoren insbesondere die Rolle der Informationstechnologien (IT-Systeme) in der Phase der Wissensgenerierung mithilfe von Data Mining durch Kombination neuer Datenquellen. Standen damals noch unternehmensinterne Intranets im Vordergrund, so stellen heute vor allem externe Datenquellen im Internet eine neue Art von Informationsquelle dar. Diese Arbeit versucht daher, einen konkreten Lösungsvorschlag für das Wissensmanagement in der Produktentwicklung zu liefern. Die Integration der neu generierten Daten in bestehende Wissensmanagementsysteme ist jedoch nicht der Kern der Arbeit.

Beginnend mit einer detaillierten Beschreibung der NLP-Strömungen wird ein tiefergehendes Verständnis für die aktuellen Algorithmen, Ansätze und die hier entwickelte Methode möglich. Diesem konzeptionellen Beschreibungsansatz folgt eine konkretere Darstellung von NLP, indem Fragestellungen, sprachverarbeitende Aufgaben und deren Funktionsweisen im Detail beschrieben werden. Mit diesen Erklärungen ist es möglich, die zugrundeliegende Logik bei den verschiedenen NLP Fragestellungen zu verstehen.

Der Stand der Technik der Nutzerintegration in der Produktentwicklung baut auf der Erklärung im Kapitel Ausgangssituation auf. Er wird ergänzt um die bestehenden Methoden zur Nutzerintegration aus den Fachdomänen Design, Marketing und technische Produktentwicklung (Engineering). Die Methoden werden dann verglichen und nach ihrem NLP-Potenzial bewertet.

Anhand ausgewählter internationaler Publikationen werden die Erkenntnisse aus der Potenzialanalyse noch einmal dargestellt und untermauert. Alle untersuchten Publikationen versuchen, nutzergenerierte Inhalte für die Produktentwicklung nutzbar zu machen. Die Analyse dieser aktuellen Forschungsarbeiten vergleicht die Modellierungsansätze, Methoden und Forschungsziele. Die Ergebnisse der Analyse werden anschließend zusammengefasst und bewertet. Aufbauend auf dieser Analyse werden der

konkrete Handlungsbedarf und die Zielsetzung zur bereits vorgestellten Problemstellung der vorliegenden Arbeit abgeleitet.

3.1 Natural Language Processing (NLP)

Zugunsten einer dialektischen, thematischen Analyse der verschiedenen Arten und Trends von NLP, wird auf eine chronologische Übersicht der Hauptvertreter in diesem Bereich verzichtet⁶. Für ein besseres Verständnis der thematischen Hauptströmungen von NLP, wird das von Charles William Morris in den 1930 und 1940er Jahren geprägte Konzept der Semiotik (Zeichentheorie) durch die Unterscheidung zwischen **Syntaktik**, **Semantik** und **Pragmatik** im Folgenden näher beschrieben⁷.

Die **Semantik** beschreibt die Bedeutung und den Inhalt (des Zeichens), die **Syntaktik** die Gesetzmäßigkeiten, die Form, Anordnung etc. der Zeichen; und die **Pragmatik** das Ziel, Funktion (des Zeichengebrauchs). Vereinfacht können folgende Assoziation den drei Begriffen zugeordnet werden:

Syntaktik = Erscheinung,

Semantik = Inhalt,

Pragmatik = Wirkung.

Wie können diese Elemente helfen die NLP Strömungen zu klassifizieren? Diese Frage beantworten Cambria und White (2014) sehr ausführlich (auch wenn die Autoren Morris nicht referenzieren). Die nachfolgende Beschreibung basiert auf dieser Publikation (ebd.).

3.1.1 Trends und Paradigmen

Die Autoren Cambria und White (2014) modellieren die drei verschiedenen Paradigmen von NLP mittels der in den Betriebswirtschaften etablierten Methode der *S-Kurven* bzw. der *Springenden S-Kurven* (vgl. Abbildung 11). Dabei werden den eben eingeführten Begriffen von Syntaktik, Semantik und Pragmatik je eine Kurve zugeordnet und in den NLP Kontext überführt. Diese drei Paradigmen, Strömungen oder auch Modelle werden nun im Detail beschrieben.

⁶ An dieser Stelle soll auf die Arbeiten von Jones 1994, Moisl et al. 2000 und Cambria und White 2014 verwiesen werden.

⁷ Weder die damit verbundene Kontroverse der Ordnung dieser drei Elemente zueinander, noch eine ausführliche linguistische Abgrenzung der Begriffe Syntax und Syntaktik, können hier thematisiert werden. Da es sich hierbei um komplizierte und nach wie vor ungelöste Probleme zwischen den kommunikationstheoretisch gerichteten Wissenschaftsbereichen handelt.

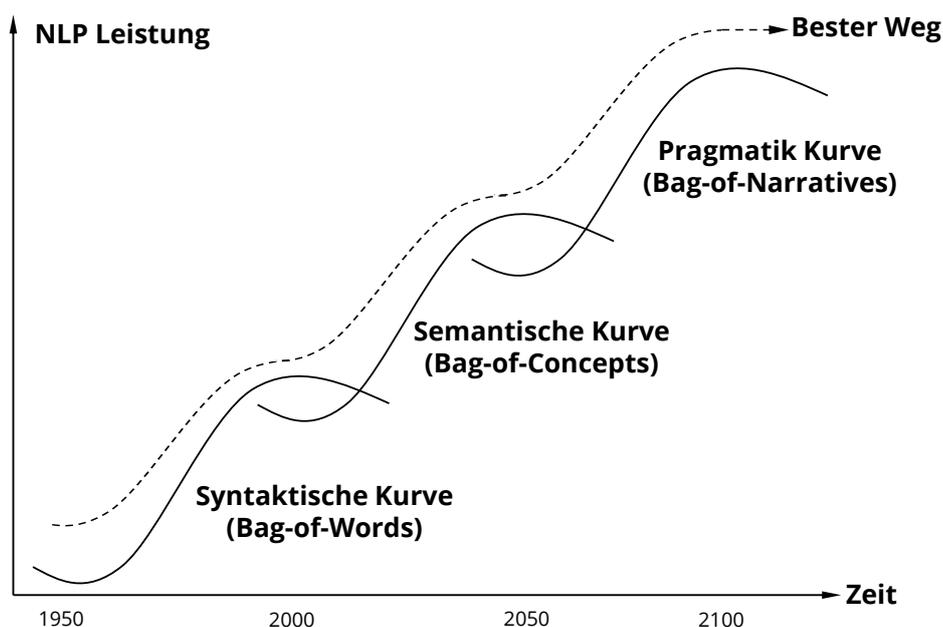


Abbildung 11: Zeitlicher Verlauf und Paradigmenwechsel von NLP (Cambria und White 2014, S. 51)

Die **Syntaktische Kurve (Bag-of-Words)** beschreibt Ansätze, die Wörter und deren Beziehungen untersucht. Dabei unterscheiden die Autoren folgende Methoden: (1) Schlagwortsuche, (2) wortbezogene Ähnlichkeit und (3) statistische Methoden.

- (1) Die einfachste Vorgehensweise dieser Ansätze durchsucht Texte nach Schlagwörtern, sog. Schlagwortsuche (Keyword-Spotting).
- (2) Eine komplexere Methode innerhalb des Bag-of-Words Paradigmas ist die wortbezogene Ähnlichkeit (Lexical Affinity). Dabei werden Worte und deren Relationen zueinander mithilfe von sog. Wortnetzen abgebildet. Das 1985 an der Universität Princeton entwickelte und stetig erweiterte WordNet ist das größte und weltweit bekannteste. Für die deutsche Sprache entwickelte die Universität Tübingen das Wortnetz mit dem Namen GermaNet. Eine sehr gute Übersicht und Funktionsweise zu WordNet haben Jurafsky und Martin (2020b) veröffentlicht.
- (3) Bei den statistischen Methoden werden annotierte Texte (als Trainingskorpus) in einen maschinellen Lernalgorithmus eingespeist und trainiert. Dieser Ansatz ist ein anhaltender Trend seit den 1990er Jahren.

Die **Semantische Kurve (Bag-of-Concepts)** beschreibt Ansätze die die Bedeutung von Wörtern und deren Wechselwirkung untersucht. Dabei können die Methoden innerhalb dieses Paradigmas in *externe* und *interne* Ansätze unterteilt werden. Externe Ansätze benötigen zusätzliche (*externe*) Informationen, wohingegen interne Ansätze nur die vorliegenden (*internen*) Informationen verwenden, um eine Aussage herzuleiten. Diese Strömung unterteilt sich in drei NLP-Arten: (1) taxonomisches NLP, (2) endogenes NLP und (3) noetisches NLP.

- (1) Taxonomisches NLP, auch als Ontologie bezeichnet, versucht eine der natürlichen Sprache angepasste, verbundene, hierarchische Semantik zu entwickeln. Damit können universelle Logikbäume (Web-Ontologie) entwickelt werden. Für die Einteilung und die Merkmalsbestimmungen werden zusätzliche Informationen benötigt, dadurch ist es ein *externes* Verfahren.
- (2) Beim endogenen NLP werden maschinelle Lernverfahren eingesetzt, um eine semantische Analyse des gesamten Textkorpus durchzuführen. Dabei wird nur der vorliegende Datensatz betrachtet. Das Vorgehen stützt sich demnach ausschließlich auf das endogene (im Inneren erzeugte) Wissen der Dokumente und ist somit ein *internes* Verfahren.
- (3) Noetisches NLP umfasst alle vom Verstand inspirierten NLP-Ansätze⁸. Dieser Ansatz wird von den Autoren als Allzwecklösung eingeführt. Er dient dazu, die Schwachstellen der anderen Ansätze zu kompensieren. Aktuelle Deep Learning Ansätze für NLP-Fragestellungen stellen eine Möglichkeit dieser Kategorie dar. Ziel dieser Methode ist die Nachbildung des menschlichen Verhaltens bei der Aufnahme, Verarbeitung und Auswertung von Texten. Dieser Vorgang benötigt keine analytische Verarbeitung der Texte. Den Autoren zufolge verfügt der Mensch über ein natürliches konstruktionsbasiertes semantisches System, das diese Aufgabe übernimmt.

Die letzte Kurve ist die **Pragmatik Kurve (Bag-of-Narratives)**. Sie beschreibt Ansätze von NLP, die die Beziehung zwischen Text und Nutzer, insbesondere das Verstehen und Erzeugen von Erzählungen, erklären. Den Autoren entsprechend ist dieses Paradigma die Zukunft von NLP. Da die Funktionsweise der menschlichen Intelligenz, durch das Verständnis darüber wie Narrative⁹ vom menschlichen Gehirn erzeugt und verarbeitet werden, erklärt werden kann. Wie werden narrative Texte (Erzählungen) aufgebaut (Zeitebenen, Handlungsstränge, Komposition, Struktur und kausaler Zusammenhang)? Welche Mechanismen laufen beim Erstellen dieser Texte im menschlichen Gehirn ab? Diese Fragen versucht die Pragmatik Kurve zu beantworten.

Die drei beschriebenen Paradigmen ermöglichen nun bestehende Ansätze, Methoden und Algorithmen einzuordnen und zu vergleichen. Dieser übergeordnete Ordnungsrahmen hilft dabei, die derzeitigen Bemühungen von Wissenschaft und Technik um NLP zu verstehen. Entsprechend der Analyse von Cambria und White basieren die heutigen NLP-Ansätze zum größten Teil auf den Grundprinzipien der Syntaktischen Kurve.

⁸ Der Begriff Noetik: (ursprünglich Lehre vom Denken) beschreibt den Inhalt der menschlichen Erkenntnis und wie Wissen zustande kommt.

⁹ Unter einem narrativen Text versteht man die sprachliche Darstellung und Gestaltung einer Handlung und/oder Begebenheit bzw. eines Erlebnisses.

3.1.2 Methoden, Prinzipien und Funktionsweisen

Die wichtigsten Fragestellungen, sprachverarbeitende Aufgaben und Funktionsweisen des Bag-of-Words-Paradigmas werden im Folgenden beschrieben. Da es sich hierbei um reale Implementierungen und öffentlich zugänglichen Software-Code handelt, wird die Online-Dokumentation der Implementierung von TextBlob¹⁰ als Wissensgrundlage dieses Kapitels genutzt (Loria 2021). Die folgenden NLP-Funktionen werden im Detail beschrieben:

- (1) Part-of-Speech Tagging (Wortartbestimmung),
- (2) Sentiment Analysis (Stimmungsanalyse),
- (3) Tokenisation (Tokenisierung),
- (4) Words Inflection and Lemmatization (Wortflexion und Lemmatisierung),
- (5) Spelling Correction (Rechtschreibkorrektur),
- (6) n-Grams,
- (7) Vektorisierung und
- (8) Classifier und Clustering (Klassifikator und Gruppieren).¹¹

(1) Die Methode **Part-of-Speech-Tagging (POS-Tagging)** leitet sich von dem englischen Wort Part-of-Speech ab, was Wortart bedeutet. Beim POS-Tagging werden also die Wortarten und Satzzeichen eines Textes bestimmt. Die deutsche Sprache hat zehn verschiedene Wortarten: Substantiv, Artikel, Adjektiv, Pronomen, Numeral, Verb, Adverb, Präposition, Konjunktion und Interjektion. Die englische Sprache nur acht. Das am häufigsten verwendete Verfahren von POS-Tagging basiert auf sog. Supervised Machine Learning Algorithmen. Der Aufbau eines POS-Taggings ist komplex und bedarf eines fundierten Wissens über die jeweilige Sprache. Beispielsweise gilt für die lexikalische Ordnung im englischen Satzbau die Regel: Erst Subjekt, dann Prädikat und zuletzt Objekt. Solche sprachspezifischen Regeln werden in einem Datensatz, einem sogenannten Textkorpus, abgespeichert. Der größte Textkorpus, genannt TIGER-Korpus, für die deutsche Sprache ist von der Universität Stuttgart. Der Textkorpus besteht aus 900.000 Token (ausführlicher Erklärung folgt) (50.000 Sätzen), die aus der *Frankfurter Rundschau* stammen (Tik 2021). Neben POS-Tags und syntaktischen Strukturen berücksichtigt dieser Datensatz morphologische und lemmatische Informationen (ebd.). Folgende populäre Methoden verwenden Textkorpora: *Hidden-Markov Modell*, *Eric Brills Verfahren* oder *Decision Tree* (Entscheidungsbäume). Zum besseren Verständnis der Funktionsweise dieser Methoden, wird die Logik der Methoden üblicherweise in Graphen kombiniert mit einer besonderen Abkürzungslogik

¹⁰ Jede andere Implementierung einer Python-basierten NLP-Bibliothek wäre ebenfalls denkbar gewesen.

¹¹ Diese Funktionsbeschreibungen sind vielmehr Eigennamen und werden daher auch im weiteren Verlauf nicht ins Deutsche übersetzt.

visualisiert. Dabei wird der Satz nach Wortarten und Satzkonstruktionen (Syntax) aufgeteilt. Die Hauptworte der deutschen Sprache und die dazugehörigen Tags zeigt Abbildung 12 (ein Ausschnitt einer ausführlichen Tag-Tabelle findet sich im Anhang A.2 - Abbildung 54).

- | | |
|--------------------------|--------------------------|
| 1. Nomina (N) | 7. Adverbien (ADV) |
| 2. Verben (V) | 8. Konjunktionen (KO) |
| 3. Artikel (ART) | 9. Adpositionen (AP) |
| 4. Adjektive (ADJ) | 10. Interjektionen (ITJ) |
| 5. Pronomina (P) | 11. Partikeln (PTK) |
| 6. Kardinalzahlen (CARD) | |

Abbildung 12: Hauptworte und deren Abkürzungen nach STTS (Stuttgart-Tübingen-TagSet) (nach Schiller et al. 1999)

Weiterhin besteht die Möglichkeit grammatikalische Funktionen (Adverbialbestimmungen etc.), Textkategorien (Überschrift, Titel etc.) und semantische Aussagen (Zeitbestimmung, Ortsangabe, Sinn und Zweck etc.) den Wortgruppen zuzuordnen, um ein noch besseres Textverständnis zu ermöglichen (Taylor et al. 2003).

Das folgende Beispiel soll die Komplexität und die NLP-Leistung dieser Methode verdeutlichen. Die Analyse des Teilsatzes [...] *das bereits erreichte Wahlergebnis* [...] mit POS-Tagging identifiziert neben der einfachen Wortartbestimmung, auch den inneren grammatikalischen Zusammenhang zwischen dem Wortpaar *bereits erreichte* (vgl. Abbildung 13).

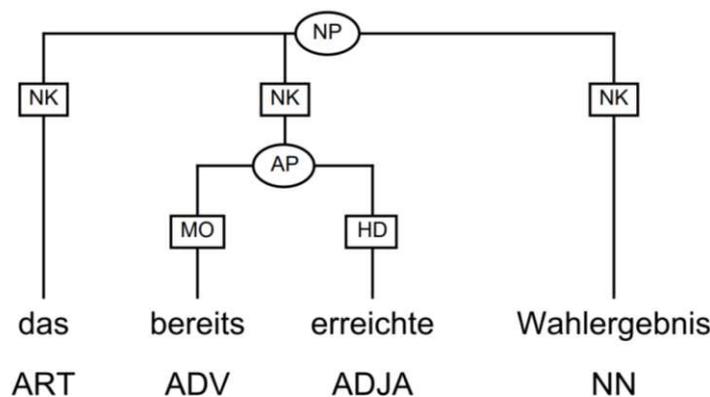


Abbildung 13: Graphendarstellung für das POS-Tagging eines einfachen Teilsatzes (HU BERLIN)

Die Genauigkeit und die Leistung dieser Methode ist neben der mathematischen Modellierung natürlich sehr stark von dem verwendeten Text-Korpus (Bag-of-Words) abhängig. Eine umfassende Liste der nutzbaren Korpora für die NLP-Bibliothek NLTK (Natural Language Toolkit) zeigt Tabelle 3.

Tabelle 3: Ausschnitt verschiedener Textkorpora für die NLP-Bibliothek NLTK (Bird et al. 2009) (Eine vollständige Übersicht ist dem Anhang A.1 zu entnehmen)

Korpus-Name	Compiler	Inhalte
Brown Corpus	Francis, Kucera	15 Genres, 1,15 Mio. Wörter, getaggt, kategorisiert
CESS Treebanks	CLiC-UB	1 Mio. Wörter, getaggt und geparst (Katalanisch, Spanisch)
Chat-80 Data Files	Pereira & Warren	Weltgeografische Datenbank
CMU Pronouncing Dictionary	CMU	127k Einträge
CoNLL 2000 Chunking Data	CoNLL	270k Wörter, getaggt und gechunked
CoNLL 2002 Named Entity	CoNLL	700k Wörter, verschlagwortet und mit Namen versehen (Niederländisch, Spanisch)

(2) Die nächste Methode **Sentiment Analysis (Stimmungsanalyse)** bestimmt die Stimmung eines Textes. Die Stimmung eines Satzes beschreibt, wie positiv oder negativ eine Textpassage gemeint ist. Diese Aussage ist stark subjektiv. Daher wird neben der Bestimmung der Stimmungswerte oft auch der Grad der Subjektivität des Textes ermittelt. In dem hier beschriebenen Fall werden nur die Adjektive des Satzes analysiert, um die Stimmung zu bestimmen.

Für die Berechnung des Stimmungswertes wird eine zuvor erstellte Datenbasis, nämlich das Lexikon, genutzt. Das Lexikon stellt somit den bekannten und gleichzeitig auch aktiven Wortschatz dar. Dieses Subjektivitäts-Lexikon besteht aus Adjektiven. Für jedes Adjektiv wird eine Vielzahl an Kennzahlen aufgelistet. Für den hier verwendeten Fall stellen der Stimmungswert (positiv/negativ, -1.0 – +1.0) und Subjektivitätswert (objektiv/subjektiv, -0.0 – +1.0) die zwei wichtigsten Kennzahlen dar. Der Vollständigkeit halber werden auch die anderen Informationen des Subjektivitäts-Lexikons aufgeführt. Die Zuverlässigkeit (reliability) gibt an, ob das Adjektiv per Hand markiert wurde (1.0) oder hergeleitet wurde (0.7). Weiterhin werden unterschiedliche Bedeutungen der Adjektive mit aufgenommen und klassifiziert, beispielsweise lächerlich (mitleidig) = negativ, lächerlich (humorvoll) = positiv¹². Auch der Querverweis zu anderen lexikalischen Datenbanken wie WordNet3 oder Cornetto (Vossen et al. 2007) wird angegeben. Die Klassifikation der Wortart (Part-of-Speech, POS) entspricht der Penn Treebank Nomenklatur, demnach *JJ* für Adjektive. Den Einfluss des Adjektivs auf das nächste Wort, beschreibt der Wert für die Intensität (intensity). Für eine objektive Bewertung und Einschätzung von Datenbanken, wird die Leistungsfähigkeit des Lexikons bei der Anwendung auf einen bekannten Datensatz angegeben. In dem Fall des Datensatzes für englische Filmkritiken (Pang & Lee polarity dataset v2.0) beträgt die Genauigkeit für das verwendete Lexikon 75 % (P 0,76, R 0,75, F1 0,75).

Beispielauszug aus dem verwendeten Lexikon:

```
<word form="yellow"
cornetto synset id="n a-509864"
wordnet_id="a-00385756"
pos="JJ"
sense="of the color intermediate between green and orange in the color
spectrum"
polarity="0.0"
subjectivity="0.0"
intensity="1.0"
confidence="0.8" />
```

Wie ist es möglich, mit dieser Liste Stimmungswerte aus Wortgruppen zu ermitteln? Dafür ist es notwendig den Einfluss anderer Worte auf das Adjektiv zu verstehen. Der verwendete lexikalische Ansatz berücksichtigt vier Arten von Einflüssen: **Negation, Verstärkung, Kombination** und **Ignoranz**.

Negation beschreibt das Phänomen, wenn die Wortbedeutung verneint wird. Oftmals geschieht das durch Worte wie *kein, nicht, weder...noch, nichts, niemand* usw. Durch die Negation wird die Bewertung

¹² Linguistisch gesehen ist es fraglich, ob das Wort *lächerlich* in der Sprache positiv verwendet wird.

des Adjektivs umgekehrt. Um diesen Einfluss der Negation zu berechnen wird der Zahlenwert für die Polarität mit $-0,5$ multipliziert. Ferner hat die Negation keinen Einfluss auf die Subjektivität.

Die **Verstärkung** von Adjektiven ist durch Worte (sog. Intensifier oder Modifier) wie sehr, viel, mega, voll etc. möglich. Diese Worte sind ebenfalls in dem Lexikon enthalten. Der Einfluss des Verstärkungsworts auf das Adjektiv beschreibt der Intensitäts-Wert. Dieser Wert ist immer größer als 1. Um den Einfluss der Verstärkung zu berechnen, wird der Zahlenwert für die Polarität mit dem Intensitätswert des Verstärkungswortes multipliziert. Im Unterschied zu der Negation ändert sich auch der Wert der Subjektivität durch die Verstärkung. In der gleichen Art wie die Polarität wird die Subjektivität auch mit dem Zahlenwert des Verstärkungsworts multipliziert.

Die **Kombination** von Negation und Verstärkung ist möglich und wird durch Multiplikation der einzelnen Faktoren berechnet. Weiterhin berücksichtigt der Algorithmus die Kombination von Wortgruppen indem die ermittelten Werte für Polarität und Subjektivität gemittelt werden.

Die **Ignoranz** bezeichnet die Fähigkeit des Algorithmus Wörter, die entweder nicht im Lexikon aufgeführt werden oder Wörter die explizit ausgeschlossen sind, nicht in der Kalkulation zu berücksichtigen.

(3) Die Methode **Tokenisation** (Tokenisierung) teilt einen Text in kleinere Teile auf, sogenannte Token. Dieser Token kann aus Zeichen, Wörtern oder Wortgruppen bestehen. Dieser Vorgang wird oftmals genutzt um anschließende Auswertungen effizienter zu gestalten. Bei der Tokenisierung eines Satzes kann beispielsweise die Häufigkeitsverteilung der Wörter mit Token schneller berechnet werden, als wenn alle Zeichenketten einzeln verglichen werden.

(4) Eine weitere sehr übliche NLP-Vorgehensweise wird als **Words Inflection and Lemmatization** (Wortflexion und Lemmatisierung) bezeichnet. Der Ursprung dieser Ansätze stammt aus der Linguistik, nämlich aus dem Bereich der Morphologie. Die Morphologie untersucht Wörter, deren Aufbau und Regeln. Dabei stellt ein Morphem die kleinstmögliche sinnhafte zu untersuchende Einheit eines Wortes dar. Bekanntere Beispiele sind die Klassen der Prä- und Suffixe, wie beispielsweise „un-“ und „-in“. Bei der Lemmatisierung wird ein Wort wieder auf seine Grundform reduziert. Bei dem entgegengesetzten Vorgang, der Flexion (Beugung) ändert sich nicht das Wort, sondern es werden nur zusätzliche grammatikalische Informationen dem Wort hinzugefügt. So werden Substantive und Adjektive dekliniert und Verben konjugiert. Dabei werden bei der Flexion keine neuen Wörter (Lexeme) gebildet. Die Wortbildung geschieht beispielsweise durch Morpheme. Das stellt den größten Unterschied zwischen Flexion und Lemmatisierung in der Linguistik dar. Im Rahmen von NLP werden diese Regelmäßigkeiten für den Aufbau von Wörtern und deren Bedeutung genutzt, um weitere Erkenntnisse über die darin enthaltenen Sätze oder Wörter zu gewinnen. Unter Zuhilfenahme der Methode der Tokenisierung, ist es möglich verschiedene Worte auf gemeinsame Lemma zu reduzieren

und mittels eines Tokens zu verarbeiten. Dadurch wird die Berechnung effizienter und Muster in dem Datensatz können besser erkannt werden.

(5) **Spelling Correction** (Rechtschreibkorrektur) ist die am häufigsten verwendete NLP-Methode in der heutigen Zeit. Einen sehr direkten und einfach zu verstehenden Lösungsweg für den Aufbau einer automatisierten Rechtschreibkorrektur beschreibt Peter Norvig auf seiner Website¹³. Die Funktionsweise wird kurz im Detail beschrieben. Ausgangspunkt ist ein zu prüfender Text. Aus diesem Text werden alle Wörter gezählt und für jedes Wort die

$$\text{Wahrscheinlichkeit eines Wortes} = \frac{\text{Anzahl des einen Worts}}{\text{Anzahl aller Wörter}}$$

ermittelt. Dann werden die einzelnen Worte in zwei Kategorien unterteilt – in bekannte und unbekannte Worte. Die Klassifizierung wird durch einen Vergleich mit einer Datenbank (Textkorpus oder Wörterbuch) ermittelt. Worte, die nicht in der Datenbank gefunden wurden, werden als unbekannte Worte klassifiziert und stellen potentielle Wörter für eine Rechtschreibkorrektur dar. Diese Worte werden nun systematisch verändert und anschließend wieder mit dem Textkorpus verglichen. Die Veränderungen können in vier Kategorien unterteilt werden: löschen, vertauschen, ersetzen und einfügen. Dabei werden auf Buchstabenebene die Wörter entsprechend der Kategorien systematisch verändert. Anschließend wird das neue Wort wieder mit dem Textkorpus verglichen. Wird das Wort dann gefunden, ist die Rechtschreibkorrektur für dieses Wort erfolgreich abgeschlossen. Bei diesem sehr einfachen Algorithmus werden nur Wörter korrigiert, die mit ein oder zwei Änderungen wieder ein bekanntes Wort ergeben. Die Erweiterung des Algorithmus um die Wahrscheinlichkeitsverteilung einzelner Worte in einer Sprache, basieren auf einem großen Textkorpus. Die daraus abgeleitete Fehlerabschätzung bietet einen verbesserten Algorithmus. Dies soll hier an dieser Stelle nicht weiter beschrieben werden.

(6) Eine weitere sehr verbreitete Methode ist **N-Grams** (N-Gramms). Dabei wird der Text in Bestandteile, sog. *Grams*, aufgeteilt. Grams können Buchstaben oder auch Wörter sein. Diese Methode bietet die Möglichkeit wichtige Textinhalte automatisiert zu extrahieren. Untersucht werden dabei typischerweise Wortpaare (Bigram, N=2) oder 3er-Wortgruppen (Trigram, N=3), aber auch abstrakte Fragestellungen, wie beispielsweise: Welche 3er-Zeichenkombination sind in der deutschen Sprache am häufigsten?¹⁴ Um einen Text zu erschließen werden die verschiedenen N-Grams gewichtet. Eine sehr alte, aber für die Textverarbeitung noch immer sehr nützliche Vorgehensweise ist das sogenannte

¹³ <https://norvig.com/spell-correct.html>

¹⁴ Entsprechend der Untersuchungen des Instituts für Deutsche Sprache ist „EN_“ die am häufigsten auftretende 3-Zeichenkombination in der deutschen Sprache.

TF-IDF-Verfahren oder -Maß. Dieser Begriff setzt sich aus TF für term frequency (Vorkommenshäufigkeit) und IDF für inverse document frequency (inverse Dokumenthäufigkeit) zusammen und ist eine statistische Messgröße zur Beurteilung der Relevanz von N-Grams.

$$TF(t, f) = \frac{\text{Anzahl von } t \text{ in } d}{\text{Anzahl von Wörtern in } d}$$

$$df(t) = \text{Auftreten von } t \text{ in Dokument}$$

Statistische Verteilung von Wörtern in Texten, insbesondere die Kombination von Wortarten und deren Häufigkeitsverteilung ist das große Potenzial dieser Analysemethoden. Weiterhin stellt die „Nähe“ von Worten in Sätzen ein zentrales Merkmal zur Textanalyse dar, insbesondere, wenn dabei Verben betrachtet werden.

(7) **Vektorisierung** beschreibt den Vorgang, Textdaten in numerische Daten zu überführen. Durch diese Transformation können numerische Methoden zur Auswertung der Textdaten angewendet werden. Bag-of-Words (BoW) und Word Embedding (Word2Vec) sind zwei gängige Methoden um diese Aufgabe zu erfüllen. Beide Möglichkeiten werden hier kurz vorgestellt. Beginnend mit der Vorgehensweise von BoW.

Bei der BoW-Methode geschieht die Transformation in drei Schritten: (1) Worte zählen, (2) tf-Idf-Werte berechnen, (3) Vektor ableiten. Bei (1), Wörter zählen, werden alle Wörter gezählt und anschließend für jeden einzelnen Text die Worthäufigkeit ermittelt. (vgl. Abbildung 14).

Teilsatz	Schlüsselwörter			Vektor
	<i>du</i>	<i>bist</i>	<i>toll</i>	
<i>Du und du sind toll</i>	2	1	1	[2,1,1]
<i>Toll bist auch du</i>	1	1	1	[1,1,1]
<i>Wir sind auch toll</i>	0	0	1	[0,0,1]

Abbildung 14: Worthäufigkeitstabelle - Ein Zwischenschritt bei der Vektorisierung von Texten

Im nächsten Schritt werden die Vektoren in Bezug auf den gesamten Datensatz gewichtet. Das wird mithilfe des (2) tf-Idf-Werte (**T**erm **F**requency-**I**nverse **D**ocument **F**requencies)¹⁵ berücksichtigt. Dieser Wert berechnet sich nach der Gleichung

$$tf - IDF = TF(t, f) \times \log \frac{\text{Dokumente Gesamt}}{\text{Dokumente mit Wort}}$$

Dabei berücksichtigt der Logarithmus-Term (Idf) die Gewichtung seltener Wörter in allen Dokumenten des Textkorpus. Der tf-IDF wird für jedes einzelne Wort ermittelt. Der letzte Schritt überführt

¹⁵ Wörtliche Übersetzung: Begriffshäufigkeit - inverse Dokumenthäufigkeit

am wenigsten komplexe Algorithmus, der zufriedenstellende Ergebnisse liefert ist die logistische Regression. Anhand dieses Algorithmus soll der Wirkzusammenhang zwischen Text, Algorithmus und Aussage verdeutlicht werden.

Für die logistische Regression wird die Sigmoid Funktion genutzt, um auf Basis eines Inputs Werte zwischen 0 und 1 zu ermitteln (vgl. Abbildung 16).

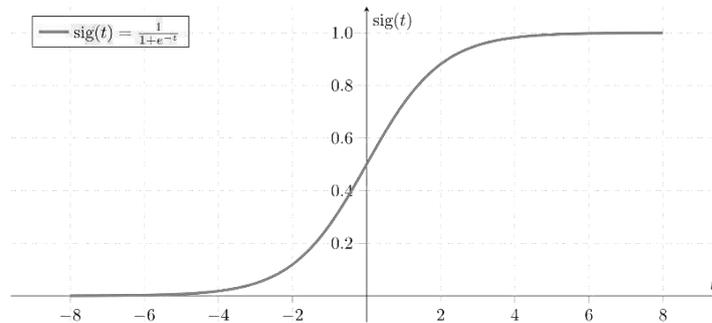


Abbildung 16: Die Sigmoid Funktion bildet die Grundlage der Einteilung bei der logistischen Regression

Je nachdem, wo der Schwellenwert liegt, wird dann die eine oder andere Kategorie für die Einstufung unterhalb oder oberhalb dieses Wertes ausgewählt. Aktuelle Python-Bibliotheken ermöglichen mit nur einer Zeile Code eine Klassifizierung auf Basis einer logistischen Regression. Daher wird hier auf die Beschreibung der zugrundeliegenden Mathematik und die Anpassung der Sigmoid Funktion verzichtet. Der notwendige Input für die logistische Regressionsanalyse ist der vorher erstellte Vektorsatz. Auf der Basis dieser Zahlenwerte wird die Sigmoid Funktion dann durch interne Rechenoperationen (Dimensionsreduktion etc.) so angepasst, dass der Datensatz mit Hilfe der Funktion in die zuvor zugeordneten Klassen eingeteilt werden kann.

3.1.3 Bewertungskriterien

Oftmals ist es möglich NLP-Funktionen mit verschiedenen Algorithmen oder Ansätzen zu lösen. Daher ist es notwendig die Lösungen miteinander zu vergleichen. Der nachfolgende Absatz beantwortet die Frage: Wie können NLP-Verfahren bewertet werden?

Es existieren keine direkten Bewertungskriterien für NLP-Verfahren. Vielmehr werden die Gütekriterien zur Bewertung von Machine Learning Algorithmen verwendet. Das Vorgehen besteht aus mehreren Schritten. Um die Güte des Algorithmus zu bewerten ist es üblich, den vorbereiteten Datensatz in Trainings- und Testdaten zu unterteilen. Mithilfe der Trainingsdaten wird der Algorithmus trainiert und angepasst. Das Ergebnis wird dann anhand der Testdaten überprüft. Dafür wird die Kategorie durch den Algorithmus berechnet und dann mit der vorher ermittelten (richtigen) Kategorie verglichen. Anhand dieser Abweichungen lassen sich verschiedene Kennwerte für die Qualität der Ergebnisse ermitteln. Die dafür üblichen Kennwerte sind: **Accuracy**, **Precision**, **Recall**, **F1-Score** und **Confusion Matrix**. Was die Werte bedeuten und wie sie ermittelt werden, wird im Folgenden beschrieben.

Der Kennwert **Accuracy** (Genauigkeit) beschreibt wie treffsicher das Verfahren ist und berechnet sich aus dem Quotienten:

$$Accuracy = \frac{\text{Anzahl korrekt klassifiziert}}{\text{Anzahl gesamt klassifiziert}}$$

Bei Datensätzen mit ungleich verteilten Klassen (eine Klasse ist viel häufiger vertreten), kann der Kennwert *Accuracy* einen falschen Rückschluss auf die Genauigkeit des Algorithmus zur Folge haben. Um eine exaktere Aussage zur Treffsicherheit (das Wort ist in diesem Fall sehr treffend) zu erhalten, wird jede Klassifizierung in 4 Gruppen unterteilt. Ausgangspunkt stellt die „Wahl der Klasse“ dar. Bei einem binären Klassifizierungsproblem (Spam und Nicht-Spam) ergeben sich die folgenden 4 Szenarien für *Spam*.

1. Es wurde Spam gewählt – was korrekt ist (True Positiv – **TP** | Treffer)
2. Es wurde Spam nicht gewählt – was korrekt ist (True Negativ – **TN** | korrekter Ausschuss)
3. Es wurde Spam gewählt – was falsch ist (False Positiv – **FP** | Fehlalarm)
4. Es wurde Spam nicht gewählt – was falsch ist (False Negativ – **FN** | Verpasst)

Diese verschiedenen Szenarien und die daraus ermittelten Werte werden zur besseren Übersichtlichkeit typischerweise in einer sogenannten Confusion-Matrix¹⁶ aufgelistet. Auf der Grundlage der in der Confusion-Matrix ermittelten Werte kann ein genauerer Kennwert, nämlich der *Recall-Wert*, berechnet werden.

Der *Recall* Kennwert berechnet sich aus

$$Recall = \frac{\text{Treffer}}{\text{Treffer} + \text{Verpasst}}$$

und stellt somit ein Gütekriterium für die Überdeckung – also die Erkennung der Fehler – dar. Der Gegenspieler dieses Kriteriums ist die *Precision* (Genauigkeit). Dieses berechnet sich nach

$$Precision = \frac{\text{Treffer}}{\text{Treffer} + \text{Fehlalarm}}$$

und gibt Rückschlüsse wie präzise der Algorithmus funktioniert. Das Ziel des Algorithmus ist es, eine große Überdeckung, bei gleichzeitiger hoher Präzision zu erreichen. Um dieses Ziel in Abhängigkeit eines Parameters abzubilden, wurde der *F1-Score* eingeführt. Dieser berechnet sich nach

$$F1 = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$$

¹⁶ Es existiert kein etablierter Deutscher Begriff dafür. Die Begriffe *Wahrheit*, *Verwechslung*, *Durcheinander* oder *Verwirrung* scheinen hierfür eine passende Übersetzung zu sein.

und bildet somit das harmonische Mittel¹⁷ aus den zwei vorherigen Kennwerten. Je höher der *F1-Score* ist desto besser ist das Modell. Der F1-Score ist eine besondere Art des F β -Scores. Wobei β die Gewichtung von Recall zu Precision repräsentiert. Der F β -Score berechnet sich in seiner allgemeinen Form entsprechend:

$$F\beta = (1 + \beta^2) \times \frac{\text{Recall} \times \text{Precision}}{(\beta^2 \times \text{Precision}) + \text{Recall}}.$$

Es existieren noch weitere Kennwerte (Matthews Correlation Coefficient, Receiver Operating Characteristic etc.). Diese Kennwerte müssen immer im Kontext des Datensatzes interpretiert werden, da diese Gütekriterien je nach Anwendung falsche Schlüsse zulassen können.

Zwischenfazit

Die vorgestellten Methoden, Prinzipien und Funktionsweisen computergestützter Verarbeitung natürlicher Sprache stellt die aktuelle Wissensgrundlage moderner NLP-Anwendungen dar. Trotz der immensen Bestrebungen seit mehr als 30 Jahren existiert keine Lösung wie NLP-Systeme in realen Anwendungen logische Schlussfolgerungen ziehen oder das Wissen der Welt in einer allgemeinen und robusten Weise nutzen können (Bird et al. 2009). Diese große Einschränkung verhindert einen übergreifenden Lösungsansatz. Daher fokussieren sich aktuelle Forschungen auf die Entwicklung domänenspezifischer Lösungen für NLP. Anhand dieser Einzellösungen lassen sich die Schwierigkeiten bei der Anwendung der vorgestellten Methoden und Paradigmen phänomenologisch nachvollziehen. Exemplarisch wird im Folgenden eine Auswahl an bestehenden Lösungen für verschiedene Anwendungsfelder beschrieben.

3.2 Methoden zur Nutzerintegration

Welche etablierten Ansätze und Methoden für die Integration der Kundschaft in den Produktentwicklungsprozess existieren? Welche Informationen werden dabei extrahiert? Wie werden diese Informationen verarbeitet? Es existiert eine Vielzahl an Methoden zur Verbesserung des Kundenverständnisses durch das Externalisieren von Wissen. Folgende Methoden sind im Produktdesign bekannte Vorgehensweisen: Persona, User-Needs, Kundenprofile, User Journey Maps, Vor-Ort-Besuche oder Empathy Maps. Diese Methoden ermöglichen eine Vielzahl an Wahlmöglichkeiten bei der Umsetzung. Ihre Ergebnisse sind oftmals sehr stark von der jeweiligen Umsetzung abhängig. Mehr Vorgaben und stringendere Abläufe bei der Umsetzung haben die Methoden aus dem Marketing-Bereich: Conjoint-Analysis, Menu-Based-Choice-Verfahren (MBC), Experteninterviews und Anwenderbefragung. Wie auch die Methoden aus dem Ingenieurbereich: Usability-Tests, Handlungsanalyse und House-of-Quality.

¹⁷ Diese Kennzahl ist ebenfalls ein Mittelwert. Dieser Mittelwert hat aber den Vorteil die Gewichtung mit zu berücksichtigen; im Gegensatz zu arithmetischen Mittel.

Tabelle 4 listet die genannten Methoden, eine **Kurzbeschreibung**, ihren **Aufwand** bei der Durchführung und die **verwendeten Informationen** und **Ressourcen** (Zielstellung oder Bestandteile der Methode) auf. Für jede Methode wurde eine Potentialanalyse durchgeführt. Für die Potentialanalyse wird angenommen, dass nutzergenerierte Inhalte nützliche Informationen über das Produkt enthalten. Weiterhin beschränkt sich diese Potentialanalyse auf Produktrezensionen und auf Online-Handels-Plattformen. Sie basiert auf den Erfahrungen des Autors, welche Informationen diese Rezensionen enthalten¹⁸. Auf diesen Annahmen aufbauend werden nun die drei Kriterien näher erläutert: (1) Information vorhanden, (2) Extraktion durch NLP möglich und (3) Bewertung des NLP-Potentials.

Das Kriterium **(1) (Teil)Information in der Rezension vorhanden** bewertet, ob die in der Methode verwendeten Informationen und Inhalte in Kundenrezensionen thematisiert werden. Wenn diese Information in Rezensionen beschrieben werden, dann wurde der Wert *Ja* zugewiesen. Wenn die Informationen aus Rezensionen verschiedener Produkte extrahiert werden müssen, wurde der Wert *Produktvarianten* vergeben. Der Wert *Ausschnittsweise* wurde vergeben, wenn in den Bewertungstexten sinnvollerweise nur Teile der Methode angesprochen werden.

Das Kriterium **(2) Extraktion durch NLP möglich** schätzt auf Basis bestehender NLP-Methoden wie zielführend die Wissensextraktion sein würde. Neben dem Wert *Ja*, wurde auch der Wert *fraglich* vergeben. Eine zusätzliche Erklärung bedarf das vergebene Merkmal *fraglich* für die Methode Lead-User. Unter der Annahme, dass Lead-User Rezensionen verfassen, ist es aktuell für NLP-Algorithmen *fraglich* diese als Lead-User-Rezension zu identifizieren. So folgt für die Methode des Lead-Users, die Annahme, dass auch diese Personengruppen Rezensionen verfassen. Die Identifikation eines Lead Users ist jedoch eine Herausforderung für aktuelle NLP-Algorithmen. Daher wurde hier der Wert *fraglich* für das Merkmal (2) vergeben.

Das letzte Kriterium **(3) Bewertung des NLP-Potentials** berücksichtigt die Reduzierung des Aufwands bei der Methodendurchführung und die Verbesserung der Methode durch den zusätzlichen Informationsgewinn. Für eine bessere Veranschaulichung wurden sogenannte Harvey Balls, das sind kreisförmige Ideogramme, verwendet. Je ausgefüllter die Kreise, desto größer ist die Erleichterung bei der Methodenumsetzung und/oder eine große Verbesserung der Methode.

Das Ergebnis dieser Untersuchung zeigt, dass alle bestehenden Methoden einen hohen bis sehr hohen Aufwand bei der Methodendurchführung aufweisen. Die Untersuchung von Produktrezensionen mittels NLP-Methoden bietet das Potential diesen Aufwand stark zu reduzieren. Durch diese Reduktion werden die Arbeitsbereiche im Marketing, der technischen Produktentwicklung und des Designs gestärkt.

Im nächsten Kapitel werden aktuelle Herausforderungen anhand ausgewählter Veröffentlichungen im Detail beschrieben.

¹⁸ Eine breite wissenschaftliche Studie sollte diese Annahme nochmal im Detail evaluieren.

Tabelle 4:

Übersicht Methodenkatalog zur Kundenintegration und Potentialanalyse zur Informationsanreicherung der Methoden auf Basis NLP extrahierter nutzergenerierten Inhalten

	Zusätzliche Erklärung der Methode			Potentialanalyse für eine NLP-Erweiterung			
	Kurzbeschreibung	Aufwand	Verwendete Informationen/ Ressourcen	(Teil)Information in Rezensionen vorhanden	Extraktion durch NLP möglich	Bewertung des NLP-Potentials	
Methoden zum besseren Verständnis des Kunden	Vor-Ort-Besuche bei Anwendern	<ul style="list-style-type: none"> - Semi-Strukturiertes Vorgehen - Beobachtung und Fragen - Produktnutzung - Zusätzliche Erklärungen - Nachfragen 	●	<ul style="list-style-type: none"> - Kontext, Umfeld - Produktnutzen, andere Produkte, Anwender - Probleme 	X	X	●
	Experteninterviews und Anwenderbefragung	<ul style="list-style-type: none"> - Semi-Strukturiertes Vorgehen - Fragen (spezifische und offene) - Interviewleitfaden 	●	<ul style="list-style-type: none"> - Experte - Fragenkatalog/Leitfaden - Antworten des Experten 	X	X	●
	Handlungsanalysen, Beobachtungsprotokoll, SUS – System Usability Scale	<ul style="list-style-type: none"> - Beschreibung einer zu erfüllenden Aufgabe mit dem Produkt - Beobachtung der Arbeitsabläufe - Aufgabenerfüllung (Varianten) - Nachträgliche Bewertung der Tätigkeit 	●	<ul style="list-style-type: none"> - Beobachtung (Video) - Kommentierung während des Vorgangs - Teilschritte zur Erfüllung - Nachträgliche Bewertung 	(X) ¹	X	●
	Lead-User Methode	<ul style="list-style-type: none"> - Lead-User zur Identifikation wichtiger Markttrends & Produktmerkmale - Integration in den Entwicklungsprozess 	●	<ul style="list-style-type: none"> - Lead-User - Produktmerkmale - Markttrends 	X	fraglich	●
	Conjoint-Analyse	<ul style="list-style-type: none"> - Struktur der Präferenz der Konsumenten schätzen 	●	<ul style="list-style-type: none"> - Gesamturteil aus Alternativen (Stimuli) - Ausprägungen verschiedener Merkmale (Eigenschaften) 	X ²	X	●
	Menu-Based-Choice-Verfahren (MBC)	<ul style="list-style-type: none"> - Preis und Merkmal Abhängigkeiten beim Kunden simulieren und vorhersagen 	●	<ul style="list-style-type: none"> - Auswahlprozesse und Präferenzanalyse 	X ²	fraglich	●
	Kundenprofile, Persona und User Needs	<ul style="list-style-type: none"> - repräsentative, aber fiktive Nutzer erstellt - in Bezug zu dem Produkt und der relevanten Umwelt 	●	<ul style="list-style-type: none"> - Demografische Daten, Psychografisch, Verhalten und Geografisch Merkmale 	(X)	X	●
	User Journey (Map)	<ul style="list-style-type: none"> - ganzheitliche Betrachtung des Kunden mit den Produkt, Nutzerziele und Anforderungen können daraufhin abgeleitet werden 	●	<ul style="list-style-type: none"> - Phase, Schritte, Vorgehen, Emotionale Kurve, Emotionen, Touchpoints 	(X)	X	●
	House-of-Quality	<ul style="list-style-type: none"> - Kundenanforderungen in Qualitäts- und Produktmerkmale zu überführen 	●	<ul style="list-style-type: none"> - Produktmerkmale, Bewertungen - Funktionen und technische Aspekte 	X	X	●
	Empathy Map	<ul style="list-style-type: none"> - bekanntes Wissen über den Nutzer zu externalisieren 	●	<ul style="list-style-type: none"> - was der Nutzer sagt, denkt, tut und fühlt 	X	X	●

○ nicht vorhanden	◐ gering	◑ moderat	◒ hoch	● sehr hoch
-------------------	----------	-----------	--------	-------------

Legende:

1 Ausschnittsweise und Retrospektiv

2 Produktvarianten notwendig

3.3 Online Bewertungen als Forschungsgegenstand

Nutzergenerierte Inhalte und dabei vor allem Online-Rezensionen sind ein sehr ausgereiftes Forschungsfeld. Die international am meisten publizierten Beiträge zu diesem Thema stammen aus den Bereichen Business Economics, Computer Science, Social Science, Information Science und Engineering¹⁹. Diese Arbeit kann weder einen allumfassenden Stand der Wissenschaft aller Bereiche, noch in ihren Wechselwirkungen beschreiben. Um die in dieser Arbeit vorgestellten Erkenntnisse und Vorgehen in den wissenschaftlichen Kontext einzuordnen, beschränkt der Autor sich auf den Forschungsbereich des *Engineerings* und betrachtet dabei ausschließlich die *Design*-Perspektive. Begründet werden diese Einschränkungen durch die Entscheidung eine Lösung für Produktentwickler zu erarbeiten, deren Wirkungsumfeld die (technische) Produktgestaltung darstellt.

Die für diese Arbeit wichtigen Forschungsaktivitäten (Modellierungsansätze, Methoden und Forschungsziele) werden durch eine detaillierte Beschreibung, Analyse und Synthese anhand relevanter identifizierter Primärstudien erläutert (vgl. Tabelle 5).

Tabelle 5: Übersicht der analysierten Publikationen

Nr.	Titel	Autoren	Journal	Erscheinungsjahr
1	Prioritising engineering characteristics based on customer online reviews for quality function deployment	Jin, Jian; Ji, Ping; Liu, Ying	JOURNAL OF ENGINEERING DESIGN	2014
2	A multi-objective PSO approach of mining association rules for affective design based on online customer reviews	Jiang, Huimin; Kwong, C. K.; Park, W. Y.; Yu, K. M.	JOURNAL OF ENGINEERING DESIGN	2018
3	Predicting Future Importance of Product Features Based on Online Customer Reviews	Jiang, Huimin; Kwong, C. K.; Yung, K. L.	JOURNAL OF MECHANICAL DESIGN	2017
4	A Systematic Methodology Based on Word Embedding for Identifying the Relation Between Online Customer Reviews and Sales Rank	Suryadi, Dedy; Kim, Harrison	JOURNAL OF MECHANICAL DESIGN	2018
5	Mitigating Online Product Rating Biases Through the Discovery of Optimistic, Pessimistic, and Realistic Reviewers	Lim, Sunghoon; Tucker, Conrad S.	JOURNAL OF MECHANICAL DESIGN	2017
6	Mining Changes in User Expectation Over Time From Online Reviews	Hou, Tianjun; Yannou, Bernard; Leroy, Yann; Poirson, Emilie	JOURNAL OF MECHANICAL DESIGN	2019
7	Extracting Customer Perceptions of Product Sustainability From Online Reviews	El Dehaibi, Nasreddine; Goodman, Noah D.; MacDonald, Erin F.	JOURNAL OF MECHANICAL DESIGN	2019

Entsprechend der Nummerierung in Tabelle 5 werden die Veröffentlichungen nacheinander beschrieben und anschließend anhand der Modell-Elemente, Forschungsfragen, Methoden und Programmierabläufe diskutiert.

1) Die Autoren Jin et al. widmen sich der Untersuchung, technische Produktmerkmale auf Basis von Online-Kundenrezensionen zu priorisieren. In der Publikation aus dem Jahr 2014 kommen die Autoren zu dem Ergebnis, dass eine Priorisierung durch schriftliche Kundenmeinungen zu Produktmerk-

¹⁹ Die englischen Bezeichnungen der Kategorien werden häufiger verwendet. Die deutsche Übersetzung in der gleichen Reihenfolge: Betriebswirtschaft, Informatik, Sozialwissenschaft und Ingenieurwesen

malen möglich ist (Jin et al. 2014, S. 320). Die verwendete Methode nutzt einen ordinalen Klassifikationsansatz und einen paarweisen Vergleich. Dieser vergleicht paarweise wie die Gewichte der technischen Merkmale aus Bewertungspaaren erlernt werden. Außerdem wird ein ganzzahliges lineares Optimierungsproblem formuliert, um die paarweisen Ergebnisse in die ursprünglichen Niveaus der Kundenzufriedenheit zu transformieren [ebd.]. Die Methode wird durch eine explorative Fallstudie für Kundenrezensionen über Farbdrucker erprobt. Der untersuchte Datensatz bestand aus 770 Bewertungen zu vier verschiedenen Farbdruckern und wurde von zwei Webseiten (Amazon.com und Epson.com) entnommen. Die Autoren versuchen mit diesem Ansatz, Kundenrezensionen direkt für die Produktgestaltung nutzbar zu machen, indem zum Beispiel das House-of-Quality einfacher und direkter durch dieses Modell erstellt werden kann (Jin et al. 2014, S. 304–305).

2) Die Autoren Jiang et al. widmen sich der Untersuchung von Assoziationsregeln für affektives Design auf Basis von Kundenrezensionen. In der Publikation aus dem Jahr 2018 kommen die Autoren zu dem Ergebnis, dass es möglich ist Designattribute mit einer Logik aus Kundenrezensionen zu verknüpfen (Jiang et al. 2018). Die in dem Beitrag verwendete Methode zur Generierung von Assoziationsregeln für das affektive Design auf der Grundlage von Online-Kundenrezensionen nutzt Opinion Mining unter Verwendung von Online-Kundenrezensionen und Assoziationsregel-Mining auf der Grundlage eines multi-objektiven Ansatzes der Partikelschwarmoptimierung (PSO). Die Methode wird durch eine Fallstudie im Bereich Fahrzeuge erprobt. Der untersuchte Datensatz bestand aus 1259 Bewertungen zu sechs verschiedenen Fahrzeugdesigns. Die Autoren versuchen mit diesem Ansatz Kundenrezensionen direkt für die Produktgestaltung nutzbar zu machen, indem beispielsweise die Einträge des Morphologischen Kastens auf Basis des Modells miteinander verknüpft werden können. (Jiang et al. 2018)

3) Die Autoren Jiang et al. widmen sich der Vorhersage der Bedeutung von Produkteigenschaften durch Kundenrezensionen. In der Publikation aus dem Jahr 2017 kommen die Autoren zu dem Ergebnis, dass es möglich ist die Gewichtung der Bedeutung von Produkteigenschaften vorherzusagen (Jiang et al. 2017). Die in dem Beitrag verwendete Methode zur Vorhersage verwendet Text und Opinion Mining um die Online-Bewertungen zu analysieren und Produktmerkmale zu extrahieren. Eine Fuzzy-Inferenz-Methode wird verwendet, um die Wichtigkeitsgewichtung von Produktmerkmalen zu bestimmen, indem sowohl Häufigkeiten als auch Stimmungswerte (sentiment score) aus dem Opinion Mining verwendet werden. Die Vorhersage der zukünftigen Wichtigkeit von Produktmerkmalen wird mittels einer Fuzzy-Zeitreihenmethode erreicht [ebd.]. Die Methode wird durch eine Fallstudie für Haushaltsgeräte validiert. Der verwendete Datensatz bestand aus Rezensionen aus dem Zeitraum von 2010-2013 und wurde in acht diskrete Zeitspannen unterteilt, um einen zeitlichen Verlauf abzubilden. (Jiang et al. 2017).

4) Die Autoren Suryadi und Kim untersuchen den Zusammenhang zwischen dem Verkaufsrang von Produkten und den Kundenrezensionen. In der Publikation aus dem Jahr 2018 kommen die Autoren zu dem Ergebnis, dass es einen starken Zusammenhang zwischen dem Verkaufsrang und der Rezension existiert. Insbesondere sind die im Titel genannten Produktmerkmale entscheidend für den Verkaufsrang. Die in dem Beitrag verwendete Methode nutzt einen kombinierten Ansatz aus POS, Word Embedding (word2vec) und X-means Clustering. In einem ersten Schritt werden Produktmerkmalswörter erzeugt. Es folgt die Identifizierung von Stimmungswörtern und deren Intensität und die Bestimmung der Verbindung von Wörtern aus dem Abhängigkeitsbaum. Die Methode schließt mit der Verknüpfung von Variablen aus den Rezensionen mit dem Verkaufsrang eines Produktes durch ein Regressionsmodell ab. Die Fallstudie aus dem Bereich der elektronischen Konsumgüter belegt diesen Zusammenhang. Der verwendete Datensatz basierte auf 2 Produktgruppen: Laptops und Wearables²⁰. Insgesamt wurden 1990 Datenpunkte für 35 verschiedene Wearables und 5587 Datenpunkte für 84 verschiedene Laptopvarianten untersucht. (Suryadi und Kim 2018)

5) Die Autoren Lim und Tucker untersuchen die Verbesserung von Produktbewertungen durch Klassifizierung von Rezensenten. In der Publikation aus dem Jahr 2017 kommen die Autoren zu dem Ergebnis, dass eine angepasste Sternbewertung (auf Basis der Historie der Rezensenten) die Produktqualität besser repräsentiert als die ursprüngliche Bewertung der Rezensenten. Dabei wurden Rezensenten-Gruppen identifiziert, insbesondere pessimistische und optimistische Rezensenten. Die in dem Beitrag verwendete Methode normiert die Sterne-Bewertungen. Mittels eines Minimum-Distanz-Klassifikators werden die Rezensenten in realistisch, optimistisch, pessimistisch, unzuverlässig oder unentschieden kategorisiert. Basierend auf ihren Bewertungshistorien und Produktverkaufsrängen. Schließlich bildet die vorgeschlagene Methode optimistische/pessimistische Kundenbewertungen in realistische Bewertungen ab. Der Beitrag nutzt Text-Mining nur zur Überprüfung der Modell-Ergebnisse. Die Fallstudie aus dem Bereich der elektronischen Konsumgüter auf Amazon.com belegt den Zusammenhang. Der verwendete Datensatz basierte auf 739 Produkten von 27.446 Rezensenten. Insgesamt wurden 1.689.188 Produktrezensionen untersucht. (Lim und Tucker 2017)

6) Die Autoren Hou et al. untersuchen die zeitlichen Veränderungen von Nutzererwartungen aus Online Rezensionen. In der Publikation aus dem Jahr 2019 kommen die Autoren zu dem Schluss, dass es möglich ist Produktverbesserungen abzuleiten. Die verwendete Methode identifiziert und strukturiert automatisch Produktmerkmale aus Bewertungstexten mithilfe von Text-Mining. Auf Basis des Kano-Modells, das Präferenzen von Produktattributen in fünf Kategorien klassifiziert, wird eine

²⁰ Wearables sind Produkte, die am Körper getragen werden und Computersysteme beinhalten (Smart Watch).

Conjoint-Analyse verwendet, um die strukturierten Anforderungen der Produktmerkmale (Affordanzen) quantitativ zu kategorisieren. Mithilfe der Conjoint-Analyse werden für zwei aufeinanderfolgende Produktgenerationen veränderte Kundenanforderungen identifiziert. Die Fallstudie verwendete einen Datensatz aus 102.463 Reviews aus den Jahren 2013-2018 zu Kindle E-Book-Readern. Das Besondere an dem Datensatz ist die Berücksichtigung von zwei verschiedenen Produktgenerationen. (Hou et al. 2019)

7) Die Autoren El Dehaibi et al. untersuchen den Zusammenhang zwischen wahrgenommenen und realen Nachhaltigkeitsfaktoren in Online Rezensionen. In der Veröffentlichung aus dem Jahr 2019 kommen die Autoren zu dem Ergebnis, dass die zwei Perspektiven oftmals gegensätzlich sind. Die verwendete Methode nutzt händisch gegabelte Textabschnitte, welche nachträglich durch Laien Produktmerkmalen zugeordnet werden. Anschließend wird mittels NLP (Sentiment-Analyse) ein Stimmungswert ermittelt. Der Bewertungen zu den einzelnen Produktmerkmalen werden dann durch eine lineare Regressions-Analyse ausgewertet und schlussendlich mit Nachhaltigkeitswerten einer LCA²¹ verglichen. Die Fallstudie verwendete einen Datensatz aus 1474 Amazon-Rezensionen zu einer Kaffeemaschine der Art French. Dabei wurde für die Analyse die Rezensionen in jeweils 300 Rezensionen pro Nachhaltigkeitsaspekt (sozial, ökonomisch und ökologisch) gegliedert.

3.4 Zusammenfassung, Bewertung und Abgrenzung zum Stand der Technik

Die Auswertung nutzergenerierter Inhalte für die Produktentwicklung ist ein vielfältig bearbeitetes Thema. Abbildung 17 stellt sowohl die Bewertung der vorher beschriebenen Veröffentlichungen anhand verschiedener Merkmale dar, als auch die Zielvorgabe des neuen Ansatzes (rechte Spalte: **Neuer Ansatz**). Die Bewertungskriterien unterteilen sich in drei Hauptkategorien: **Ingenieurwissenschaftliche Betrachtungsweise**, **Sprachbasierte Auswertung** und **Nutzung und Implementierung**. Für eine bessere Veranschaulichung wurden wieder *Harvey Balls* verwendet. Die Analyse der Bewertungen rechtfertigen folgende Aussagen:

Die Kategorie **Ingenieurwissenschaftliche Betrachtungsweise** kann von keinem der untersuchten Ansätze vollständig erfüllt werden. Um einen Vorteil gegenüber bestehenden Ansätzen zu erreichen, müssen diese Merkmale vollständig von der neuen Methodik erfüllt werden.

In der Kategorie **Sprachbasierte Auswertung** sind die Aussagen differenzierter herzuleiten. Dafür werden die einzelnen Merkmale erläutert.

²¹ Life-Cycle-Assessment oder Ökobilanz

Eine **Stimmungsmodellierung** ist bei fast allen Methoden eine Basisanforderung, somit auch für diese Arbeit, nicht zuletzt um eine Vergleichbarkeit der verschiedenen Ansätze zu ermöglichen.

Die **Satzbaumodellierung** wurde nur in einer Studie eingehend untersucht und wird daher nicht weiter betrachtet.

Alle Publikationen (bis auf eine) verwenden einen **manuellen Datenaufbereitungsprozess**. Und es zeigt sich, dass keines der verwendeten Vorgehen methodisch (aus der Produktentwicklungssicht) unterstützt ist. Daraus leitet sich ab: Der neuartige Ansatz wird ebenfalls einen manuellen Datenaufbereitungsprozess verwenden. Das Novum wird eine aus Produktentwicklungssicht methodische Vorgehensbeschreibung sein.

Die **automatisierte Klassifizierung** stellt bei allen Publikationen den stärksten Forschungsschwerpunkt dar. Daher wird hier in der vorliegenden Arbeit bewusst auf den Teil verzichtet, da nicht abzuschätzen ist, welcher Algorithmus-basierte Ansatz sich durchsetzen wird.²²

Die Kategorie **Nutzung und Implementierung** lässt den Schluss zu, dass die untersuchten Ansätze nicht auf eine breite Nutzung der Modelle ausgelegt sind. Der modulare Aufbau ist bei allen Publikationen teilweise erfüllt, da oftmals externe Software oder Algorithmen-Bibliotheken in den Ansatz integriert sind. Daraus folgt für einen praxisrelevanten und plattformbasierten Ansatz: Das Modell soll als Softwarelösung frei nutzbar sein, es besitzt einen modularen Aufbau, ist einfach zu bedienen (gute Usability) und ist für eine webbasierte Nutzung ausgelegt.

Ausgehend von dieser Bewertung und den hier dargestellten Defiziten werden im nächsten Kapitel der Handlungsbedarf und die Ziele der vorliegenden Arbeit erläutert.

²² Anmerkung: Die Bereitstellung des Modells wird so erfolgen, dass die Integration einer automatisierten Klassifizierung durch verschiedene Algorithmen möglich ist.

Abbildung 17: Gegenüberstellung der betrachteten wissenschaftlichen Ansätze und Einordnung der vorliegenden Arbeit

		(Ingenieurs-) Wissenschaftlich etablierte Ansätze							Neuer Ansatz	
		Jin et al. (2014)	Jiang et al. (2017)	Lim und Tucker (2017)	Jiang et al. (2018)	Suryadi und Kim (2018)	Hou et al. (2019)	El Dehaibi et al. (2019)		
Bewertungen	Ingenieurwissenschaftliche Betrachtungsweise	Produkt-Modellierung (Features)	●	●	○	●	●	●	●	●
		Kunden-Modellierung (z.B. Wahrnehmung/Einschätzung)	○	◐	● ¹	○	○ ²	◐	◐	●
		Ergebnisse direkt nutzbar für die Produktentwicklung	◐	◐	○	●	○	●	◐	●
		Bezug zu etablierten Produktentwicklungswerkzeugen	◐	○	○	●	○	●	◐	●
	Sprachbasierte Auswertung	Stimmungs-Modellierung (Text-Mining)	●	●	○ ³	●	●	●	●	●
		Satzbau-Modellierung (Bestandteile, Verbindungen)	○	○	○	○	●	○	○	○
		Automatisierungsgrad bei der Datenvorbereitung ⁴ (von manuellem Labeln bis automatisiert)	○	○	○	○	●	◐	○	○
		Methodisches Vorgehen des manuellen Vorgangs	○	○	○	○	○	○	○	●
		Automatisierte Klassifizierung bzw. Automatisierte Datenanalyse (Algorithmus-basiert)	●	●	○	●	●	●	●	○
	Nutzung und Implementierung	Modell als Softwarelösung frei verfügbar ⁵	○	○	○	○	○	○	○	●
		Modularer Aufbau	◐	○	◐	◐	◐	◐	◐	●
		Bedienbarkeit (Usability)	○	○	●	○	○	○	○	●
		webbasierte Nutzung	○	○	○	○	○	○	○	●
			¹ als Rezensent nicht als Produktnutzer ² nur implizit ³ zur Überprüfung der Ergebnisse [nicht Teil der verwendeten Methodik] ⁴ manuelles Labeling - Kreisring & automatisierte Lösung - Vollkreis ⁵ die verwendeten Algorithmen werden hierbei nicht berücksichtigt							

4 Handlungsbedarf und Zielsetzung

Der Stand der Technik und Forschung zur Untersuchung nutzergenerierter Textinformationen für die Produktentwicklung wurde in den vorhergehenden Kapiteln ausführlich beschrieben. Das ist die Ausgangssituation für das weitere Vorgehen. In diesem Kapitel werden auf dieser Basis erst der Handlungsbedarf und dann die Ziele für die vorliegende Arbeit abgeleitet.

4.1 Handlungsbedarf

Der Handlungsbedarf leitet sich aus der detaillierten Analyse und der anschließenden Bewertung in Kapitel 3 ab. Der qualitative Vergleich in Abbildung 17 zeigt einerseits die identifizierten Defizite und andererseits die für die vorliegende Arbeit zu erfüllenden Kriterien. Der Handlungsbedarf wird der besseren Nachvollziehbarkeit wegen in vier Kategorien unterteilt. Die Herleitung dieser Kategorien basiert auf der Kombination von schon beschriebenen Nutzerinformationen im Produktentwicklungskontext, insbesondere der in Abbildung 6 aufgeführten Aspekte (Informationsgrundlage, Analyseziele und datenbasierte Methoden) und der für die Detailanalyse verwendeten Merkmale in Abbildung 17 (Ingenieurwissenschaftliche Betrachtungsweise, sprachbasierte Auswertung und Nutzung und Implementierung). Daraus ergeben sich die folgenden Kategorien: Datenaufbereitung, -analyse und -visualisierung sowie die Nutzung und Implementierung. Deren Herleitung soll nun kurz beschrieben werden, um anschließend den Handlungsbedarf für jede einzelne Kategorie abzuleiten.

Die Kategorie Datenaufbereitung bündelt die ingenieurwissenschaftliche Perspektive (vgl. Abbildung 17) auf die Informationsgrundlagen (Kundenbedürfnisse, Kaufverhalten, Produkteigenschaften) (vgl. Abbildung 6). Weiterhin werden ebenfalls die Analyseziele (Produktverbesserung und Abhängigkeiten zwischen Kundschaft und Produkten) (vgl. Abbildung 6) innerhalb dieser Kategorien berücksichtigt. So wird innerhalb dieser Kategorien beschrieben, wie Ingenieur:innen in der Produktentwicklung nutzergenerierte Texte erstellen sollten und mit welchem Ziel dies geschieht.

Unter der Kategorie Datenanalyse, werden die verglichenen Merkmale der sprachbasierten Auswertung (vgl. Abbildung 17) und die datenbasierte Methoden (vgl. Abbildung 6) zusammengefasst.

Die letzten Kategorien Datenvisualisierung und Nutzung und Implementierung ergeben sich durch das Ziel einer Anwendung im Produktentwicklungsprozess. Durch diese beiden Kategorien soll der Handlungsbedarf auf eine praktische Umsetzung der theoretisch-wissenschaftlichen Erkenntnisse gerichtet werden.

Der für die Arbeit abgeleitete Handlungsbedarf wird im Folgenden anhand der vier hergeleiteten Kategorien beschrieben: **Datenaufbereitung**, **-analyse** und **-visualisierung** sowie die **Nutzung und Implementierung** des Modells.

Datenaufbereitung

Die Kategorie Datenaufbereitung führt die ingenieurwissenschaftliche Perspektive (vgl. Abbildung 17) auf die Informationsgrundlagen (Kundenbedürfnisse, Kaufverhalten, Produkteigenschaften) (vgl. Abbildung 6) zusammen und beschreibt somit, wie Ingenieur:innen in der Produktentwicklung nutzergenerierte Texte aufbereiten sollten. Die sieben ausgewählten Veröffentlichungen belegen, dass die Phase der Datenaufbereitung einen wesentlichen Anteil und wichtigen Schritt in der gesamten Datenverarbeitung einnimmt. Umso erstaunlicher ist es, dass diese Phase wenig bis kaum beachtet wird. Die für die Datenaufbereitung verwendeten Ansätze lassen sich in zwei Kategorien einteilen: *automatisierte* und *manuelle* Ansätze. Die automatisierten Ansätze verwenden verschiedene Algorithmen, die aufwendig auf die Anwendung abgestimmt werden müssen. Für die händischen Lösungsvorschläge werden die Texte gelesen und ohne Softwareinsatz bearbeitet. Das bedeutet, der Rezensionstext wird von einer Person gelesen und nach im Vorhinein festgesetzten Regeln klassifiziert. Oftmals geschieht diese Klassifizierung in dem Textpassagen Schlagwörtern (Label) zugeordnet werden. Die vorangestellte Analyse verdeutlicht, dass der Vorgang des Labelns²³ einen großen Einfluss auf das Ergebnis der Textauswertung hat. Dennoch existiert keine ausgearbeitete Vorgehensweise für diesen Prozess. Oftmals fehlen Fachleute für die aufwendige Anpassung der Parameter der Algorithmus-basierten Ansätze. Daher wird der Handlungsbedarf für diese Arbeit auf die **methodische Unterstützung des Labeling-Vorgangs** beschränkt. Dieser Ansatz hat den Vorteil, dass das Wissen der Produktentwicklung in den kritischen Prozess der Datenaufbereitung berücksichtigt wird und somit das Ergebnis direkt für die Produktentwicklung nutzbar ist. Im Mittelpunkt stehen Produktverbesserungen und die Identifizierung von Zusammenhängen zwischen Kundenwahrnehmungen bzw. -bedürfnissen und dem Produkt, insbesondere den Produkteigenschaften, als Analyseziel. Wie das Ziel erreicht werden soll, wird unter dem nächsten Begriff der Datenanalyse beschrieben.

Datenanalyse

Die untersuchten Veröffentlichungen verdeutlichen, dass die vorgeschlagenen Verfahren für die Analyse der Daten sehr unterschiedlich sind. Die Ansätze unterscheiden sich in der Komplexität, den Zwischenschritten und den verschiedenen Algorithmen. Die Analyse der unstrukturiert vorliegenden Texte stellt dabei fast immer einen zentralen Bestandteil dar. Insbesondere die Kombination von quantitativen und qualitativen Merkmalen der linguistischen Datenverarbeitung. Aktuell ist nicht klar, welche Lösungen aus der Softwareentwicklung, Informatik oder Datenwissenschaft (Computer und Data Science) sich durchsetzen oder weiterentwickelt werden. Der Handlungsbedarf für die Datenanalyse beschränkt sich somit bei dieser Arbeit auf die **Adaption der vorhandenen natürlichen**

²³ Die Definition des Duden für labeln lautet: benennen oder betiteln. Diese Definition passt für den Anwendungsfall nicht vollständig. Der Begriff Label – als Substantiv - kann auch Schlagwort bedeuten und somit kann labeln auch als Verschlagworten verwendet werden. Verschlagworten bedeutet dann: Etwas (in dem Falle eine Textpassage) mit einem Schlagwort versehen. Der Begriff Label oder gelabelt wird aus Gründen der Übersichtlichkeit im Weiteren verwendet.

Sprachverarbeitungs-Algorithmen unter Berücksichtigung quantitativer und qualitativer Merkmale.

Datenvisualisierung

Die untersuchten Veröffentlichungen geben keine Lösungsvorschläge für eine adressatengerechte Visualisierung der Analyseergebnisse. Für eine ingenieurwissenschaftliche Aufbereitung müssen die **Ergebnisse der Analyse in Kennzahlen überführt und in Bezug zu dem Produkt dargestellt** werden. Dadurch wird eine datenbasierte Entscheidungsgrundlage geschaffen.

Nutzung und Implementierung des Modells

Wiewohl alle Veröffentlichungen anhand eines realen Datensatzes ihre Methode validiert haben, sind die beschriebenen Anwendungen und Methoden noch nicht praxisnah implementiert worden. Auch wenn der praktische Nutzen nicht im Mittelpunkt der wissenschaftlichen Veröffentlichungen steht, so ist doch festzustellen, dass alle vorgestellten Modelle sehr weit von einer praxisorientierten Umsetzung entfernt sind. Die Anzahl an verschiedenen digitalen Werkzeugen, Programmiersprachen und Softwarelösungen erschweren die Nachvollziehbarkeit und hindern den Wissenszuwachs in dem Bereich der Textauswertung für die Produktentwicklung. Oftmals werden verschiedene Domänen und Fachexpertisen benötigt um das Modell zu entwickeln. Das wird immer dann deutlich, wenn Autoren in den Veröffentlichungen ihren Dank Dritten aussprechen, z.B. für die Implementierung der Softwarelösung. Diese Diskrepanz zwischen Informationsgeber (Ingenieur:in) und Informationsverarbeiter (Software oder Entwickler) ist ein grundlegendes Problem und kann im Rahmen dieser Arbeit nicht gelöst werden. Dennoch ist die fehlende Anwendung ein wichtiges Motiv für diese Arbeit: **Es wird eine praxisnahe Implementierung des Modells angestrebt.**

4.2 Zielsetzung

Ziel dieser Arbeit ist es, ein praxistaugliches Modell für die Auswertung von nutzergenerierten Textdaten zu entwickeln und dieses in eine webbasierte Anwendung zu überführen. Des Weiteren soll hier ein Vorgehen beschrieben werden, welches es ermöglicht, die zukünftigen Entwicklungen der Informatik und Softwareentwicklung einfacher im Kontext der Produktentwicklung zu erproben und für die spezifischen Anwendungsfälle anzupassen. Dadurch werden sowohl Forschungsinstitute als auch Industrieunternehmen befähigt neuartige Technologien aus dem Bereich der Softwareentwicklung für Ihre Anwendungen in den Ingenieurwissenschaften nutzbar zu machen. Die Fähigkeit, bestehende Softwarelösungen im technischen Alltag zu nutzen, wird ein immer wichtigerer Faktor für die Wettbewerbsfähigkeit von Unternehmen sein. Aus diesem übergeordneten Ziel und unter Berücksichtigung der im Handlungsbedarf abgeleiteten Forschungsschwerpunkte werden folgende Teilziele formuliert:

- Teilziel 1:** Entwicklung einer Methode für Ingenieur:innen in der Produktentwicklung zur Beschriftung (labeln) von nutzergenerierten Textdaten.
- Teilziel 2:** Entwicklung einer Logik zur Auswertung gelabelter Textdaten durch Kombination bestehender quantitativer und qualitativer linguistischer Algorithmen.
- Teilziel 3:** Erstellung eines kontextspezifischen und anforderungsgerechten Visualisierungskonzepts der Analyseergebnisse.
- Teilziel 4:** Praxisnahe prototypische Implementierung der Teilziele 1-3.
- Teilziel 5:** Validierung des Gesamtmodells anhand einer realen Fallstudie.

Im Folgenden wird die Vorgehensweise zum Erreichen der Teilziele kurz beschrieben.

Teilziel 1: Entwicklung einer Methode für Ingenieur:innen in der Produktentwicklung zur Beschriftung (labeln) von Textdaten

Unter der Berücksichtigung einer anschließenden Analyse der Textdaten wird auf Basis traditioneller Werkzeuge aus der Produktentwicklung ein Verfahren zum Labeln von Textdaten abgeleitet. Die vorgeschlagene Methode hat das Ziel, die bereits vorhandenen Informationen aus der Produktentwicklung systematisch in potentielle Label zu überführen. Produktarchitektur, Lebenszyklusphasen und User-Experience bilden dabei die theoretische Grundlage der Methode. Die Methode bezweckt Produktentwickler:innen zu befähigen, nutzergenerierte Texte nach produktspezifischen Anforderungen zielorientiert und vergleichbar aufzubereiten. Ziel der systematischen Aufbereitung der Informationen ist eine signifikante Qualitätssteigerung der Analyseergebnisse.

Teilziel 2: Entwicklung einer Logik zur Auswertung gelabelter Textdaten durch Kombination bestehender quantitativer und qualitativer linguistischer Auswertalgorithmen.

Basierend auf dem in Kapitel 3 beschriebenen Stand der Technik zu bestehenden Ansätzen wird eine Verarbeitungslogik abgeleitet, indem die einzelnen Methoden zur Datenverarbeitung und –analyse verglichen, bewertet und anschließend ausgewählt werden. Die Vergleichsparameter werden aus dem Handlungsbedarf und der Zielsetzung abgeleitet. Darauf aufbauend werden die einzelnen Teilschritte zu einem Ablaufdiagramm zusammengeführt. Die verwendeten Algorithmen werden durch Python-Software-Code erläutert.

Teilziel 3: Erstellung eines kontextspezifischen und anforderungsgerechten Visualisierungskonzepts der Analyseergebnisse.

Aus der Beschreibung in Kapitel 3 zum Stand der Technik zeigt sich sehr deutlich, dass aktuell keine Untersuchungen oder Praxisbeispiele zur Visualisierung der Analyseergebnisse existieren. Für die Entwicklung einer Lösung wird ein pragmatischer Ansatz gewählt, der die Anforderungen aus der Sicht einer praxisorientierten Produktentwicklung definiert und diese dann in Form einer grafischen Benutzeroberfläche (dashboard) implementiert. Das Ziel des Dashboards soll es sein, die Ergebnisse der Analyse in einer Form darzustellen, die es erlaubt, sie unmittelbar in den frühen Phasen des Produktentwicklungsprozesses als Informationsbasis zu nutzen.

Teilziel 4: Praxisnahe prototypische Implementierung der Teilziele 1-3.

Das Gesamtmodell wird in ein webbasiertes Framework (Django) überführt. Neben der Implementierung und Bereitstellung [Methode: Phasen der Software-Entwicklung – Deployment] auf einem externen Server über eine Linux-Partition werden hier die Strukturen für die Datenverarbeitung und die zugrundeliegenden Datenbanken erläutert. Dieses Kapitel der Arbeit ist nicht für Software-Entwickler geschrieben, sondern vielmehr soll hier ein Verfahren für Ingenieur:innen beschrieben werden, um mit *einfachen* Mitteln eine skalierbare und praxisnahe Softwareumgebung aufzubauen.

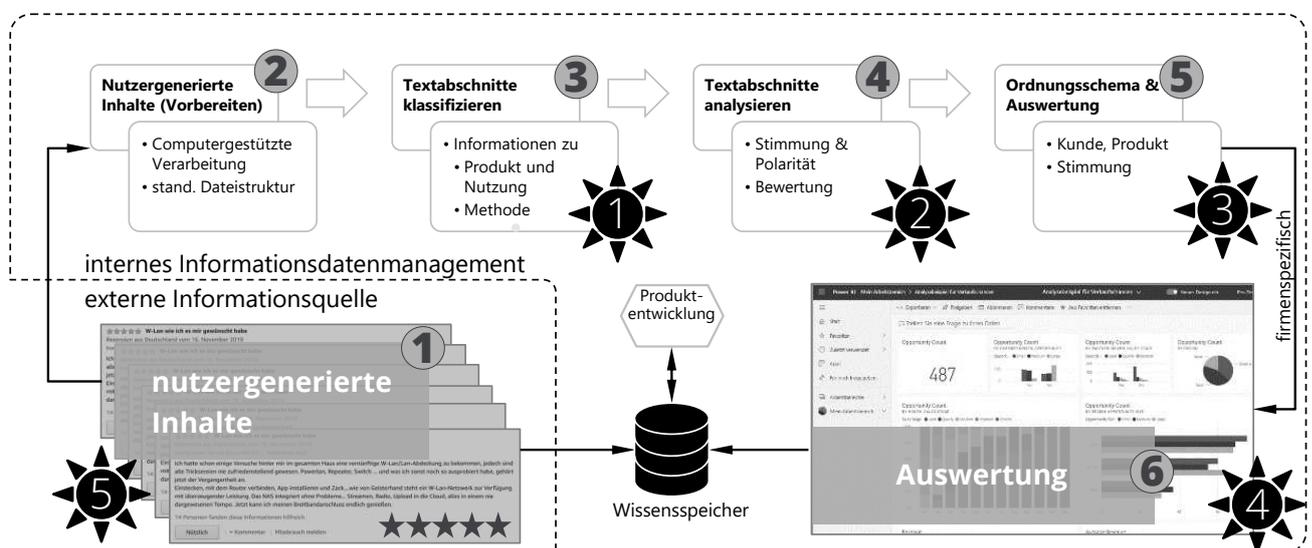


Abbildung 18: Einordnung der formulierten Teilziele (Sterne 1-5) in den Gesamtprozess (Kreise 1-6) für die Auswertung von nutzergenerierten Inhalten.

Teilziel 5: Validierung des Gesamtmodells anhand einer realen Fallstudie

Die in den Teilzielen 1-4 gewonnenen Informationen werden anhand einer Fallstudie über Rezensionen von Handwerkzeugen von der Website Amazon.de illustriert. Die experimentelle Studie soll nachweisen, dass der hier beschriebene Ansatz geeignet ist, um nutzergenerierte Inhalte für die Produktentwicklung auszuwerten. Dieser Nachweis wird durch die Erfüllung der einzelnen Modellelemente

erbracht: Vorgang des Text labelns, Algorithmus-basierte Auswertung, Visualisierung und die webbasierte Implementierung.

Zusammenfassung

Den Gesamtprozess für die Auswertung nutzergenerierter Inhalte und die Einordnung der formulierten Teilziele stellt Abbildung 18 dar. Die logische Abfolge besteht aus sechs Teilschritten, beginnend mit dem nutzergenerierten Inhalt (1), der Datenvorbereitung (2), der Klassifizierung der Textabschnitte (3), der anschließenden Analyse (4), der Überführung in ein Ordnungsschema (5) und der finalen Auswertung und Visualisierung der Ergebnisse (6). Die im vorherigen Abschnitt entwickelten Teilziele (Sterne in Abbildung 18) sind den Zwischenschritten zugeordnet. Die Integration der Auswertung und Visualisierung in das interne Unternehmensnetzwerk (Wissensspeicher oder –datenbank) wird nicht berücksichtigt.

5 Entwicklung des Modells

Unter Berücksichtigung der Ausgangssituation wurden verschiedene Teilziele formuliert, um die Defizite in der Auswertung nutzergenerierter Inhalte für die Produktentwicklung zu verbessern. In einem ersten Schritt werden für die identifizierten Defizite Lösungen entwickelt. Diese Teillösungen werden dann in einem zweiten Schritt zu einem Gesamtmodell zusammengeführt. Die wichtigsten drei Teillösungen sind: (1) Entwicklung einer Methode zur Beschriftung von nutzergenerierten Textdaten, (2) Entwicklung einer Logik zur Auswertung gelabelter Textdaten und (3) Erstellung eines Visualisierungskonzepts. Vor der Lösungsfindung werden immer die Anforderungen für die Teilziele abgeleitet. Für die (1) Methode zur Beschriftung ist der Fokus auf der Produktentwicklungsperspektive und auf der Möglichkeit, Informationen aus dem Produktdatenmodell flexibel und anpassbar in den Datenaufbereitungsprozess zu integrieren. Die größte Herausforderung für die Entwicklung einer (2) Logik zur Auswertung der Textdaten ist die praxisrelevante Nutzbarmachung bestehender Software-Algorithmen aus dem NLP-Umfeld. Aufbauend auf diesen Ergebnissen wird eine webbasierte Umsetzung des Gesamtmodells (Kapitel 6) und seine anschließende Validierung anhand einer realen Fallstudie (Kapitel 7) beschrieben.

5.1 Entwicklung einer Methode zur Beschriftung von nutzergenerierten Textdaten

Wie können Ingenieur:innen in der Produktentwicklung befähigt werden, Textdaten zu beschriften und was soll damit erreicht werden? Es ist davon auszugehen, dass die Textdaten der Nutzer verschiedene Informationen beinhalten können. Diese Informationen betreffen: Das Produkt und seine Produkteigenschaften oder –merkmale, deren Ausprägung, die Nutzung (oder Verwendung), den Kaufprozess, den Preis, die Informationen zum Anbieter oder der Marke des Produktes und viele weitere. Die Verfasser der Texte bewerten diese verschiedenen Themen positiv, neutral oder negativ (vgl. Abbildung 19). Neben der Bewertung der Themen kann die Textpassage eine weitere Dimension haben, nämlich die der Subjektivität. Der Grad der Subjektivität reicht von objektiv bis subjektiv. Abbildung 19 stellt den Zusammenhang zwischen der Textpassage, den Merkmalen und deren Ausprägung dar. Daraus folgt, dass jede Textpassage anhand zweier Charakteristika beschrieben werden kann: das *Was* und das *Wie*. Der Stimmungs- und Subjektivitätswert beschreibt das *Wie* und der Inhalt das *Was*. Die Textpassage kann Informationen über das Produkt, das Nutzererlebnis oder weitere Merkmale enthalten (vgl. Abbildung 19). Die Auswahl dieser Merkmale wurde in mehreren explorativen Vorstudien an realen Online-Reviews untersucht.

			Kundenstimme (Wahrnehmung & Meinung)				
			Stimmung			Subjektivität	
			Positiv	Neutral	Negativ	Subjektiv	Objektiv
Informationen aus dem Produktdatenmodell	Produkt	Funktion					
		Bauteil					
		Lebenszyklusphasen					
	Nutzererlebnis	Vor der Nutzung					
		Während der Nutzung					
		Nach der Nutzung					
	Weitere Informationen	Hersteller (Marke)					
		Weitere					

Abbildung 19: Charakterisierung von Textpassagen anhand von Formulierungen und Inhalten und deren Merkmalen und Ausprägungen

Eine ganzheitliche Untersuchung, welche (produktentwicklungsrelevanten) Inhalte in Reviews thematisiert werden, stellt eine noch nicht vollständig beantwortete und spannende Forschungsfrage in diesem Bereich dar. Die Annahme, dass die Inhalte der Textpassagen entweder Produkt, Nutzererlebnis oder andere Informationen beinhalten, ist für die Entwicklung einer Methode zur Beschriftung nutzergenerierter Textdaten ausreichend. Doch wie können diese Merkmale und die entsprechenden Ausprägungen Textdaten zugeordnet werden? In dem hier vorgestellten Framework wird ein hybrider Ansatz verwendet. Die Formulierungsmerkmale, wie Stimmung und Subjektivität werden automatisiert ermittelt, wobei die Inhaltsmerkmale manuell zugeordnet werden. Somit werden im Nachfolgenden nur die Inhalte, also die Merkmale **Produkt**, **Nutzererlebnis** und **weitere Informationen** betrachtet.

Dabei repräsentieren die Ausprägungen für das Merkmal **Produkt** die klassische konstruktionstechnische Perspektive. Die Produktarchitektur, die Kombination aus Produktstruktur (**Bauteile**) und Funktionsstruktur (Hierarchie der **Hauptfunktionen** und **Subfunktionen**), ist die Wissensgrundlage für die Produktmerkmale. Die Zuordnung, ob eine Textpassage einer dieser Ausprägungen zugeordnet werden kann, stellt für die Produktentwickler keine große Herausforderung dar und wird daher nicht näher ausgeführt. Eine weitere Informationsquelle aus der Produktentwicklung sind die Lebenszyklusphasen nach DIN EN ISO 12100: **Transport, Montage und Installation, in Betrieb nehmen, Verwendung, Demontage, außer Betrieb nehmen** und **Entsorgung** des Produktes. Auch die Zuordnung ist in den meisten Fällen uneindeutig und ohne großen Aufwand möglich. Das Merkmal *Nutzererlebnis*, oftmals wird auch der Begriff **User-Experience** verwendet, unterteilt das Erlebnis in drei Phasen: **Vor der Nutzung, während der Nutzung, nach der Nutzung** (DIN EN ISO 9241-210). Die Einordnung einer Textpassage in **vor**, **während** oder **nach der Nutzung** ist oftmals schwierig. Hier hilft es sich den Entstehungsprozess der Rezension zu verdeutlichen. Die Person macht bei der Nutzung des Produktes Erfahrungen, reflektiert diese und verfasst anschließend die Rezension. Zum Zeitpunkt des Verfassens sind die Handlungen abgeschlossen. Im Falle einer neuen Computertastatur und der Beschreibung über den Tastenanschlag der selbigen, wäre die Nutzung des Produktes und der Vorgang des Verfassens der Rezension zeitgleich. Daher sind die Rezensionen immer eine

nachträgliche Beschreibung von Erfahrungen und Sachverhalten. Für die hier vorliegende Arbeit wird aber die gewählte Zeitform oder die in der Rezension beschriebene Phase als Entscheidungskriterium definiert. Aus den explorativen Studien für die Voruntersuchung konnte gezeigt werden, dass sowohl Markennamen, Konkurrenzprodukte und weitere Informationen wie Preis oder Versand wichtige Themen in den Rezensionen waren. Aus Gründen der Übersichtlichkeit und um die Methodentwicklung nicht frühzeitig zu spezifisch zu entwickeln, werden die zwei Ausprägungen *Marke* und *weitere Informationen* betrachtet.



Abbildung 20: Prozessbeschreibung der entwickelten Methode zur Beschriftung von Textdaten

Der Initialprozess *Produktinformationen aufbereiten* besteht aus den Teilschritten *Produktstruktur erstellen*, *relevante Lebenszyklusphasen identifizieren* und *Marken- und Produktname benennen*. Diese Prozessschritte sind je nach Produkt und Detailgrad ein sehr aufwendiges Vorgehen, stellen aber an den Anwender der Methode (sofern die Informationen aus der Produktentwicklung verfügbar sind) keine besondere Anforderung. In einem exemplarischen Test dieser Methode stellten vor allem die Teilschritte 3 und 4 die Anwender der Methode vor Probleme. Daher werden diese Teilschritte im Folgenden näher erläutert. Die nachfolgenden Beschreibungen der Teilschritte sind aus den Feedbackgesprächen mit den ersten Testanwendern der Methode abgeleitet worden.

Phase 3 und 4: Wortgruppen bilden und anschließend Label zuordnen

Die Ausgangssituation für Phase 3 ist, dass der Rezensionstext und die möglichen Label vorliegen. Die Aufgabe besteht nun darin, den Rezensionstext in Wortgruppen aufzuteilen (Phase 3) und diesen Wortgruppen passende Label zuzuordnen (Phase 4). Den Zusammenhang der Phasen 3 und 4 stellt Abbildung 21 schematisch dar. Die Randbedingungen für die Auswahl der Wortgruppen und die Zuordnung der Label werden im Nachfolgenden näher erläutert. Die Wahl der Wortgruppe richtet sich in erster Linie nach ihrem Inhalt und nach der Prämisse, möglichst viele Wörter zu erfassen. Je länger die Wortgruppe ist, desto besser sind die Ergebnisse der automatisierten Stimmungsauswertung. Weiterhin sollte darauf geachtet werden, dass der Text vollständig in Wortgruppen eingeordnet wird (vgl. Abbildung 21 horizontal schraffiertes Viereck) und dass die Mehrfachzuordnung von Wörtern vermieden wird (vgl. Abbildung 21 vertikale Schraffur).

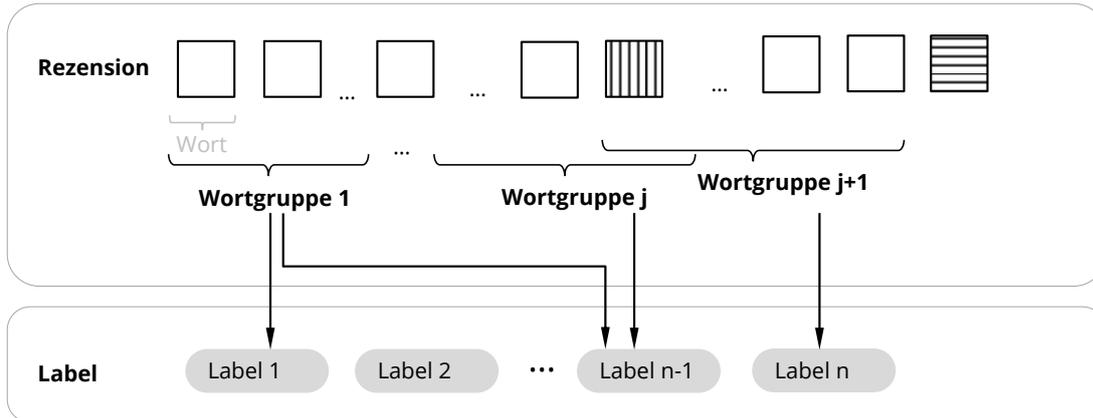


Abbildung 21: Schematischer Ablauf der Zuordnung der Wortgruppen der Rezensionen zu Labeln

Die Wortgruppen müssen originalgetreu übernommen werden und dürfen nicht durch Hinzufügen oder Entfernen von Wörtern verändert werden. Diese Einschränkung ist für eine Weiterentwicklung der Methode ein zentraler Punkt. Wenn es eine Möglichkeit gibt, vorhandene Rezensionen nach einem festen Schema „verbessert“ in Wortgruppen zu zerlegen, dann könnten anschließende Analysen mitunter noch bessere Ergebnisse liefern. Notwendige Voraussetzung für die erfolgreiche Umsetzung der Methode ist die Zuordnung eines Labels pro Wortgruppe. Weiterhin sollten alle Wörter verschlagwortet werden.

Aus dieser Vorgehensbeschreibung können folgende Kennwerte, Arbeitsanweisungen und angestrebte Zielwerte abgeleitet werden (vgl. Tabelle 6). Anhand der Kennwerte ist es möglich, den Labelvorgang zu quantifizieren. Die Arbeitsanweisungen helfen bei einer praxisnahen Implementierung der Methode.

Tabelle 6: Kennwerte, praktische Arbeitsanweisungen und theoretische Zielwerte für die Zuordnung der Label

Kennwerte	Arbeitsanweisungen	angestrebte Zielwerte
Anzahl Wortgruppen	Teile die Rezension in so wenig Wortgruppen wie möglich.	→ 1
Anzahl Wörter pro Wortgruppe	Jede Wortgruppe ist so lang wie möglich.	→ ∞
Mehrfachzuordnung von Wörtern	Jedes Wort ist nur einer Wortgruppe zugeordnet.	→ 0
Anzahl verwendeter Labels	Verwende so wenig Labels wie möglich.	→ 1
Label pro Wortgruppe	Jede Wortgruppe hat nur ein Label.	→ 1
Anzahl nicht gelabelter Wörter	Jedes Wort gehört zu einer Wortgruppe.	→ 0

Das Ergebnis nach Phase 5 besteht aus einer Rezension, die in mehrere Wortgruppen unterteilt ist, jeder Wortgruppe sind verschiedenen Label zugeordnet. Dieser Vorgang wird nun für alle Rezensionen des Produktes (vgl. Abbildung 10 - Produkt-Rezensions-Sammlung) durchgeführt. Nachdem die

gesamte Produkt-Rezensions-Sammlung in Wortgruppen eingeteilt und mit Label versehen worden ist, ist diese Phase abgeschlossen. Nun müssen diese Wortgruppen analysiert werden. Die Logik zur Auswertung dieser gelabelter Textdaten wird im nächsten Kapitel erarbeitet.

5.2 Entwicklung einer Logik zur Auswertung gelabelter Textdaten

Nachdem die einzelnen Textpassagen von Hand kategorisiert wurden, stellt sich nun die Frage, wie diese aufbereiteten Textdaten automatisiert und für die Produktentwicklung nutzbar ausgewertet werden können. Im Handlungsbedarf wurde argumentiert, dass vorhandene Sprachverarbeitungs-Algorithmen adaptiert werden. Um dieses Ziel zu erreichen, werden zunächst die Anforderungen an die Analyse abgeleitet, dann die Funktionalität der Sentiment-Analyse beleuchtet und schließlich eine leistungsfähige bestehende Softwarebibliothek zur Textverarbeitung ausgewählt.

5.2.1 Anforderung an die Textanalyse und Auswertlogik

Aus der Herleitung des Handlungsbedarfs in Kapitel 4.1 und dem Ergebnis der entwickelten Methode in Kapitel 5.1 ergeben sich für das Teilziel 2 folgende Fragen und Anforderungen.

Was ist der Aufbereitungsfokus der Ergebnisse? Die verwendete Logik muss fähig sein, die aufbereiteten Inhalte zu verarbeiten, in eine gesamtheitliche Struktur zu überführen und schließlich eine Aussage aus dem geschaffenen Zusammenhang abzuleiten, die für die Ingenieur:innen in der Produktentwicklung interpretierbar ist.

In Kapitel 2.4 wurden bestehende Ansätze zur Analyse von Nutzerinformationen für die Produktentwicklung beschrieben. Basierend auf den zwei Kategorien: Informationsgrundlage und Analyseziel werden die verschiedenen Anforderungen an den Aufbereitungsfokus der Ergebnisse beschrieben. Die Auswahl der Methode ist NLP bzw. Text Mining. Auf eine detailliertere Beschreibung der verschiedenen Nutzungsszenarien der Ingenieur:innen und Designer:innen wird hier verzichtet.

Die drei Dimensionen bei der datenbasierten Auswertung nach Zhan et al. (2018) wurden im Kapitel 2.4 ausführlich beschrieben und werden nun für das Modell angepasst. Als Informationsgrundlage dienen Kundenrezensionen. Diese Daten beinhalten Informationen über die Kundenbedürfnisse und Produkteigenschaften. Unter der Voraussetzung, dass eine Rezension immer dann verfasst wird, wenn vorher das Produkt gekauft wurde, wird der Punkt Kaufverhalten nur implizit betrachtet. Der Rezensionstext kann auch Informationen über vorherige Käufe (von Konkurrenzprodukten) der Person beinhalten. Weiterhin beinhaltet der Text Wünsche, Einschätzungen, Bewertungen, Verlangen der Kunden in Bezug auf das Produkt, insbesondere über dessen Funktion und Merkmale. Die nächste Dimension ist das Analyseziel, welches im Folgenden spezifiziert wird. Das Analyseziel für diese Arbeit ist die Produktverbesserung, indem die Abhängigkeiten zwischen Kundschaft und Produkt identifiziert werden und anschließend in Bezug auf die Kundenbedürfnisse bewertet (positiv - negativ und

objektiv – subjektiv) werden. Die letzte Dimension ist die (datenbasierte) Methode. Die Auswertung und Verarbeitung der Texte erfolgt durch Text Mining bzw. durch Natural Language Processing Verfahren.

Informationsgrundlagen	Analyseziele	Datenbasierte Methoden
Kundenbedürfnisse	Verbesserungen - Produkt	Text Mining / NLP
Kaufverhalten (implizit)	Abhängigkeiten identifizieren - Kunde & Produkt	
Produkteigenschaften - Produkthanforderungen		

Abbildung 22: Anforderungsprofil für die Textanalyse (Anwendungsfall auf die Übersicht von Abbildung 6 angewendet)

Damit diese Informationen entsprechend analysiert werden können, ist es notwendig, die manuell beschrifteten Daten anwendungsneutral und objektiv zu verarbeiten, die geäußerten Kundeneinstellungen automatisiert zu identifizieren und anschließend mit den technischen Merkmalen zu verknüpfen. Um eine Vergleichbarkeit der verschiedenen Label zu ermöglichen, muss die Auswertungslogik in der Lage sein, Stimmungswerte (positiv, negativ oder neutral) und Subjektivität aus den Textpassagen zu erkennen und in eine Kennzahl umzuwandeln sowie die Label zueinander in Beziehung zu setzen. Durch den Stimmungswert ist es möglich, die Wahrnehmung der Kundschaft gegenüber den Produktmerkmalen zu bestimmen. Darüber hinaus kann durch die Bestimmung der Subjektivität unterschieden werden, ob die Beschreibung reale Sachverhalte hinsichtlich des Produktes enthält oder nur eine subjektive Meinung ausdrückt.

Für den praktischen Nutzen dieser Arbeit, stellt insbesondere die Integration der Analyseziele durch die Ingenieur:innen und Designer:innen in den Produktentwicklungsprozess, einen besonders kritischen Schritt dar. Dieser wird im Folgenden näher beschrieben. Diese Beschreibung dient als Voraussetzung für die in diesem Kapitel zu entwickelnde Auswertungslogik.

5.2.2 Integration der Analyseziele in den Aufgabenbereich der Produktentwicklung

Der Produktentwicklungsprozess (PEP) und die Tätigkeiten in den verschiedenen Phasen wurden im Kapitel 2.1.4 aus den drei Perspektiven Ingenieurwesen, Management und Agile (Software)Entwicklung beschrieben. Die übergeordnete Aufgabe dieser Arbeit besteht darin, den einzelnen Aufgaben im Entwicklungsprozess einen Lösungsraum zu eröffnen und für diesen Lösungsraum eine *datenbasierte Entscheidungsgrundlage* zu ermöglichen (vgl. Abbildung 1). Dabei stellt der *Wissensspeicher* im *internen Unternehmensnetzwerk* diesen Lösungsraum für die *Produktentwicklung* bereit (vgl. Abbildung 18). Wie die

einzelnen Tätigkeiten in den verschiedenen Phasen der Produktentwicklung die Analyseziele nutzen, wird im Folgenden für die drei Perspektiven näher beschrieben.

Entsprechend der Zusammenfassung in Kapitel 2.7 werden folgende Analyseziele betrachtet:

- i) **Produktverbesserungen** und
- ii) **Abhängigkeiten identifizieren zwischen Kunden und Produkt.**

Für diese Arbeit wurde die **ingenieurbasierte Perspektive** des PEP in drei Phasen unterteilt: Klärung der Aufgabe, Konzeptualisierung und Entwurf und Ausarbeitung. Wie hängen die Analyseziele und die verschiedenen Aufgabenbereiche zusammen? Anhand mehrerer Leitfragen wurden die verschiedenen Tätigkeiten und die Analyseziele für die 3 Perspektiven gegenüber gestellt (vgl. Abbildung 23).

Perspektive	Produktentwicklungsphasen oder Tätigkeiten		
Ingenieurs-basierte	Klären der Aufgabe	Konzipieren	Entwerfen und Ausarbeiten
Management-basierte	Discovery	Incubation	Acceleration
Agile	Iterieren , Inkrementell, Empirisch	Iterieren , Inkrementell, Empirisch	Iterieren , Inkrementell, Empirisch

Analyseziele - Produktverbesserung - Abhängigkeiten zwischen Kunde und Produkt identifizieren	Welche Kundengruppen (KG) existieren?	Welche Produktfunktionen (PF) werden genannt?	Welche Produktmerkmale (PM) werden genannt?
	Wie nutzen diese KG das Produkt?	Wie werden die PF bewertet?	Wie werden die PM bewertet?
	Welche Probleme traten auf?	Sind diese Bewertungen subjektiv oder objektiv?	Sind diese Bewertungen subjektiv oder objektiv?
	--> Auf Basis dieser Analyse kundenspezifischeres Verständnis des Kunde-Produkt-Verhältnis	--> Auf Basis dieser Analyse verbesserte Produktkonzepte ableiten	--> Auf Basis dieser Analyse detaillierte Produktverbesserungen ableiten

Abbildung 23: Integration der Analyseziele in den Produktentwicklungsprozess

Die Gegenüberstellung zeigt, dass mithilfe der durch die Analyse beantworteten Fragen für jede einzelne Phase die Produktentwicklungsphasen besonders gestärkt werden. So kann für die erste Phase ein verbessertes kundenspezifisches Verständnis über das Verhältnis zwischen Kundschaft und Produkt abgeleitet werden. Darüber hinaus können in den nächsten beiden Phasen Verbesserungen entweder auf Konzeptebene oder auf Detailebene für die Produkte abgeleitet werden. Diese Informationen können dann von den Ingenieur:innen genutzt werden, um sie in technische Lösungen umzusetzen.

5.2.3 Stimmungsanalyse

Das notwendige Ergebnis aus der Textanalyse der gelabelten Wortgruppen ist die automatisierte Einordnung der Aussage in positiv, negativ oder neutral und ob die Aussage objektiv oder subjektiv (formuliert) ist. Das Ziel der Analyse besteht darin, den zugeordneten Label einen „Stimmungswert“ zuzuordnen, indem die Wortgruppe analysiert wird. Die Analyse bewertet die Wortgruppe auf zwei Arten: a) positiv, negativ oder neutral und b) objektiv oder subjektiv (vgl. Abbildung 24).

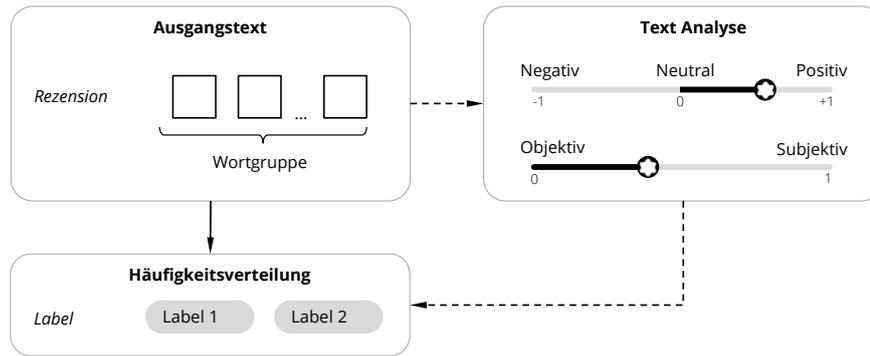


Abbildung 24: Schematischer Aufbau der Textanalyse der Wortgruppen

5.2.4 Verwendete Textverarbeitungsalgorithmen

Der verwendete Algorithmus stammt aus der Bibliothek *TextBlob* und basiert auf einem lexikalischen Ansatz und der Implementierung der Python Bibliothek NLTK. Durch diese Wahl wurden bei der Implementierung des Modells die Änderbarkeit und die Nachvollziehbarkeit der Softwareimplementierung wichtiger bewertet als ein performanteres Analyseergebnis. Eine Verbesserung ist z.B. durch den Einsatz der VADER-Bibliothek möglich (Bonta et al. 2019). Weiterhin wurde die interne Schnittstelle zur Google Translate API verwendet. Die umgesetzten NLP-Funktionen im Web-Tool listet Tabelle 7 auf.

Tabelle 7: Übersicht der NLP-Funktionen des webbasierten Tools

	umgesetzt		nachträglich erweiterbar
	automatisiert	händisch	
Part-of-Speech	X		
Sentiment Analysis	X		
Tokenization			X
Words Inflection and Lemmatization			X
Spelling Correction	X		
n-Grams			X
Classifier und Clustering		X	

Für nicht umgesetzte NLP-Funktionen wurde die nachträgliche Integration in das Tool untersucht und bewertet. Die Auswertung lässt den Schluss zu, dass alle aktuellen NLP-Funktionen (vgl. Kapitel 3.1) mithilfe des Tools abbildbar sind. Der implementierte Funktionsumfang ist fast vollständig. Den größten Kritikpunkt stellt die händische Umsetzung der Klassifizierung (Classifier und Cluster) dar. Die händische Umsetzung mithilfe des entwickelten Labelvorgangs mit der anwenderfreundlichen GUI bietet aber die Möglichkeit für Anwender (keine Software-Experten), das Tool ohne Anpassungen zu nutzen. Zusätzlich ermöglicht das Tool die Verwendung aktueller Cluster-Algorithmen. Die Anpassbarkeit der Parameter stellt hier die größte Schwachstelle in der Bedienbarkeit dar.

5.3 Grobkonzept für die Visualisierung

Bestehende Ansätze um nutzergenerierte Texte auszuwerten, scheitern an der praxisnahen Implementierung und Nutzbarmachung der zum Teil sehr aufwendigen und komplexen Logik. Damit die Modelle für die Produktentwicklung nutzbar sind, bedarf es einer ingenieurgerechten Darstellungsform der Analyseergebnisse. Nur eine richtige Aufbereitung der Daten kann die notwendigen Maßnahmen in der Produktentwicklung auslösen. Auf Basis von analysierten Daten Entscheidungen abzuleiten, stellt für die Produktentwicklung kein Novum dar. Die Nutzung und grafische Darstellung komplexer Algorithmen stellt Ingenieur:innen, aber oftmals vor große Herausforderungen. Für die direkte Erstellung eines Visualisierungskonzepts fehlen heute den Ingenieur:innen oftmals noch Methoden oder Vorgehensmodelle. Aus diesem Grund widmet sich das nachfolgende Kapitel der Erstellung eines nutzerspezifischen, anforderungsgerechten und pragmatischen Visualisierungskonzepts der Analyseergebnisse. Anhand des Fallbeispiels wird der Lösungsweg exemplarisch aufgezeigt.

5.3.1 Anforderung an das Visualisierungskonzept

Im Folgenden wird der Aufbereitungsfokus der Ergebnisse für die Produktentwicklung konkretisiert. Wie müssen die Daten aufbereitet werden, so dass die Analyseergebnisse direkt für die Produktentwicklung genutzt werden können? Und was bedeutet in dem Kontext „direkt genutzt“? Für die Entwicklung einer Lösung für das Teilziel 3 (Erstellung anforderungsgerechten Visualisierungskonzepts) sind folgende Anforderungen besonders wichtig: die ausgewerteten Informationen haben einen direkten Bezug zur Produktentwicklung, die einzelnen Textpassagen und die daraus resultierenden Ergebnisse sollten nachvollziehbar aufbereitet werden, eine ermittelte Rangfolge ist durch die Anordnung der Elemente direkt erkennbar, die Auswertungsergebnisse werden farblich unterstützt und für die Interpretation der Analyseergebnisse sind keine Vorkenntnisse von Text-Mining notwendig. Weiterhin ist der Zugriff auf die Visualisierung geräteunabhängig und im besten Fall über das Internet möglich. Auf eine Integration oder Anlehnung an etablierte Ingenieurwerkzeuge (Produktarchitektur, Morphologischer Kasten, Kano-Modell etc.) wird hier bewusst verzichtet, um ein möglichst flexibles Konzept zu entwickeln. Die Hauptperspektive ist darauf gerichtet, wie Produktmerkmale von Kunden beschrieben, wahrgenommen und bewertet werden. Im Folgenden wird ein grobes Visualisierungskonzept (Dashboard) entwickelt.

5.3.2 Visualisierungskonzept

In einem ersten Schritt werden die darzustellenden Elemente aufgelistet. Diese sind: Textdaten, die Label und die berechneten Kennzahlen. Dafür muss das Konzept die folgenden Elemente berücksichtigen.

Textdaten: Produkt-Rezensions-Sammlung (alle Kundenrezensionen), einzelne Kundenrezension, Wortgruppen und die zugeordneten Label.

Label (Produktdaten): Lebenszyklusphasen, Produktstruktur und Produktmarke.

Kennzahlen: Subjektivität und Polarität, Anzahl der verwendeten Label, zeitlicher Verlauf der Stimmungswerte und Häufigkeitsverteilung.

Um die Elemente entsprechend den Anforderungen an das Visualisierungskonzept darzustellen, werden drei Visualisierungsformen benötigt. Diese drei Formen sind: tabellarische Darstellung, Listenform und Graphen-Darstellung. Um die unterschiedlichen Häufigkeiten, d.h. die Anzahl, übersichtlich darzustellen, wird eine tabellarische Darstellungsform gewählt. Abbildung 25 zeigt die darzustellenden Elemente und deren Vorkommen sowie den Zusammenhang mit der Anzahl der Stimmungen und der Subjektivität. Das folgende Beispiel soll den Zusammenhang am Beispiel der Labels verdeutlichen. Daraus folgt, dass das Visualisierungskonzept der Tabellenform dann zeigt, welche Labels wie oft verwendet wurden. Es zeigt auch, wie oft das Label positiv, neutral und negativ bewertet wurde.

		Einzelwerte	Anzahl		
			Anzahl	Anzahl	
				Stimmung	Subjektivität
Text	Unstrukturiert	Kundenrezension	X	X	X
	Wortgruppe	X	X	X	
Strukturiert	Label	X	X	X	

Abbildung 25: Geeignete Kennzahlen (Einzelwerte – Anzahl) der Elemente für eine tabellarische Darstellung im Dashboard

Die Listenform ermöglicht insbesondere die unstrukturierten (Rohdaten) Kundenrezensionen und die gelabelten Wortgruppen darzustellen. Diese Auflistung der Ausgangstexte erhöht die Transparenz des Algorithmus und erleichtert das Verständnis der Kundenmeinungen; andererseits ist die Auflistung sehr unübersichtlich und lang. Daher wird sie am Ende des Dashboards platziert.

Die letzte Darstellungsform, die Graphen-Darstellung, eignet sich besonders für die Darstellung von zeitlichen Verläufen. Für die Auswertung sind insbesondere der zeitliche Verlauf der Anzahl der Bewertungen und der zeitliche Verlauf der Stimmungswerte der Labels für die Produktentwicklung von Interesse.

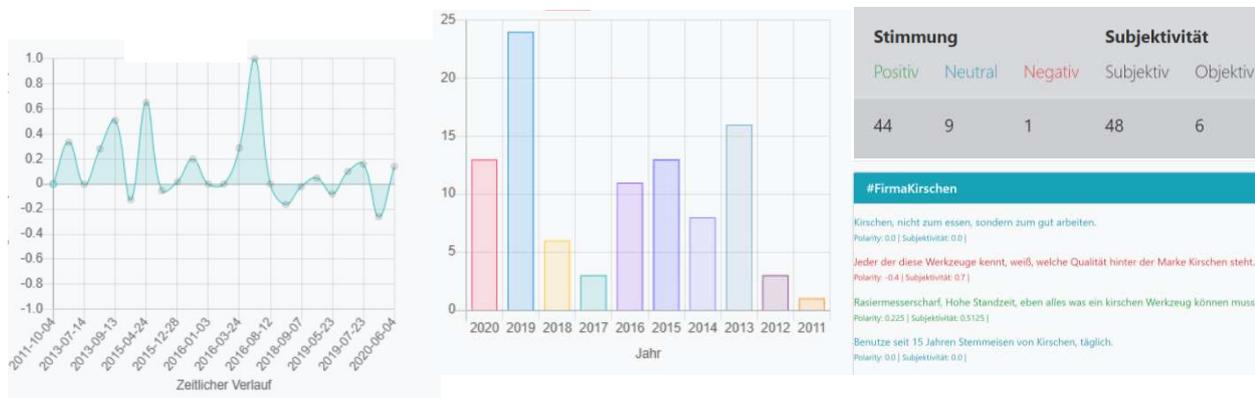


Abbildung 26: Dashboard-Dummy der Visualisierungskonzepte (links: zeitlicher Verlauf mit Graph, mitte: zeitlicher Verlauf Balkendiagramm, rechts oben: tabellarische Form, rechts unten: Listendarstellung)

6 Webbasierte Umsetzung des Modells

Nach Kenntnis des Autors existiert keine ganzheitliche webbasierte Implementierung eines Text-Mining-Tools für die Produktentwicklung. Dies bedeutet, dass Ingenieur:innen bei der Entscheidungsfindung für Produktverbesserungen nicht mit aktuellen Softwarelösungen unterstützt werden können. Dies kann dazu führen, dass notwendige Produktverbesserungen übersehen und Kundenbedürfnisse nicht berücksichtigt werden. Um diesen möglichen Fehlern mit einem systematisch und methodischen Lösungsansatz zu begegnen, wird in dieser Arbeit ein webbasiertes Modell vorgeschlagen. Ziel ist es dabei, die theoretisch ausgearbeiteten Teilziele 1-3 durch die Entwicklung einer Software praktisch umzusetzen. Die entwickelte Software berücksichtigt sowohl die Trends und Anforderungen aus den ingenieurwissenschaftlich begründeten Ansätzen als auch die zugrundeliegenden *modernen* Anforderungen an Software (z.B. webbasiert, skalierbar, modular, nutzerfreundlich etc.).

Aus diesem Grund ist es notwendig, die Anforderungen an die Implementierung für eine praxisnahe Anwendung zu formulieren. Die Umsetzung des Modells löst die in Kapitel 4.1 (Handlungsbedarf) abgeleiteten Probleme, indem die theoretisch und methodisch entwickelten Lösungen mittels einer Software instanziiert wird. Das dabei gewählte Vorgehen wird im Folgenden näher dargestellt: (6.1) beschreibt die Anforderungen für die Implementierung, (6.2) das verwendete Framework und die Software-Entwicklungsumgebung, (6.3) den groben Aufbau des Gesamtmodells und (6.4) die Umsetzung der einzelnen Lösungsvorschläge im Detail. Dabei werden vor allem die (6.4.1) selbstentwickelten graphischen Benutzerschnittstellen (graphical user interface, GUI), (6.4.2) die Bedienung durch den Nutzer (Funktionsumfang) und (6.5) die zugrundeliegenden logischen Datenbankstrukturen näher beschrieben. Abschließend werden (6.6) die Ergebnisse in Bezug sowohl zu den Teilzielen 1-3, als auch den formulierten Anforderungen vorgestellt. In Kapitel 6.7 wird ein Zwischenfazit gezogen.

6.1 Anforderungen an die Implementierung

Für den Produktentwickler von morgen ist der Einsatz von softwaregetriebenen Lösungen, d.h. die Implementierung und das Testen, eine immer wichtigere Aufgabe, um den Herausforderungen der modernen Produktentwicklung effizienter und effektiver zu begegnen. Aus dieser Motivation heraus, werden Anforderungen an die Implementierung formuliert. Folgende Annahmen und Hintergründe werden für die Herleitung der Anforderungen berücksichtigt: Erstens müssen die Anforderungen den Aspekten aus Kapitel 4.1 (Handlungsbedarf) genügen; zweitens muss die einfache Anwendung der Implementierung für Nicht-Softwareentwickler:innen, insbesondere für softwareaffine Ingenieur:innen möglich sein. Ziel des Kapitels ist es nicht, eine vollständige Dokumentation bestehend aus Lasten- und Pflichtenheft nach der IEEE Richtlinie für Software Anforderungen (1983) zu erarbeiten, vielmehr werden die wichtigsten Bestandteile (Produktfunktionen, Benutzermerkmale, Annahmen

und Abhängigkeiten, funktionale und nicht-funktionale Anforderungen, externe Schnittstellen) selektiv vorgestellt.

Aus dem formulierten Handlungsbedarf, der aus dem Vergleich der aktuell existierenden Lösungen identifiziert worden ist, wurden vier Anforderungen für die Nutzung und Implementierung definiert: (1) Modell muss frei verfügbar sein, (2) modularer Aufbau, (3) gute Bedienbarkeit und (4) webbasierte Nutzung. Weiterhin wurde die Funktionalität der sprachbasierten Auswertung, insbesondere der Stimmungsmodellierung als Anforderung formuliert. Für die Implementierung gilt die Annahme, dass der Nutzer (auch „Benutzermerkmale“ in der IEEE Dokumentation genannt) ein technisches Verständnis aufweist und Erfahrung in der Bedienung verschiedener Ingenieurs-Software hat und eine überdurchschnittliche Sachkenntnis mit dem Umgang auf Internetseiten oder browserbasierten Applikationen hat. Der Nutzer des Modells (also der Implementierung) ist nicht der Entwickler des Modells. Der Entwickler des Modells hat ebenfalls einen technischen Hintergrund, muss darüber hinaus aber eine tiefgehende Expertise in der Anwendung und dem Aufbau von Software-Code haben.

Eine weitere Anforderung leitet sich aus der Beobachtung bestehender Lösungsansätze für die Textverarbeitung ab. Es fällt auf, dass hauptsächlich Python als Programmiersprache verwendet wird. Diesem Trend folgend, ist die Programmiersprache für die Implementierung ebenfalls Python. Diese Entscheidung ist technisch nicht unbedingt nötig, da viele Programmiersprachen sehr gute Schnittstellen zu Python (sog. Bridges) integriert haben. Vielmehr reduziert eine native Implementierung in Python die Komplexität, was sowohl zu einer besseren Nachvollziehbarkeit für den Leser führt, als auch die Weiterentwicklung der Implementierung vereinfacht.

Die Implementierung muss deutsche Rezensionen verarbeiten. Das begründet sich sowohl aus der definierten Zielgruppe des zukünftigen Nutzers der Implementierung, als auch daraus, dass der deutsche Markt als Maßstab für neuartige Produkte verwendet wird.

6.2 Software-Entwicklungsumgebung und verwendetes Framework

Das verwendete Framework heißt *Django*. *Django* ist in Python geschrieben. Es ist ein quelloffenes Webframework – also eine Software zur Entwicklung von dynamischen Webanwendungen. Webframeworks haben das Ziel, schnell lauffähige Webanwendungen zu erstellen. Dazu werden häufig verwendete Funktionen von Webanwendungen (Authentifizierung, Datenbankzugriffe, Admin- und Userbereiche, Formulare etc.) direkt integriert – was den Entwicklungsaufwand stark reduziert. Neben *Django* existieren noch viele weitere Frameworks. Die bekanntesten und am meisten verwendeten sind: *React* (Java-Script), *Angular* (TypeScript bzw. JavaScript Nachfolger), *Laravel* (PHP), *Ruby on Rails* (Ruby). Die folgenden Websites verwenden eines dieser Frameworks: Facebooks (teilweise React), Instagram (React und Django), YahooMail (React), MacArthur Foundation (Django), Mozilla (Django), Pinterest (Django), Upwork (Angular), Paypal (Angular), Airbnb (Ruby on Rails), Shopify (Ruby on Rails), Kickstarter (Ruby on Rails).

Ein weiteres sehr bekanntes auf Python-basiertes Framework ist *Flask* (Netflix und Reddit basieren auf *Flask*). In einer Voruntersuchung testete der Autor beide Frameworks und entschied sich, wegen besserer Usability Erfahrung in der Testphase und besserer Dokumentation der Anwendungen, für *Django*.

Die genutzte Softwareentwicklungsumgebung, auch integrierte Entwicklungsumgebung (Integrated Development Environment, IDE) bezeichnet, ist PyCharm (Community Edition Version 2019.1.4) der Firma JetBrains (vgl. Abbildung 27).

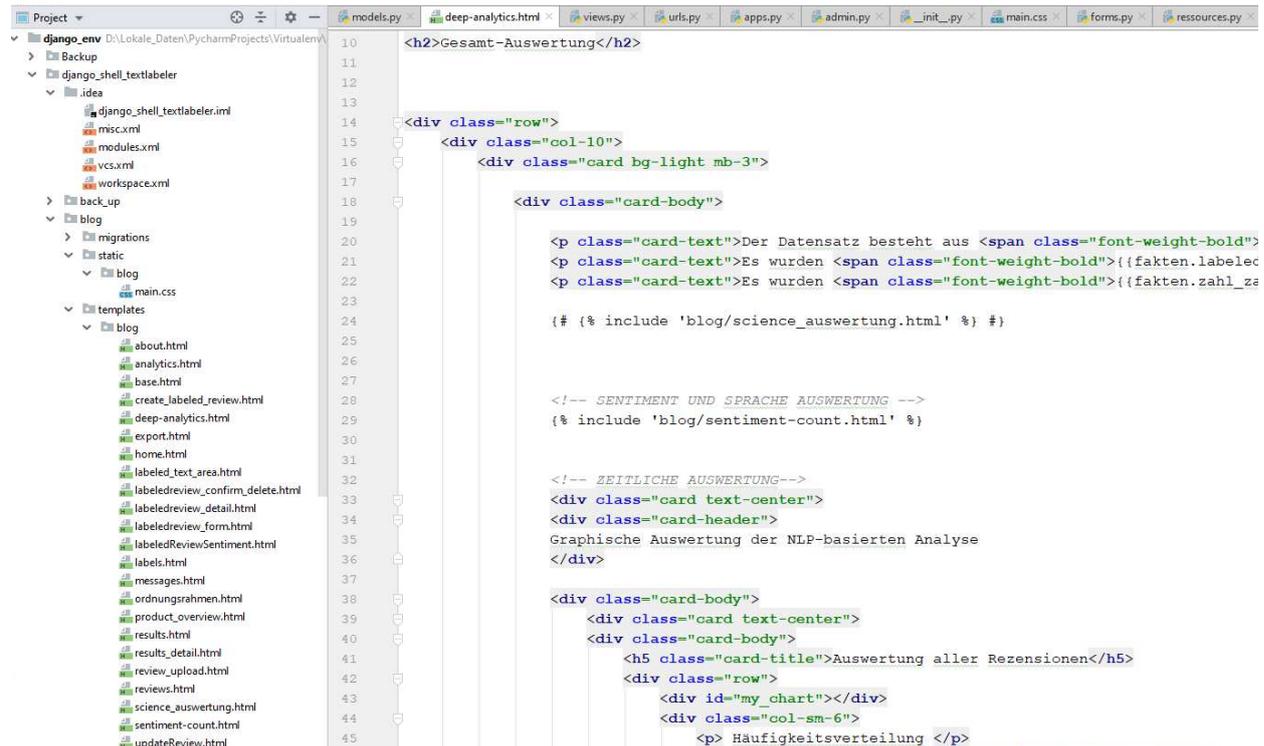


Abbildung 27: Verwendete Entwicklungsumgebung PyCharm der Firma JetBrains

Dieses Programm basiert ebenfalls auf Python und bietet eine sehr umfassende native Implementierung für die Entwicklung von Webframeworks. Zum Testen und Entwickeln des Softwarecodes wurde PyCharm auf einem Laptop (Lenovo T480S – mit Intel™ Core™ i7-8550U CPU @ 1.80GHz 1.99 GHz und 16,0 GB RAM) mit einem 64-bit Windows 10 Pro Betriebssystem installiert. Weiterhin wurde die interne Windows-PowerShell (vgl. Abbildung 28) zur Erstellung eines virtuellen Servers genutzt. Die grafische Ausgabe erfolgte mit dem Webbrowser Google Chrome.

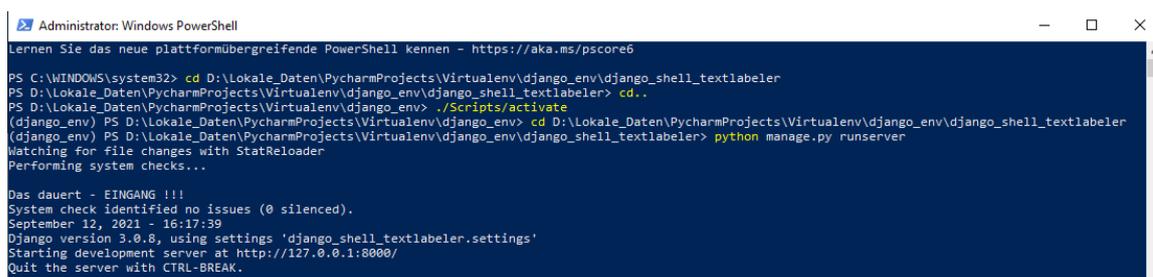


Abbildung 28: Windows PowerShell - Befehle zum Starten des lokalen Servers für die Webapplikation

Um zu prüfen, ob die Implementierung auch außerhalb des lokalen Servers funktioniert, wurde bei *Linode* (ein Cloud-Hosting-Anbieter) ein Server gemietet und mit dem *Linux*-Betriebssystem *Ubuntu* (Version 18.04) installiert und dann das entwickelte Gesamtmodell hochgeladen. Diese Bereitstellungsumgebung (Deployment Environment) hat sich als tragfähig herausgestellt. Es konnte gezeigt werden, dass die Funktionen erfüllt wurden und dass mehrere Personen darauf zugreifen konnten.

6.3 Webbasierte Applikation – Aufbau, GUI und Funktionsweise

Entsprechend der vorher beschriebenen Anforderungen und den einzelnen Funktionsumfängen, die sich aus den Teilzielen entwickelt haben, wird in diesem Kapitel der Aufbau der webbasierten Applikation beschrieben. Dabei werden die verschiedenen GUI's, deren Aufgaben und die Zusammenhänge untereinander beschrieben. Es wird hier bewusst auf den zugrundeliegenden Software-Code und die Django-spezifische Logik verzichtet. Diese Informationen werden separat im darauffolgenden Kapitel 6.4 beschrieben. Diese Aufteilung vereinfacht die Nachvollziehbarkeit und Verständlichkeit für den Funktionsaufbau der webbasierten Applikationen für Ingenieur:innen, Manager:innen, Designer:innen und andere Interessierte ohne tiefere Softwarekenntnisse. Das anschließende Kapitel 6.4 beschreibt die Applikation für softwareaffine Ingenieur:innen. Es ermöglicht ihnen die Applikation in der Architektur und dem Aufbau zu verstehen und somit Änderungen und Anpassungen vorzunehmen. Für den Softwareentwickler ist die nachfolgende Beschreibung, der (digitale) Anhang und das Git-Hub Repository ausreichend, um die Applikation vollständig zu verstehen.

Die webbasierte Applikation ist wie eine klassische Website aufgebaut und wird in einem Browser (im Falle dieser Dokumentation – *Google Chrome*) dargestellt. Abbildung 29 stellt die verschiedenen Bestandteile (graphischen Elemente) und deren Verknüpfung dar und wird im Folgenden näher erläutert. Begonnen wird links oben mit der GUI. Hier wird die zu untersuchende Rezension visualisiert. Die Darstellung ist die Hauptseite des Tools beim Labeln. Auf dieser graphischen Oberfläche werden folgende Bestandteile von oben nach unten dargestellt: der Rezensionstext, die zur Verfügung stehenden **Label** (als Toggle-Button integrierte Schaltflächen), ein **leeres Textfeld** (diese Textfeld nimmt später die zu labelnden Textabschnitte auf), die schon **gelabelten Textpassagen** und zuletzt eine Navigationsleiste (Paginator) um zur nächsten Rezension zu gelangen. Weiterhin befindet sich auf der Seite ganz oben ein Menü. In dem oberen Menü, kann auf folgende Seiten (Oberflächen) **Auswertung**, **Backend** und **Profil- und Account-Management** gewechselt werden.

Nachdem die einzelnen Elemente und deren Beziehungen nun grob vorgestellt worden sind, werden nun die Funktionsumfänge und deren Bedienung detailliert beschrieben.

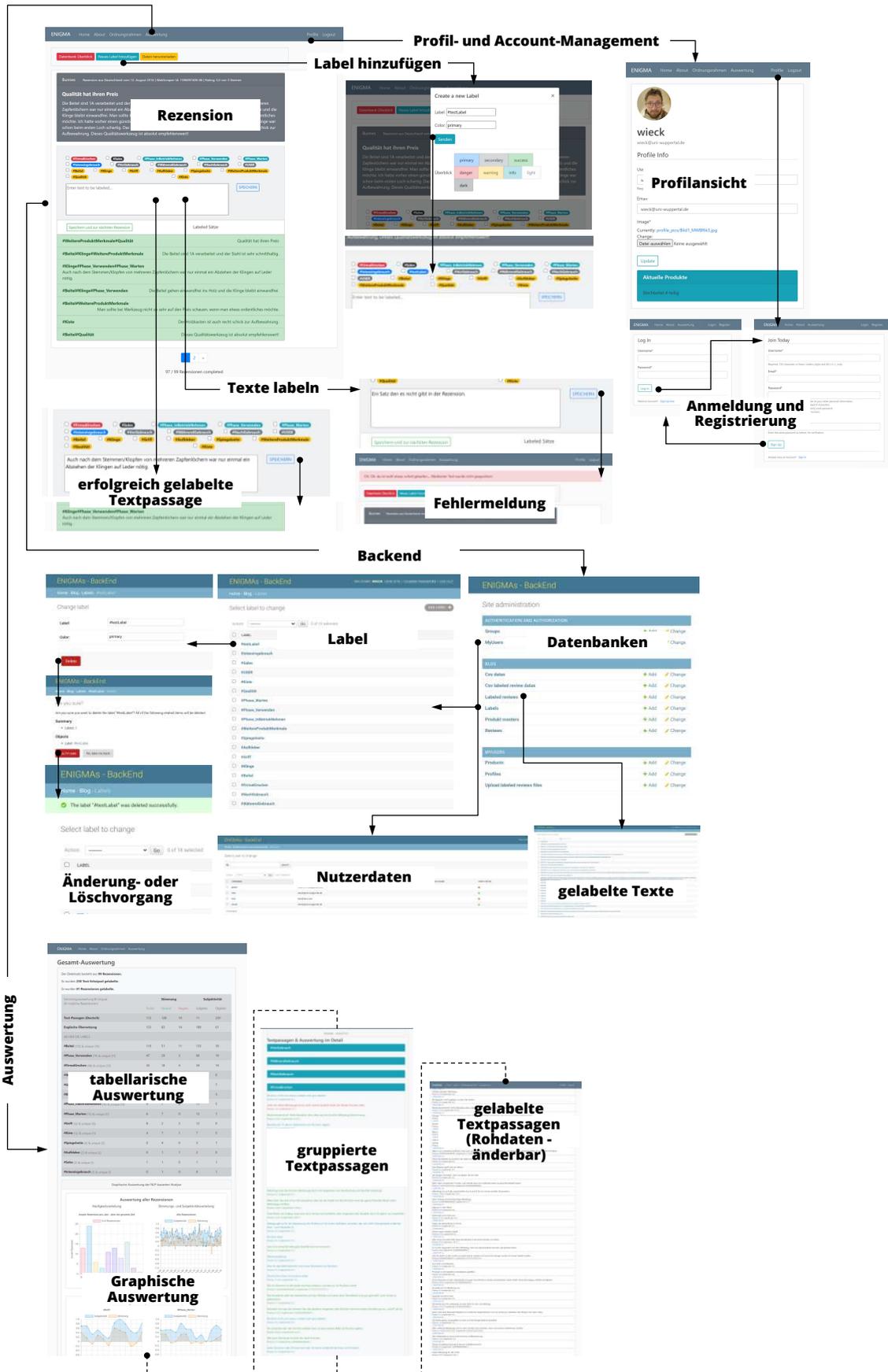


Abbildung 29: Bestandteile und Verknüpfungen der webbasierten Applikation des Modells

6.3.1 Textdaten hochladen

In einem vorausgegangenen Schritt wurden die nutzergenerierten Textdaten extrahiert und in einer Textdatei (vorzugsweise csv-Format) gespeichert. Üblicherweise geschieht das über einen Webcrawler. Ein Webcrawler durchsucht Internetseiten und extrahiert deren Daten. Um ein weniger anfälliges und wartungsärmeres Tool zu entwickeln, wurde bei der Implementierung das Webcrawlen absichtlich nicht mitimplementiert. Bevor die nutzergenerierten Textdaten gelabelt werden, müssen diese in das Tool (oder auch Website) geladen werden. Über eine Abfrage kann eine Datei vom lokalen Rechner hochgeladen werden. Das Tool überprüft ob das Dateiformat passend ist und ob die Daten entsprechend der Vorgabe in der Datenbankstruktur vorliegen. Nach erfolgreichem Hochladen gibt es eine Rückmeldung, dass die Daten nun abgelegt worden sind. Dabei wird jedem Datenpunkt, also einer Rezension, eine eindeutige Ident-Nummer zugeordnet. Nachdem der Datensatz hochgeladen worden ist, kann dieser nun weiter prozessiert werden. Neben dem Import ist natürlich auch der Export des gesamten Datensatzes (nach dem Labeln und weiterer Analyseschritte) möglich.

6.3.2 Texte labeln

Die hochgeladenen Rezensionen werden nun einzeln dargestellt. Die Darstellung einer Rezension pro Seite hat sich bewährt. Abbildung 30 zeigt den Aufbau der GUI. Um nun Textpassagen verschiedenen Label zuzuordnen, wird die gewünschte Textpassage (mit der Maus) markiert und dann per Drag-and-Drop (Ziehen und Ablegen) in das leere Textfeld darunter verschoben. Anschließend werden die passenden Label oberhalb des Textfeldes angeklickt. Sind alle gewünschten Label ausgewählt, wird der Speichern-Button gedrückt. Nach erfolgreichem Speichern wird kurz eine Bestätigung eingeblendet. Die gelabelte Textpassage nebst Label wird unterhalb der Rezensionen dargestellt und in einem neuen Datensatz abgespeichert. Es ist zu erwähnen, dass es an dieser Stelle nicht möglich ist, den Text zu ändern. Ebenfalls ist es nicht möglich Textpassagen zu kürzen. Diese Kontrollfunktion verhindert, dass beim Labeln Fehler (zusätzliche Zeichen gelöscht oder hinzugefügt) versehentlich auftreten oder dass der Datensatz bewusst manipuliert wird. Sind alle Textpassagen der Rezension gelabelt worden, so wird der Button mit der Beschriftung „Speichern und zur nächsten Rezension“ gedrückt. Wenn dieser Button gedrückt wird, steht die Rezension nicht mehr zum Labeln zur Verfügung und wird nicht mehr angezeigt. Es ist natürlich möglich über das Backend-Panel (Button: Datenbank Überblick) diesen Zustand von *complete* wieder auf *wip* oder *neu* zu setzen, um weitere Informationen hinzuzufügen. Die Änderung und Korrekturmöglichkeiten der gelabelten Textpassagen ist entweder über das Backend oder durch eine separate Funktionalität für den Nutzer möglich (vgl. Anhang A.3 - Abbildung 55).

The screenshot shows the ENIGMA web tool interface. At the top, there is a navigation bar with 'ENIGMA', 'Home', 'About', 'Ordnungsrahmen', 'Auswertung', 'Profile', and 'Logout'. Below this, there are three buttons: 'Datenbank Überblick', 'Neues Label hinzufügen', and 'Daten herunterladen'. The main content area is titled 'Bunnies Rezension aus Deutschland vom 12. August 2016 | WebScraper-Id: 1596097406-86 | Rating: 5,0 von 5 Sternen'. The review text is: 'Qualität hat ihren Preis. Die Beitel sind 1A verarbeitet und der Stahl ist sehr schnitthaltig. Auch nach dem Stemmen/Klopfen von mehreren Zapfenlöchern war nur einmal ein Abziehen der Klingen auf Leder nötig. Die Beitel gehen einwandfrei ins Holz und die Klinge bleibt einwandfrei. Man sollte bei Werkzeug nicht so sehr auf den Preis schauen, wenn man etwas ordentliches möchte. Ich hatte vorher einen günstigen Beitel aus dem Baumarkt probiert (Billigstahl aus China), aber die Klinge war schon beim ersten Loch schartig. Das Geld hätte ich auch wegwerfen können. Der Holzkasten ist auch recht schick zur Aufbewahrung. Dieses Qualitätswerkzeug ist absolut empfehlenswert!'. Below the text is a section for selecting labels with checkboxes and colored buttons for various terms like '#FirmaKrischen', '#Sales', '#Phase InBetriebNehmen', '#Phase Verwenden', '#Phase Warten', '#Wartungsgebrauch', '#Vorgebrauch', '#Währendgebrauch', '#Nachgebrauch', '#ZUSA', '#Beitel', '#Klinge', '#Griff', '#Aufkleber', '#Spiegelseite', '#WeitereProduktMerkmale', and '#Kiste'. There is an input field 'Enter text to be labeled...' and a 'SPEICHERN' button. Below this is a table of 'Labeled Sätze' with columns for the label and the corresponding text. The table contains seven rows of labeled sentences. At the bottom, there is a pagination bar with '1', '2', and a right arrow, and a status '97 / 99 Rezensionen completed'.

Abbildung 30: Graphische Oberfläche des Web-Tools zum Labeln der Rezensionen (Home-View)

6.3.3 Label bearbeiten

Wenn das Tool zum ersten Mal genutzt wird, dann gibt es die Möglichkeit, einen Standard-Label Satz mit einem Mausklick zu erstellen. Wenn während des Beschriftungsvorgangs weitere Label benötigt werden, ist es jederzeit möglich ein neues Label zu erstellen. Hierfür wird auf die Schaltfläche mit der Bezeichnung *Neues Label hinzufügen* geklickt. Ein neues Fenster öffnet sich mit der Abfrage des Label-Namens und der gewünschten Farbe (vgl. Abbildung 31). Dieser Vorgang ist auch über das Backend-Panel möglich. Die Label-Erstellung ist ein wiederkehrender Vorgang und sollte somit für eine bessere Bedienbarkeit des Tools direkt über einen Button abrufbar sein.

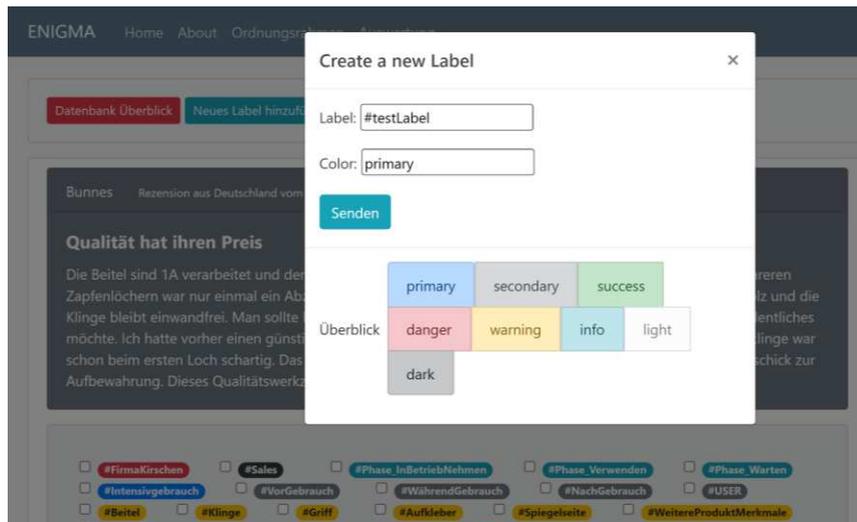


Abbildung 31: GUI zur Bestimmen des neuen Labels und einer Farbe

6.3.4 Datenauswertung

Die Auswertung der Textdaten ist auf einer eigenen Seite des Tools implementiert. Die Seite besteht aus vier Bereichen: **Meta-Analyse**, **Label-Auswertung**, **Diagramm-Bereich**, **Text-Detail-Bereich**. Abbildung 32 zeigt die Anordnung und den Aufbau der verschiedenen Bereiche.

Die **Meta-Analyse** berechnet wie viele Rezensionen insgesamt zur Verfügung stehen, wie viele Textgruppen davon extrahiert (gelabelt) wurden und wie viele Rezensionen ignoriert wurden. Letzteres wird über die Differenz der Zahlen für *Der Datensatz besteht aus YY Rezensionen* und *Es wurden XX Rezensionen gelabelt* beschrieben. Sie verschafft einen Überblick über den Gesamtvorgang und hilft die Qualität des Datensatzes zu beurteilen.

Die **Label-Auswertung** schließt daran an. Die Kriterien für die Auswertung der Label sind die Stimmungen (positiv, neutral, negativ) und die Subjektivität (subjektiv oder objektiv). Die jeweils berechneten Werte für Stimmung und Subjektivität werden in der Tabelle nicht aufgeführt, vielmehr wird auf Basis der Berechnung und festgelegter Schwellwerte das Label eingeordnet. Die Grenzwerte haben einen großen Einfluss auf die Auswertung. Auf diese Einflüsse wird in dem Fallbeispiel näher eingegangen. Neben diesen Kriterien wird zusätzlich analysiert, in wie vielen Rezensionen das Label verwendet wurde (*unique*). Dabei wird unterschieden, ob das Label in einer Rezension mehrfach verwendet wurde. Durch diese Differenzierung ist es möglich, den Trend der Label besser zu erkennen und einzelne Reviews in der Auswertung nicht zu überbewerten. Label die nicht verwendet worden sind, werden nicht aufgeführt.

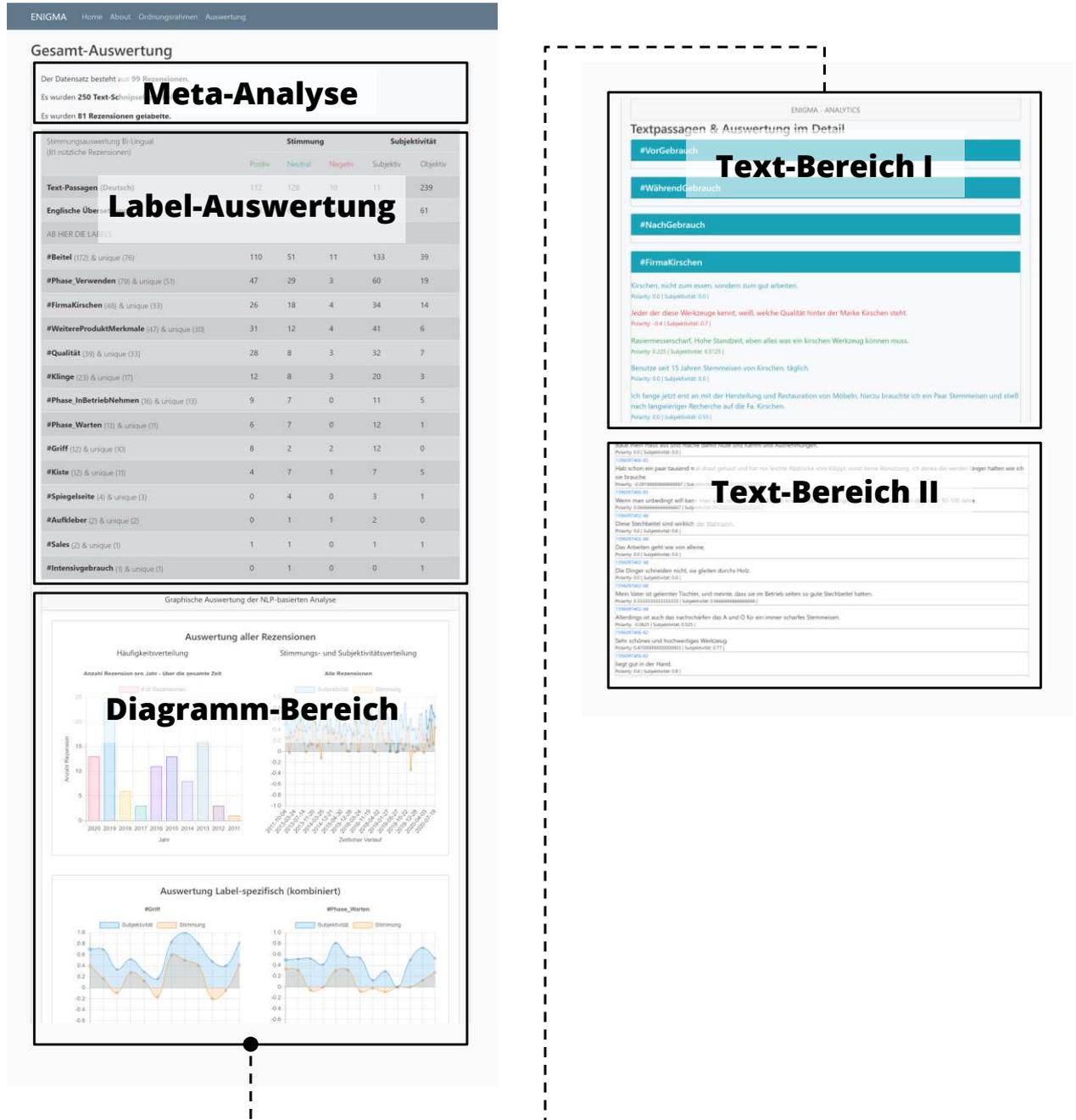


Abbildung 32: Überblick der verschiedenen Bereiche der Datenauswertung

Der **Diagramm-Bereich** der Auswertung besitzt verschiedene Diagramme: (1) Anzahl Reviews über Zeit, (2) Stimmungsverlauf des Labels über der Zeit, (3) Stimmungsverlauf der Rezension. In der hier implementierten Basisversion wird im Diagramm (1) Anzahl der Reviews über der Zeit pro Jahr dargestellt. Diese Darstellung ist vor allem für die erste Analyse relevant. Hierbei werden Trends erkennbar und auch frühere Änderungen am Produkt können früher identifiziert werden. Die Korrelation zwischen Bewertungen und Verkaufszahlen ist ein zusätzliches Kriterium. So ermöglicht die Darstellung die Schätzung der potenziellen Verkaufszahlen. In der E-Commerce-Branche sind Verhältnisse von 1:10 typisch. Das bedeutet, dass jeder 10. kaufende Person im Durchschnitt eine Rezension verfasst. Diesen Wert gilt es für verschiedene Produktbranchen anzupassen, um ein

besseres Schätzergebnis zu erzielen. Das nächste Diagramm (2) stellt den Stimmungsverlauf des Labels über der Zeit dar. Auf diese Weise lassen sich Trends, Spitzen oder starke Schwankungen in den Labels, also der Kundenwahrnehmung eines bestimmten Merkmals, sofort erkennen. Es ist möglich diesen Diagramm-Typ für alle Label darzustellen. Das letzte Diagramm (3) stellt den Stimmungsverlauf der gesamten Rezensionen dar. Mithilfe dieser Auswertung ist es möglich, die einzelnen Label in Bezug auf die Gesamtstimmung der Rezensionen zu vergleichen. In den ersten Testphasen wurden versucht die beiden Graphen aus (3) und (2) in einem Diagramm darzustellen, um die Auswertung für den Nutzer eindeutiger und besser zu gestalten. Das verwendete Softwarepaket *canvasJS* bietet die Möglichkeit die Subjektivitäts- und Stimmungswerte zusammen in einem Graphen abzubilden (vgl. Abbildung 33).

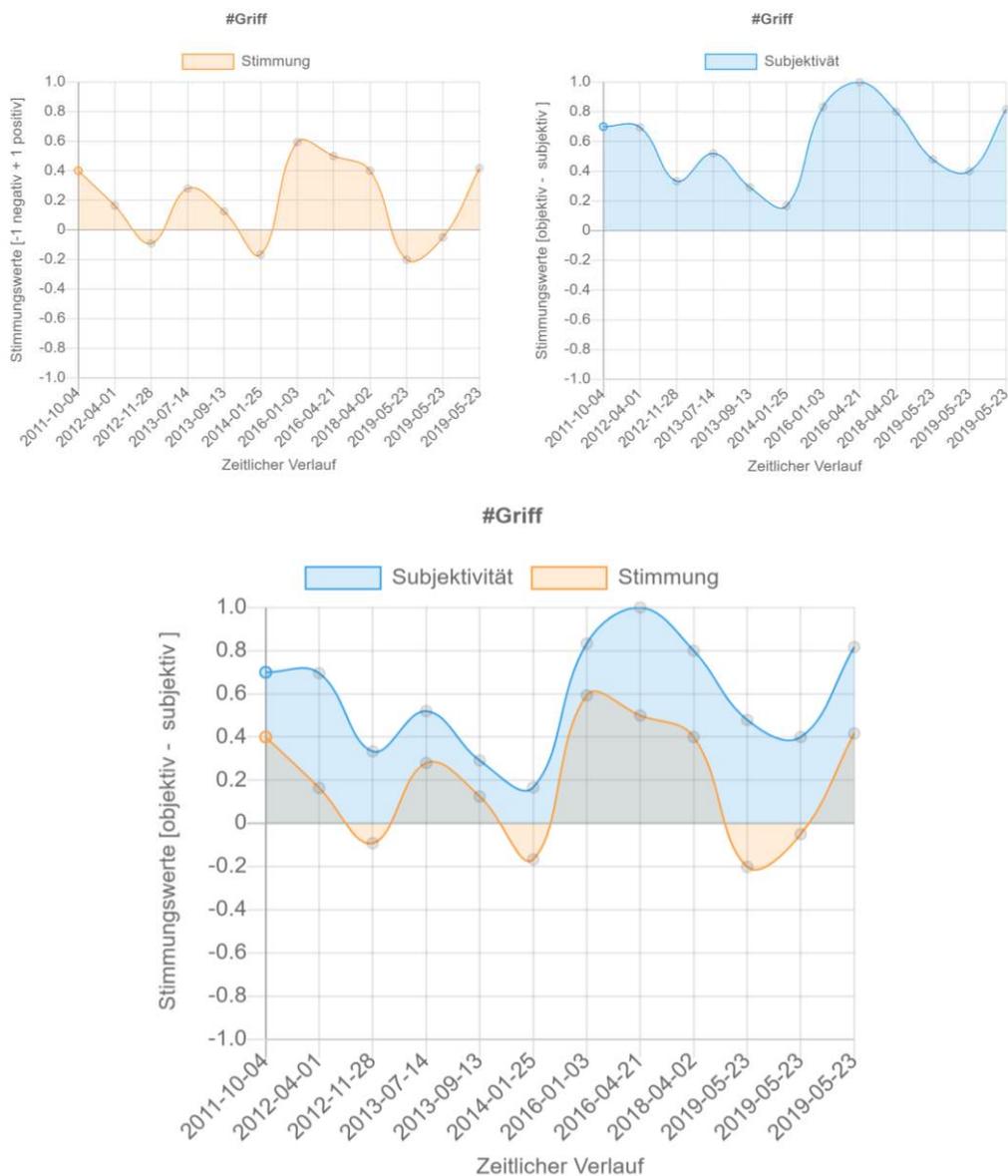


Abbildung 33: Aufbau und Zusammensetzung der graphischen Auswertung

Der **Textdetail-Bereich** unterteilt sich in zwei Bereiche. Der erste Bereich führt im Detail die Textpassagen für jedes Label und die ermittelten Werte für die Polarität und die Subjektivität auf. Die Textpassagen sind entsprechend der Label sortiert und der Eingruppierung nach positiv (grün), neutral (blau) und negativ (rot) eingefärbt. In den ersten Testphasen hat sich gezeigt, dass die Transparenz der Ergebnisse für den Nutzer ein wichtiges Kriterium ist. Außerdem war es wichtig, welche Informationen in der Rezension behandelt wurden. Auf Basis dieser zwei Erkenntnisse ist diese umfangreiche Liste auf die Seite (unterhalb der Graphen) der Auswertung angeordnet worden. Der zweite Bereich listet alle gelabelten Textpassagen auf. Anhand eines Mausklicks ist es möglich diese Rezensionen zu bearbeiten. Sollten Fehler oder Ungenauigkeiten während der Analyse sichtbar werden, kann die spezifische Rezension bearbeitet werden.

6.3.5 Datenmanagement und Datenbankstruktur

Das entwickelte Datenbankmanagementsystem orientiert sich sehr stark an den Vorgaben und Richtlinien für die Implementierung von Django. Für eine detaillierte Beschreibung empfiehlt es sich, die aktuelle Django-Dokumentation zur jeweiligen Datenbank und deren Implementierung als zusätzliche Informationsquelle heranzuziehen. Der Vollständigkeit halber werden alle Datenbanken beschrieben, wobei die Datenbank, die für die Verwaltung der Textdaten zuständig ist, die neuartigste ist.

Die Daten sind in drei verschiedene Datenbanken aufgeteilt: (1) **Autorisierung und Authentifizierung**, (2) **Blog** und (3) **Profilinformationen der User** (vgl. Abbildung 34). Wie diese Datenbanken strukturiert sind, wird im Nachfolgenden beschrieben.

The screenshot shows the Django Admin interface for 'ENIGMAs - BackEnd'. At the top, there is a navigation bar with 'WELCOME, WIECK' and links for 'VIEW SITE / CHANGE PASSWORD / LOG OUT'. Below this, the 'Site administration' section is visible. It is divided into three main categories, each with a blue header bar:

- AUTHENTICATION AND AUTHORIZATION**: Contains 'Groups' and 'MyUsers', each with '+ Add' and 'Change' (pencil icon) buttons.
- BLOG**: Contains 'Csv datas', 'Csv labeled review datas', 'Labeled reviews', 'Labels', 'Produkt masters', and 'Reviews', each with '+ Add' and 'Change' buttons.
- MYUSERS**: Contains 'Products', 'Profiles', and 'Upload labeled reviews files', each with '+ Add' and 'Change' buttons.

Abbildung 34: Datenbankstruktur im Backend des Tools

Die Datenbank oder das Model (1) **Autorisierung und Authentifizierung** verwaltet die Login-Daten und die Zugriffsrechte der Nutzer. Für die Nutzung des Tools ist es notwendig einen Account anzulegen. Diese nutzerbezogenen Informationen werden dann beim Login -Vorgang in dieser Datenbank verglichen. Das Tool ermöglicht den Nutzern verschiedene Rechte und Gruppen zuzuweisen. Folgende zwei Nutzergruppen wurden implementiert: Admin (bei Django als *Superuser* bezeichnet) und normaler User. Der Superuser hat die Möglichkeit in einem separaten Administrationsbereich auf die verschiedenen Datenbanken zuzugreifen und Änderungen vorzunehmen. Bei der Umsetzung wurde das bereits vorhandene Modell von Django für die Verwaltung der Authentifizierung und Autorisierung der Benutzer verwendet. Der Aufbau des Modells strukturiert sich wie folgt:

User-Model (Ausschnitt)

```
class AbstractUser(AbstractBaseUser, PermissionsMixin):
    username_validator = UnicodeUsernameValidator()
    username = models.CharField(
[...]
    first_name = models.CharField(_('first name'), max_length=30,
blank=True)
    last_name = models.CharField(_('last name'), max_length=150, blank=True)
    email = models.EmailField(_('email address'), blank=True)
[...]
```

Die somit erzeugte Datenbankstruktur ist über das Backend des Tools abrufbar (vgl. Abbildung 35).

The screenshot shows the 'Change user' interface in the ENIGMAS Backend. The page is divided into several sections:

- Benutzername und Passwort:** Fields for username and password.
- Benutzer-Informationen:** Fields for first name, last name, and email address.
- Benutzer-Status:** A section with a 'Status' dropdown menu.
- Benutzer-Gruppen:** A section with 'Available groups' and 'Choose groups' dropdowns.
- Benutzer-Berechtigung:** A section with 'Available user permissions' and 'Choose user permissions' dropdowns.
- Benutzer-Verhalten:** A section with 'Last login' and 'Data joined' fields, each with a date and time input.

Abbildung 35: Struktur der Benutzerinformationen im Backend der Webapplikation

Die Benutzerinformationen bestehen aus sechs Bereichen: Benutzername und Passwort, Benutzer-Informationen, -Status, -Gruppen, -Berechtigung und -Verhalten.

Mit dieser vollwertigen Benutzerverwaltung kann das Tool von mehreren Nutzern verschiedener Firmen gleichzeitig verwendet werden, ohne, dass diese Nutzer Informationen austauschen können. Zusätzlich ist es aber möglich Teilbereiche für Nutzer der gleichen Firma oder der gleichen Abteilung freizugeben.

Abbildung 36 zeigt zwei Webseiten-Formulare. Das linke Formular ist für die Registrierung ('Join Today') und enthält Felder für 'Username*', 'Email*' und 'Password*'. Es gibt auch eine 'Password confirmation*' und eine 'Sign Up' Button. Das rechte Formular ist für das Login ('Log In') und enthält Felder für 'Username*' und 'Password*', einen 'Log In' Button und einen Link 'Need an Account? Sign Up now'.

Abbildung 36: Abfrageformular für die Registrierung (links) und Login-Formular (rechts)

Die Datenbank (2) *Blog* verwaltet den wichtigsten Teil der Anwendung, nämlich die (ursprünglichen) Textdaten (*Reviews*), die Textanalysen (*Labeled Reviews*), die Label (*Labels*) und deren Verknüpfungen (vgl. Abbildung 34). Die einzelnen Datenpunkte und die Implementierung im Software-Code als *Model* (Djangos Bezeichnung für den Layer zur Strukturierung und Manipulation der Daten) zeigt Abbildung 37.

Review-Class im Blog-Model

```
class Review(models.Model):
    web_scraper_order = models.TextField(max_length=128, default='NULL')
    web_scraper_start_url = models.URLField(blank = True)
    author = models.CharField(max_length=128, null=True)
    title = models.TextField(blank = True)
    content = models.TextField(blank = True)
    rating = models.CharField(max_length=128, default='NULL')
    next_href = models.URLField(blank = True)
    date = models.TextField(blank = True)

    STATUS_CHOICES_FRANZ = (
        (1, 'neu'),
        (2, 'wip'),
        (3, 'done'),
    )
    status = models.IntegerField(choices=STATUS_CHOICES_FRANZ, default=1)

    def __str__(self):
        return self.web_scraper_order #|stringformat:"s"
```

Label-Class im Blog-Model

```

class Label(models.Model):
    label=models.CharField(max_length=200)
    color = models.CharField(max_length=120, default='primary')

    def __str__(self):
        return self.label

```

Anschließend werden die zwei Datenbanken *Review* und *Label* mit der Datenbank *LabeledReview* verknüpft (vgl. Abbildung 37). Dabei muss beachtet werden, dass der Bezug zwischen *LabeledReview* und *Review* in der Datenbank, so implementiert worden ist, dass beim Löschen der *Reviews* auch der verknüpfte Datensatz in der *LabeledReview*-Datenbank mitgelöscht wird. Beim Löschen eines *Labels* ist diese Logik nicht implementiert worden. Die Verknüpfung zwischen *LabeledReview* und *Label* ist ausschließlich durch die gleiche Bezeichnung gegeben.

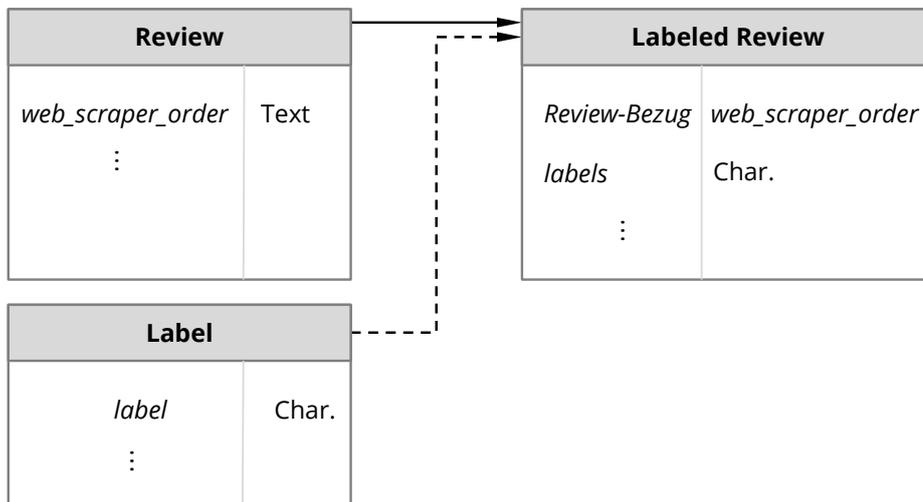


Abbildung 37: Datenbankstruktur der Rezension, Label und gelabelten Rezensionen.

Um diese Datenbankstruktur zu schaffen, wurde das eine Klasse *LabeledReview* entwickelt (vgl. Anhang A.4.1). Auf die Datenbankstruktur für die Speicherung der berechneten Stimmungswerte, der englischen Übersetzungen und anderer Informationen, soll hier nicht weiter im Detail eingegangen werden. Die Datenbank *LabeledReview* ist auch über das Backend des Tools abrufbar. Einen vollständig befüllten exemplarischen Datenpunkt zeigt Abbildung 38 links.

Die letzte Datenbank (3) speichert und verarbeitet die Profilinformatoren. Aus Sicherheitsgründen ist es gängige Praxis diese Informationen in einer separaten Datenbank zu speichern. Somit werden die kritischen Nutzerinformationen (Logindaten und Zugriffsrechte) und die Profilinformatoren separat gespeichert. Profilbild, Nutzernamen, Email und aktuelle Produkte sind die Informationen, welche in der Datenbank (3) verarbeitet werden (vgl. Abbildung 38 rechts).

The image shows two side-by-side screenshots from the ENIGMA web application. The left screenshot displays the 'Change labeled review' form for a review with ID 1596097389-4. The form includes fields for the review text ('Sehr wertiges und hochwertiges Werkzeug'), labels ('#Beitel'), and various sentiment and subjectivity scores for both German (DE) and English (ENG) versions. A 'Delete' button is visible at the bottom. The right screenshot shows the profile page for user 'wieck' (@uni-wuppertal.de). It features a profile picture, a bio, and a section for 'Aktuelle Produkte' which lists 'Stechbeitel 4-teilig'. The profile page also includes input fields for 'Username*' and 'Email*', and an 'Update' button.

Abbildung 38: Exemplarischer Datensatz der Datenbank LabeledReview (links) und Profilsicht eines Users (rechts)

Mithilfe dieser drei Datenbanken ist es nun möglich, alle benötigten Informationen zu verwalten, die notwendigen Funktionalitäten der Applikation abzubilden und die gesamte Anwendung auch außerhalb des lokalen Servers zur Verfügung zu stellen. Eine weitere Herausforderung besteht darin, die Datenbanken im Live-Betrieb zu verändern, ohne dabei Informationen zu löschen. Diese Herausforderung wird durch die Django-interne Funktion *Migration* abgebildet und ermöglicht eine updatestabile und erweiterbare Softwarelösung auch im Live-Betrieb.

6.4 Umsetzung der Applikation

Auch wenn diese wissenschaftliche Arbeit versucht, keinen reinen Softwarecode zu verwenden, sondern Funktionen und zugrundeliegende mathematische Berechnungen zu beschreiben – so bildet dieses Kapitel die größte Ausnahme. Die Gründe für die Umsetzung in Form einer Webanwendung sind: i) Geräteübergreifende Zugänglichkeit für Benutzer, ii) Integration mit anderen Systemen und iii) er-

höhte Flexibilität und Skalierbarkeit. In Anbetracht der derzeitigen Situation der KMU hat eine webbasierte Anwendung bessere Chancen auf Akzeptanz als eine lokale Implementierung, da sie sich leichter in bestehende Softwaresysteme integrieren lässt.

Für ein tieferes Verständnis der Struktur der Webanwendung, ist der Softwarecode unerlässlich. Wer die Webanwendung verstehen, verbessern oder gar erweitern möchte, sollte dieses Kapitel als Einführung in die Implementierung nutzen.

Es folgt eine kurze Erklärung zum Aufbau von Webframeworks. Die Beschreibung hilft die Funktionalitäten, Datenbankstrukturen und graphischen Benutzerschnittstellen (GUI) besser zu verstehen.

Für die Erarbeitung des Kapitels wurden neben der Online-Dokumentation der Django Software Foundation (2021), die Schriften von George (2016, 2020) und Melé (2020) verwendet.

Django ist eines von vielen verfügbaren Web-Frameworks, aber im letzten Jahrzehnt hat sich Django als führendes Framework für die Entwicklung skalierbarer, sicherer und wartbarer Webanwendungen hervorgetan. (George 2020, S. 5)

Was sind Webanwendungen²⁴ und wie sind diese strukturiert? Eine Webanwendung ist eine Software, die im Unterschied zu einer Desktopanwendung, nicht lokal auf dem Computer des Nutzers installiert ist. Die gesamte Datenverarbeitung findet auf einem Webserver statt. Für die Kommunikation zwischen Anwender und Webserver (Client-Server-Modell) gelten besondere Anforderungen (Betriebssystem, Laufzeitumgebung, http-Protokoll etc.). Für die Nutzung wird oftmals ein Webbrowser verwendet. Klassische Webframeworks sind nach dem Model-View-Control System aufgebaut und bestehen aus drei miteinander verbundenen Teilen: (1) **Model**, (2) **View** und (3) **Control**. Das **Model** definiert die Datenstruktur. Dies ist in der Regel eine Datenbank und bildet die Basisschicht einer Anwendung. Der **View**-Teil zeigt dem Anwender Daten mit HTML und CSS an. Dieser Teil ist am besten mit einer klassischen Websitearchitektur zu vergleichen. Der **Controller** verwaltet die Interaktion zwischen der Datenbank und der Ansicht. Das hier genutzte Framework Django verwendet ein ähnliches System: Ein Model-Template-View System (auch als MTV-Muster bezeichnet) (vgl. Abbildung 39). Das **Template** in Django ist mit dem **View**-Teil des MVC-Systems vergleichbar. Hintergrund dieser Änderung ist, dass moderne Webanwendungen (im Besonderen webbasierte Plattformen) ein immer wiederkehrendes Layout aufweisen (Amazon Produkt Listings, Pinterest Bilder Darstellung, Instagram oder Facebook Bilder oder Videodarstellungen etc.). Diese wiederkehrenden Layouts werden als Vorlagen (**Templates**) gespeichert und dann je nach Nutzer oder Situation mit verschiedenen Inhalten (Daten aus dem **Model**) dargestellt. Dies ist vergleichbar mit einer leeren Tageszeitungsseite, auf der zunächst leere Felder angeordnet sind, die dann täglich mit Texten und Bildern

²⁴ Web-Framework, Online-Anwendung, Webapplikation oder Web-App

gefüllt werden. Die Funktion des **Control**-Teils übernehmen bei Django die Teile **View** und **Template**. Dadurch erhalten bei Django **Templates** eine (begrenzte) Logik, der Hauptanteil der komplexeren logischen Verschachtelung und Abläufe wird über den **View** Teil organisiert.

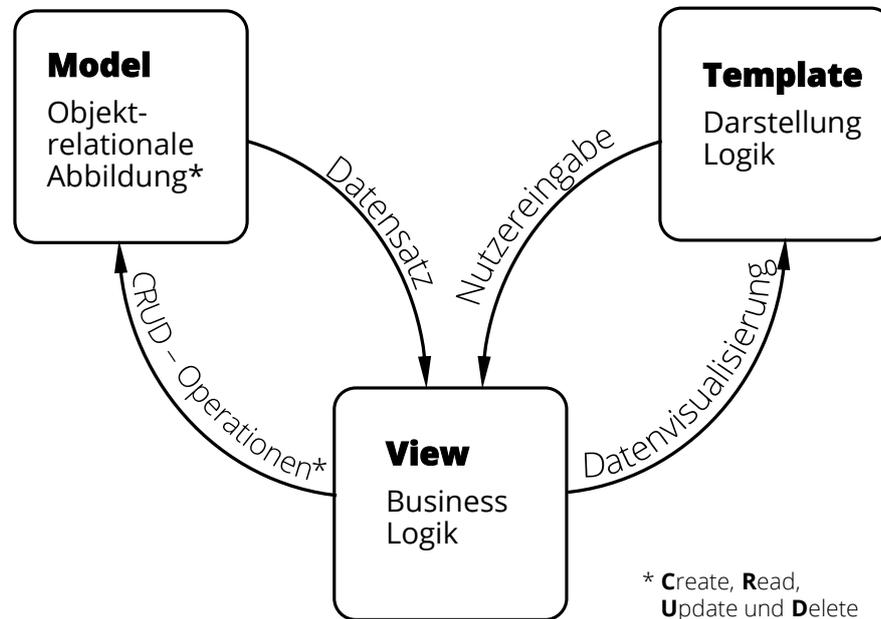


Abbildung 39: Django's Systemarchitektur MVT basierend auf dem Model-View-Control-Ansatz (angelehnt an George 2020, S. 17)

Diese drei Kernelemente werden nun im Detail beschrieben, beginnend mit dem **Model**. Die Datenverarbeitung bei Django basiert auf einem ORM (Object-relational Mapping). Die **Models** werden verwendet um auf die Datenbankstruktur zuzugreifen (CRUD - Create, Read, Uppdate und Delete). Eine komplexe Syntax (z.B. SQL) wird nicht verwendet. Ein sehr einfaches **Model** für eine Person, mit Vor- und Zuname wird in Django wie folgt definiert:

Model für eine Person:

```
from django.db import models
class Person(models.Model):
    first_name = models.CharField(max_length=30)
    last_name = models.CharField(max_length=30)
```

Eine besondere Stärke der Datenbankstrukturen bei Django liegt in der Möglichkeit, Datenbanken miteinander zu verknüpfen. Diese relationalen Datenbanken können drei Arten an Beziehungen zu einander haben: (1) **Many-to-One**, (2) **Many-to-Many** und (3) **One-to-One**. Anhand von drei realen Beispielen sollen die verschiedenen Datenbankbeziehungen erläutert werden.

Bei einer **Many-to-One** Beziehung stehen viele Instanzen einer Entität²⁵ in Bezug zu genau einer Instanz einer anderen Entität. Am Beispiel des Schachspiels soll diese Datenbankbeziehung kurz erläutert werden. Schachfiguren, die alle versuchen den gegnerischen König zu schlagen, stellen eine

²⁵ Eine Entität beschreibt einen eindeutigen, konkreten oder abstrakten Gegenstand und kann in diesem Fall als ein Model oder eine Datenbank interpretiert werden.

Many-to-One Beziehung dar. Um zwei Datenbanken mit einer Many-to-One Beziehung miteinander zu verknüpfen, verweist das eine Attribut der einen Datenbank auf ein anderes in der anderen Datenbank. Django nutzt dafür das Attribut *ForeignKey* (fremder Schlüssel).

Bei einer **Many-to-Many** Beziehung gibt es viele Instanzen einer Entität, die zu vielen Instanzen einer anderen Entität in Beziehung stehen. In Bezug auf das Schachspiel kann das erste Beispiel so abgeändert werden, dass nun alle weißen Schachfiguren gegen alle schwarzen Schachfiguren antreten. Das dafür notwendige Attribut in der Model Definition in Django ist `ManyToManyField`.

Bei einer (3) **One-to-One** Beziehung stehen genau zwei Instanzen zweier Entitäten in Beziehung zueinander. Zum Ende einer Schachpartie, dem sogenannten Endspiel, stehen in letzter Konsequenz nur noch die beiden Könige in einer One-to-One Beziehung gegenüber²⁶. Das dafür notwendige Attribut in der Model Definition in Django ist `OneToOneField`.

Diese Verknüpfung zwischen den Datenbanken auf Basis der Einträge ermöglicht oftmals ein einfacheres Datenmanagement. Beispielsweise ist es dadurch möglich, die Nutzerdaten (Passwort und Email-Adressen) und Profilinformationen mit einer One-to-One Beziehung derart zu verknüpfen, dass bei einem Löschen des Accounts automatisch die Profilinformationen ebenfalls gelöscht werden.

6.4.1 Models

Für die Umsetzung wurden mehrere Modelle in *Python* entwickelt. Diese Modelle ermöglichen eine einfachere Darstellung der Inhalte mit Forms in Django. Die im Folgenden dargestellten Modelle, sind die wichtigsten Modelle für die entwickelte webbasierte Applikation. Diese Modelle organisieren die Rohdaten der Reviews (*Review-Model*), die Label (*Label-Model*), die gelabelten Reviews (*LabeledReview-Model*) und die Auswertung (*GartenbaumAnalysis-Model*) setzen diese in Beziehungen zueinander (vgl. Anhang A.4.1).

6.4.2 Templates

Der nächste wichtige Bestandteil einer Django Webapplikation ist das sogenannte Template. Sie beinhalten die graphischen Elemente und stellen das Gerüst der visuellen Erscheinung der Webanwendung dar. Templates in Django können mit herkömmlichen Webseiten am ehesten verglichen werden. Sie setzen sich zusammen aus einem statischen und einen dynamischen Anteil. Der statische Anteil kann eine Navigationsleiste oder auch ein Footer (Fußbereich) sein (vgl. Abbildung 40).

Für moderne Webanwendungen sind oftmals die dynamischen Inhalte zentral. Zu diesen Inhalten zählen beispielsweise der Story-Feed der neuen Beiträge bei Instagram oder die neuen *Tweets* (Beiträge) bei Twitter. Um dynamische HTML-Inhalte bequem zu erstellen, verwendet Django Templates

²⁶ Den Schachregeln entsprechend wird dann die Partie als unentschieden (Remis) bewertet.



Abbildung 40: Schematischer Aufbau einer Webseite (Wireframe) (angelehnt an George 2020, S. 23)

. Für die darin verwendeten Variablen wird die Django Template Language (DTL) verwendet²⁷. Es ist wichtig zu verstehen, dass klassisches Web-Design und die Programmierung typischerweise getrennt erarbeitet werden. Dieser Logik entsprechend werden auch Templates ohne das Design entworfen. Vielmehr beinhalten Templates das WAS und nicht das WIE. Das nachfolgende Beispiel zeigt einen Ausschnitt.

```
<h1>Deine wichtigen Informationen</h1>
<p>Lieber {{ person_name }},</p>
```

Der in geschwungenen Klammern und fettgedruckte Teil der zweiten Zeile ist eine im DTL-Format beschriebene Variable, mit dem Ziel einen dynamischen Inhalt der Seite zu erzeugen, um den Namen der eingeloggten Person darzustellen. So wird die Variable `{{ person_name }}` - je nach Kontext – auf Basis des hinterlegten Modells durch den aktuellen Nutzernamen ersetzt. Beispielsweise wird dann aus dem oben definierten HTML-Code im Browser folgender Text dargestellt:

Deine wichtigen Informationen

Lieber Zuquini,

Diese Vorgehensweise scheint auf den ersten Blick sehr kompliziert und umständlich. In den klassischen Strukturen und Domänen der Web-Entwicklung ermöglicht es jedoch ein sehr modulares und dennoch vernetztes Entwicklungswerkzeug. Es ermöglicht Inhalt, Struktur und Layout voneinander getrennt zu bearbeiten und anschließend einfach zu verknüpfen. Für die Umsetzung der Webapplikationen wurden mehrere Templates erstellt. Anhand der zwei Templates *analytics.html* und *base.html*

²⁷ Alternativ für Jinja2

soll der Aufbau beispielhaft verdeutlicht werden. Anschließend wird anhand der Integration der Templates in das Django-spezifische MTV-Muster das Zusammenwirken und die Komplexität der einzelnen Komponenten beschrieben.

Template (Ausschnitt) von base.html:

```
[...]
<main role="main" class="container">
  <div class="row">
    <div class="col-md-12">
      {% if messages %}
        {% for message in messages %}
          <div class="alert alert - {{message.tags }}">
            {{message}}
          </div>
        {% endfor %}
      {% endif %}
      {% block content %}{% endblock content %}
    </div>
  </div>
</main>
[...]
```

Template analytics.html:

```
{% extends "blog/base.html" %}
{% block content%}
<h2>Analytics</h2>
<div>
  {% for LabeledReview in LabeledReviews %}
    <div class="card">
      <small><a href="{% url 'analytics-detail' LabeledReview.id %}"> {{
LabeledReview.webScraperOrder }} </a> </small>
      {{ LabeledReview.text }}
      {{ LabeledReview.labels }}
    </div>
  {% endfor %}
</div>

{% endblock content%}
```

Wie unterschiedlich der Softwarecode der analytics.html-Datei ausgegeben wird, beschreibt der nachfolgende Vergleich. Auf der einen Seite wird der Softwarecode nur im Webbrowser dargestellt und auf der anderen Seite vollständig in das entwickelte Django-Gesamtmodell integriert (vgl. Abbildung 41).

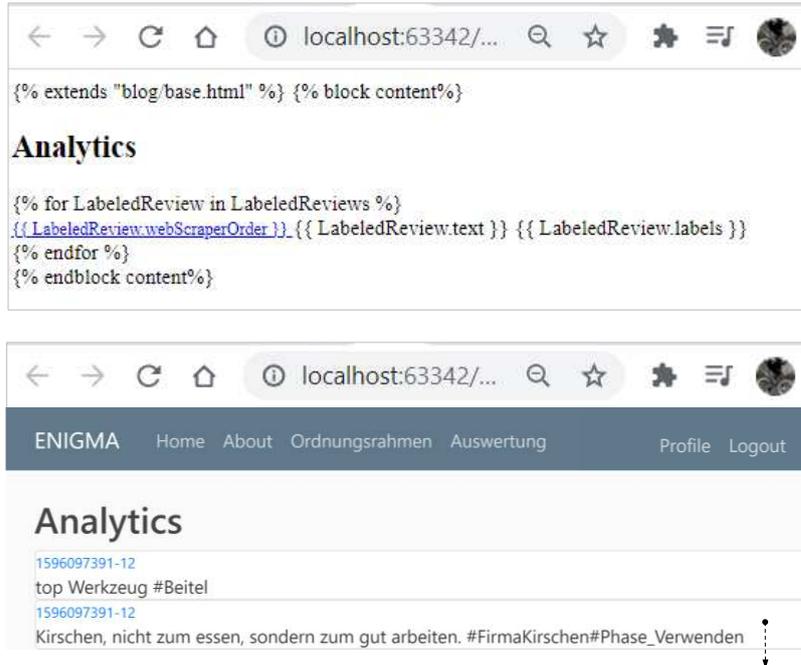
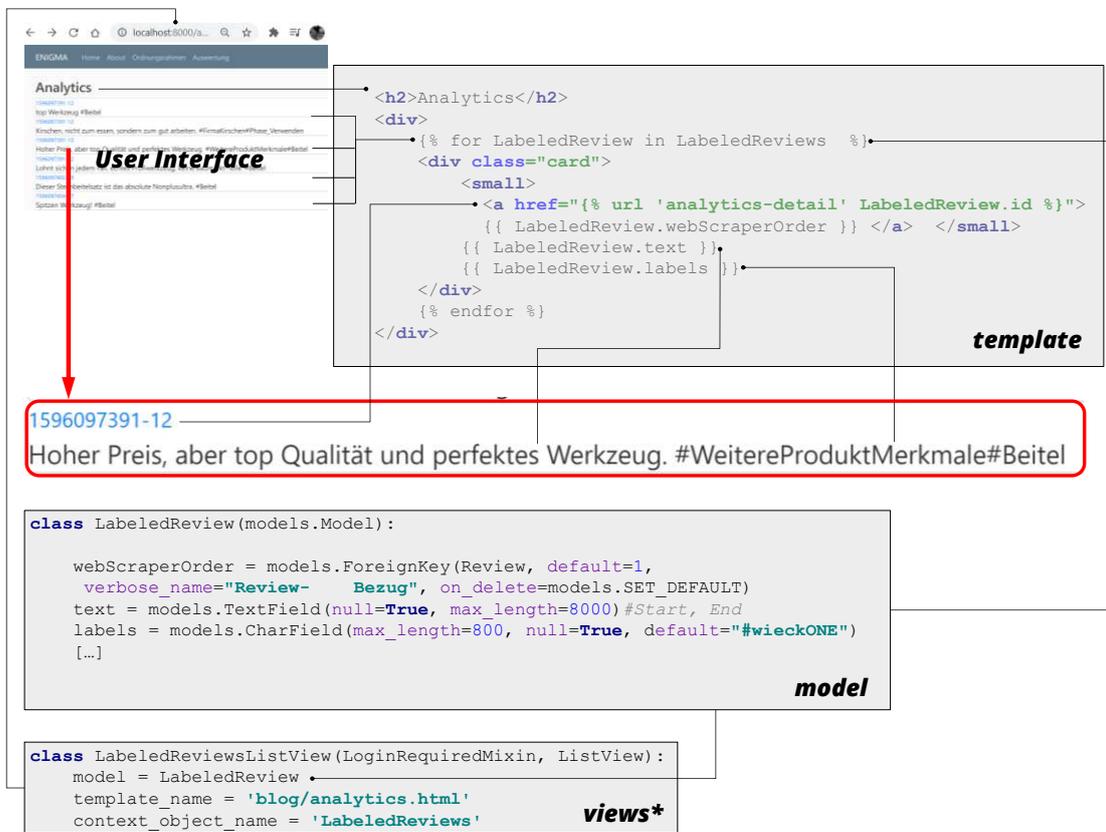


Abbildung 41: Visualisierung des Analytics-Templates ohne Integration (links) und mit Integration (rechts) in das Django-Projekt in einem Webbrowser

Wie die analytics.html-Datei im Detail mit den anderen Bestandteilen des Django-Modells interagiert zeigt die Abbildung 42 im Detail.



* Das Beispiel zeigt eine andere Art von View-Darstellungen - die sogenannten Class-Based-Views.

Abbildung 42: Beispielhafte Darstellung für den Zusammenhang zwischen Model-View-Template bei der Webapplikation

6.4.3 Views

Der letzte wichtige Bestandteil für eine Django Webapplikation ist der sogenannte View Teil. Die Bezeichnung View lässt vermuten, dass es sich wieder um die Verarbeitung der grafischen Elemente handelt. Das ist nicht der Fall. Die View-Funktion nimmt eine Web-Anfrage (request) entgegen, verarbeitet diese und gibt eine Web-Antwort (response) zurück. Für Applikationen werden alle Logik-Funktionen, kurz Views genannt, typischerweise in der Datei `views.py` gespeichert. In dieser Datei werden die Logik und die rechenintensiven Operationen der Webanwendung durchgeführt. Die Hauptdatei `views.py` besteht aus 1549 Zeilen Softwarecode und bildet den intelligenten Kern des entwickelten Tools. Die wichtigsten Funktionen sind **review_upload**, **home**, **createLabeledText**, **deep_analytics** und **create_science_table**. Die Umsetzung dieser Views in Software-Code und den damit notwendigen Import ist dem Anhang A.4.2 zu entnehmen. Im Nachfolgenden werden die einzelnen Funktionen und der Ablauf näher beschrieben.

Zu Beginn der Nutzung des Webtools ist es notwendig, dass die Rohdaten der Rezensionen hochgeladen werden. Das erfolgt über die Funktion **review_upload**. Neben der Überprüfung, ob es sich bei der hochgeladenen Datei um eine CSV-Datei handelt, werden die Rohdaten nach einem Schema importiert. Dieses Importschema überführt jeden einzelnen CSV-Eintrag in die Datenbankstruktur eines *Review-Models*. Dabei ist darauf zu achten, dass die CSV-Datei dieser Logik entspricht - andererseits muss dieses Importschema an dieser Stelle angepasst werden.

Nachdem die Rohdaten der Rezensionen hochgeladen worden sind, werden nun die einzelnen Rezensionen zum Labeln dargestellt. Diese Darstellung entspricht der Abbildung 30 und wird über die Home-Funktion bereitgestellt. Es werden nur die Rezensionen dargestellt, welche den Status 1 haben. Beim Hochladen haben alle Rezensionen diesen Status. Nachdem eine Rezension vollständig verschlagwortet wurde, wird der Status beim Speichern automatisch auf 2 geändert. Zusätzlich wurde hier eine Seitennummerierung (Paginator) implementiert. Dadurch wird immer nur eine Rezension pro Seite dargestellt.

Die Rohdaten aller Rezensionen wurden hochgeladen und so verarbeitet, dass pro Seite eine Rezension dargestellt wird. Nun können die Texte verschlagwortet werden. Die Funktion **createLabeledText** verarbeitet diesen Text und stellt sicher, dass die passenden Label richtig abgespeichert werden. Weiterhin wird überprüft, ob die Textpassage Teil der Rezension (Titel oder Inhalt) ist. Ist das erfüllt, so wird der Text als separater Datensatz gespeichert. Beim Speichern wird neben dem Text und den dazugehörigen Label auch die dazugehörige Rezension und die englische Übersetzung des Textes abgespeichert (Subfunktion *myTranslate*). Sollte dieser Speicherungsprozess nicht erfolgreich gewesen sein, erhält der Nutzer eine Fehlermeldung.

Nachdem nun alle Rezensionen vollständig verschlagwortet sind, kann die Auswertung beginnen. Die Auswertung wird in einer separaten Ansicht (DeepAnalytics) im Webtool dargestellt. Der Aufbau dieser Seite wurde bereits ausführlich beschrieben (vgl. Kapitel 6.3.4 und Abbildung 32). Der dazugehörige Software-Code verarbeitet alle gelabelten Textabschnitte und ermittelt die Kennzahlen für den gesamten Datensatz. Diese Berechnung wird immer wieder neu durchgeführt. Daher ist es möglich auch während der Verschlagwortung Zwischenergebnisse zu extrahieren. Alle wichtigen Kennzahlen dieser Berechnung werden dann in der Variable *Fakten* (entsprechend der Kontext-Konvention von Django) gespeichert und für die weitere Visualisierung der Daten genutzt (*create_science_table*).

Bemerkung: Die hier vorgestellten Funktionen der entwickelten Webanwendung stellen nur einen kleinen Ausschnitt der tatsächlich implementierten Funktionalitäten (URL, User-Data, Media, Data-Upload, Server-Implementation, Database-Updateability, Styling and Bootstrap, Error-Handling, Form-Handling, Seitenanzahl klickbar, Status-Änderungen, Admin and Super-User, Paginator, Exceptions und viele weitere) dar. Eine vollständige Dokumentation des gesamten Aufbaus ist für die vorliegende Arbeit nicht zielführend.

6.5 Zwischenfazit

Die praktische Implementierung der Lösungen für die Teilziele 1-3 (vgl. Kapitel 4.2) und die darüber hinaus definierten Anforderungen wurden mithilfe des entwickelten Tools umgesetzt. Für die Implementierung wurde das Webframework Django als Basis verwendet. Die Umsetzung erfolgte durch die Verwendung von State-of-the-Art NLP-Algorithmen, einer eigens entwickelten Methodik zum Labeln für nutzergenerierte Textdaten, zur Umsetzung der entwickelten Logik zur Auswertung der aufbereiteten Textdaten und zu einer integrierten, graphischen Aufbereitung der Analyseergebnisse. Dieses Tool ermöglicht somit Textdaten für die Produktentwicklung anforderungsgerecht auszuwerten, indem Kundenwahrnehmungen und –meinungen, mit den Produktinformationen verknüpft werden. Dieser werden anschließend automatisiert ausgewertet. Diese Implementierung ist skalierbar, da große Datenmengen und weitere Produkte integriert werden können. Weiterhin ist die Implementierung webfähig und kann somit von mehreren Anwendern und Firmen parallel über das Internet genutzt werden. Die Validierung des Tools erfolgt im nächsten Kapitel.

7 Validierung anhand einer realen Fallstudie

Das entwickelte Tool wird getestet. Anhand eines realen Datensatzes werden die einzelnen Teilziele chronologisch abgearbeitet und bewertet. Die Vorgehensbeschreibung stellt die typische Anwendung des Tools dar. Das Produktbeispiel ist ein Holzstechbeitel-Set. Dieses Produkt wurde ausgewählt, weil es verschiedene Nutzungsphasen, mehrere Produktkomponenten und eine geringe Komplexität aufweist. Es wurde darauf geachtet, dass die Anzahl an Rezensionen groß genug ist. Wie diese Rezensionen mit Hilfe des Tools ausgewertet werden, welche einzelnen Schritte dafür notwendig sind und was die Auswertung der Daten ermöglicht, wird im Folgenden näher beschrieben.

7.1 Datensatz: Rezensionen von *Amazon.de*

Der Datensatz besteht aus den Rezensionen auf der Website Amazon.de für das Stechbeitel-Set der Firma Wilhelm Schmitt Comp. GmbH & Co. KG Kirschen Werkzeuge. Die Daten von der Website wurden mit einem sogenannten Webscraper lokal als Textdatei (im csv-Format) gespeichert. Dieses Programm durchsucht Webseiten und extrahiert ausgewählte Inhalte. Anschließend wurde der Datensatz entsprechend vorbereitet, sodass nur deutschsprachige Rezensionen vorliegen. Die Struktur des Datensatzes stellt Tabelle 8 dar. Dieser Rohdatensatz besteht aus 99 potentiell nützlichen Rezensionen, liegt als .csv-Datei vor und dient als Ausgangsbasis für die anschließende Validierung des Tools.

Tabelle 8: Exemplarischer Aufbau und Bestandteile des verwendeten Datensatzes

Erklärung	Bezeichnung	Inhalt
ID	web-scraper-order	1596097393-24
Website-URL	web-scraper-start-url	https://www.amazon.de/Kirschen-1141000-Spezial-Werkzeugstahl-Handgeschmiedet[...]e=all_reviews
Autor	author	Nicky
Titel	title	Toll gemacht
Datum	date	Rezension aus Deutschland vom 4. Juni 2019
Textinhalt	content	Habe es meinem Mann zum Geburtstag geschenkt. Er war sehr begeistert
Sternebewertung	rating	5,0 von 5 Sternen
Aktion-Button	next	Weiter→
Nächste URL	next-href	https://www.amazon.de/Kirschen-1141000-Spezial-Werkzeugstahl-Handgeschmiedet[...]all_reviews

7.2 Label erstellen

Für den Beginn werden die produktspezifischen Label auf Basis des Produktbildes und des Herstellers bzw. des Markennamens erstellt (vgl. Abbildung 43). Diese Liste an Merkmalen wird als Basis für die Label genutzt und werden sukzessive im weiteren Verlauf des Label-Vorgangs erweitert. Aus diesem Vorgehen, basierend auf der in Teilziel 1 hergeleiteten Methode und der Abbildung 19, ergeben sich

folgende Label: Beitel, Klinge, Griff, Aufkleber, Aufbewahrung (Kiste), Firma Kirschen und weitere Merkmale.

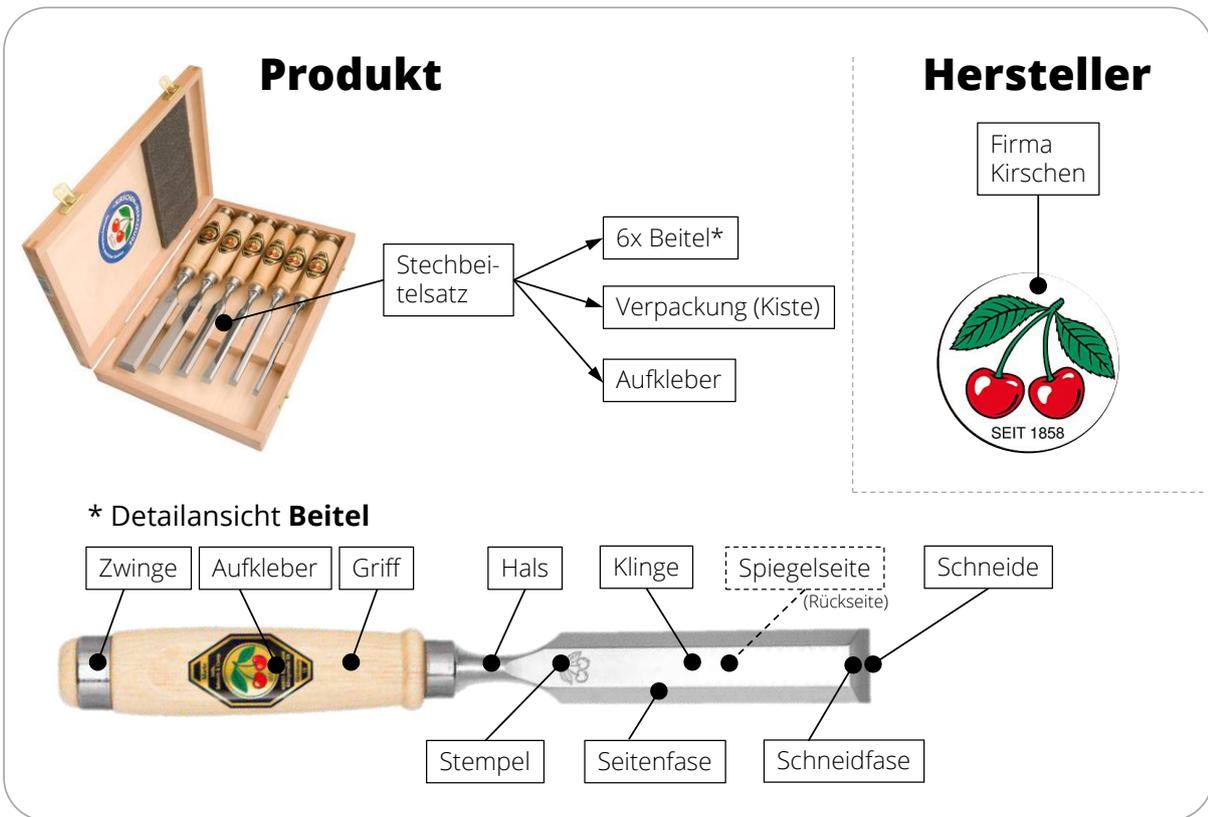


Abbildung 43: Auswahl der Produktmerkmale eines Stechbeitel-Sets der Firma Kirschen

Diese Liste an produktspezifischen oder kontextsensitiven Label, wird mit den Label für die Nutzerlebnisse (*VorGebrauch*, *WährendGebrauch* und *NachGebrauch*) und die sinnvoll erachteten Lebensdauerphasen des ausgewählten Produktes erweitert (vgl. Tabelle 9). Mithilfe dieser 13 Label werden im nächsten Schritt die extrahierten Rezensionen nach der in der vorliegenden Arbeit entwickelten Methode analysiert und kategorisiert.

Tabelle 9: Auswahl der relevanten Lebensdauerphasen für das Produktbeispiel

Lebensdauerphasen	Produkt-eignung
Transport	-
Montage und Installation	-
in Betrieb nehmen	X
Einrichten	-
Verwenden / Betreiben	X
Reinigung	-
Instandhaltung / Wartung	X
Demontage	-
außer Betrieb nehmen	-
Entsorgung	-

7.3 Rezensionen labeln

Entsprechend der entwickelten Methode (Teilziel 1) werden die Rezensionen mithilfe der Label gruppiert. Dabei wird darauf geachtet, dass die hergeleiteten Kenn- und Zielwerte (Tabelle 6) angestrebt werden. Das folgende Beispiel verdeutlicht das Vorgehen und die grafische Aufbereitung des Tools. Den Aufbau des Tools stellt Abbildung 45 dar und besteht aus (Beschreibung von von oben nach unten): Metadaten, Ort und Zeit der Rezension, interne Identifikationsnummer und vergebene Sternbewertung. Dann folgt der Titel und der Rezensionstext. Daran schließt die Auflistung der auswählbaren Label und ein freies Textfeld an, in das der zu labelnde Text eingefügt wird. Die schon gelabelten Textabschnitte dieser Rezension und die dafür vergebenen Label bilden den Abschluss. Diese Rezension besteht aus einem Titel und weiteren acht Sätzen. Der folgende Satz dieser Rezension ist besonders interessant, da dieser die aktuellen Label um einen wichtigen Aspekt erweitert:

„[...] Ich hatte vorher einen günstigen Beitel aus dem Baumarkt probiert (Billigstahl aus China), aber die Klinge war schon beim ersten Loch schartig. [...]“

Der Satz enthält Informationen zu Beitel, Gebrauch, Klinge und weitere Merkmale. Und besonders interessant, ein Konkurrenzprodukt wird erwähnt. Diese Informationen helfen, die Wechselbereitschaft der Nutzenden und ihrer bisherigen Erfahrungen zu ermitteln und anschließend zu bewerten. Um die Auswertung der einzelnen Attribute wie z.B. *Beitel*, *VorGebrauch*, *WährendGebrauch* etc. nicht zu verfälschen, muss dieser Satz separat analysiert werden. Eine Möglichkeit besteht darin ein zusätzliches Label *Competitor* (Konkurrenzprodukt) zu erstellen. In dem hier vorgestellten Beispiel wird das Label nicht weiter berücksichtigt. Für diese Rezension wurden 7 Wortgruppen mit Label versehen. Dafür wurden 6 Label insgesamt 12 Mal verwendet. Die zwei Label *Beitel* und *weitereProduktmerkmale* wurden am häufigsten (je 3-mal) genutzt. Die entsprechende Nomenklatur für diesen Sachverhalt stellt Abbildung 44 dar.



Abbildung 44: Unterscheidung zwischen Gesamtnutzung und Anzahl Rezensionen

Diese erste Auswertung beschreibt zwar im ersten Schritt nur die Häufigkeitsverteilung, stellt aber im Gesamtprozess der Verarbeitung der Rezensionen den wichtigsten und gleichzeitig aufwendigsten Prozessschritt dar. Wenn alle Rezensionen verschlagwortet sind, dann kann die Auswertung der einzelnen Label und die Stimmungsermittlung durchgeführt werden (vgl. Abbildung 45).

ENIGMA
Profile Logout

Datenbank Überblick
Neues Label hinzufügen
Daten herunterladen

Bunnes Rezension aus Deutschland vom 12. August 2016 | WebScrapper-Id: 1596097406-86 | Rating: 5,0 von 5 Sternen

Qualität hat ihren Preis

Die Beitel sind 1A verarbeitet und der Stahl ist sehr schnitthaltig. Auch nach dem Stemmen/Klopfen von mehreren Zapfenlöchern war nur einmal ein Abziehen der Klingen auf Leder nötig. Die Beitel gehen einwandfrei ins Holz und die Klinge bleibt einwandfrei. Man sollte bei Werkzeug nicht so sehr auf den Preis schauen, wenn man etwas ordentliches möchte. Ich hatte vorher einen günstigen Beitel aus dem Baumarkt probiert (Billigstahl aus China), aber die Klinge war schon beim ersten Loch scharf. Das Geld hätte ich auch wegwerfen können. Der Holzkasten ist auch recht schick zur Aufbewahrung. Dieses Qualitätswerkzeug ist absolut empfehlenswert!

#FirmaKirschen
 #Sales
 #Phase_InBetriebNehmen
 #Phase_Verwenden
 #Phase_Warten

#Intensivgebrauch
 #VorGebrauch
 #WährendGebrauch
 #NachGebrauch
 #USER

#Beitel
 #Klinge
 #Griff
 #Aufkleber
 #Spiegelseite
 #WeitereProduktMerkmale

#Qualität
 #Kiste

SPEICHERN

Speichern und zur nächsten Rezension
Labeled Sätze

#WeitereProduktMerkmale#Qualität	Qualität hat ihren Preis
#Beitel#Klinge#WeitereProduktMerkmale	Die Beitel sind 1A verarbeitet und der Stahl ist sehr schnitthaltig.
#Klinge#Phase_Verwenden#Phase_Warten	Auch nach dem Stemmen/Klopfen von mehreren Zapfenlöchern war nur einmal ein Abziehen der Klingen auf Leder nötig.
#Beitel#Klinge#Phase_Verwenden	Die Beitel gehen einwandfrei ins Holz und die Klinge bleibt einwandfrei.
#Beitel#WeitereProduktMerkmale	Man sollte bei Werkzeug nicht so sehr auf den Preis schauen, wenn man etwas ordentliches möchte.
#Kiste	Der Holzkasten ist auch recht schick zur Aufbewahrung.
#Beitel#Qualität	Dieses Qualitätswerkzeug ist absolut empfehlenswert!

1
2
»

97 / 99 Rezensionen completed

Abbildung 45: Details einer gelabelten Rezension

7.4 Textanalyse

Für jeden gelabelten Textabschnitt wird Stimmung und Subjektivität berechnet und einem Label zugeordnet. Für die Verarbeitung der Texte wurde die Python-Bibliothek *TextBlob* verwendet. In einer Voruntersuchung fiel auf, dass die deutsche Adaption dieses Softwarepaketes *TextBlobDE* andere

Ergebnisse liefert als die englische Variante (vgl. Abbildung 46). Zur Auswertung der Textpassagen werden beide Softwarebibliotheken verwendet und im oberen Bereich der Tabelle verglichen. Für die weitere Verarbeitung wird ausschließlich der performantere Algorithmus (der für die englische Sprache ausgelegt) verwendet. Das bietet die Möglichkeit zukünftige Verbesserungen für deutschsprachige Textverarbeitungsalgorithmen direkt zu integrieren und anschließend zu vergleichen.

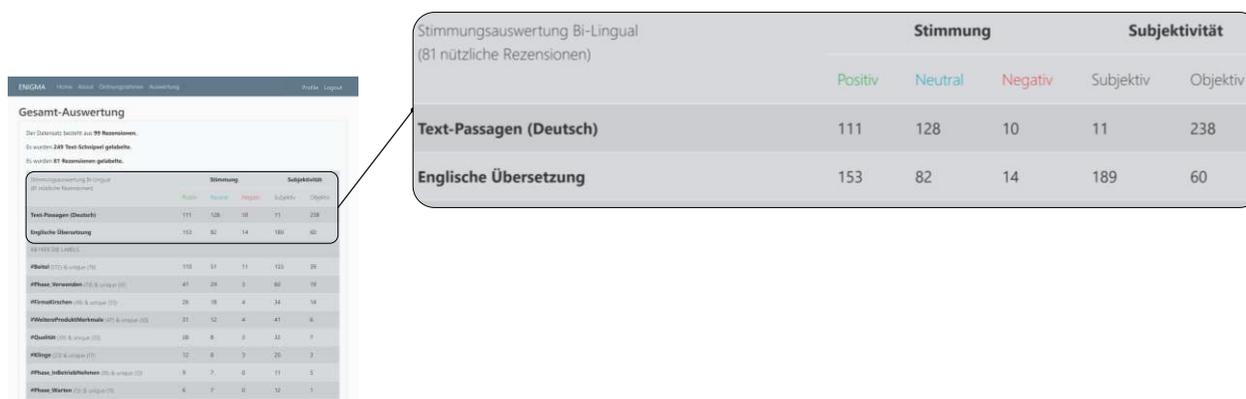


Abbildung 46: Vergleich der Ergebnisse der Textanalyse in Abhängigkeit von der Sprache (als integrierte Funktion des Tools)

Den quantitativen Unterschied der Ergebnisse stellt Tabelle 10 dar. Die prozentuale Veränderung ist sowohl bei der Subjektivität mit 71,5% als auch bei der Stimmung mit 18,5% dramatisch. Anders formuliert: Der englische Algorithmus identifiziert mehr Worte in den Textpassagen und berechnet dadurch für mehr Textpassagen sinnvolle (von Null verschiedene) Werte. Daraus resultiert, dass die Anzahl der *neutralen* (dazugehöriger Wert $\rightarrow 0$) und *objektiven* (*dazugehöriger Wert* $\rightarrow 0$) Textpassagen abnimmt und die Anzahl positiver (Wert $\rightarrow 1$), negativer (Wert $\rightarrow -1$) und subjektiver (Wert $\rightarrow 1$) steigt. Dieses Ergebnis wurde qualitativ mithilfe von Stichproben überprüft. Das Ergebnis konnte damit bestätigt und das Vorgehen als geeignet bewertet werden.

Tabelle 10: Detailuntersuchung der Veränderung der Ergebnisse der Textanalyse in Abhängigkeit von der Sprache

		Stimmung			Subjektivität	
		Positiv	Neutral	Negativ	Subjektiv	Objektiv
Deutsche Textpassage	Absolut [n=249]	111	128	10	11	238
	Relativ	44,6%	51,4%	4,0%	4,4%	95,6%
Englische Übersetzung	Absolut [n=249]	153	82	14	189	60
	Relativ	61,4%	32,9%	5,6%	75,9%	24,1%
Änderungsrate (Deutsch zu Englisch)	Absolut	42	-46	4	178	
	Relativ	16,9%	-18,5%	1,6%	71,5%	

Zusätzlich wurde der Unterschied zwischen der deutschen und der englischen Auswertung für jede einzelne Textpassage berechnet und graphisch ausgewertet. Die Stimmungs- und Subjektivitätswerte wurden ermittelt und auf einer Zeitachse (entsprechend des Erstellungsdatums der Rezension) aufgetragen. Abbildung 47 stellt den Unterschied zwischen den englischen Texten (vgl. Abbildung 47 - II Quadrant) und den deutschen Texten (vgl. Abbildung 47 - I Quadrant) dar. Die Ergebnisse zeigen, dass

die Subjektivität im Deutschen kaum sinnvoll berechnet wurde. Die Differenz (englische Werte minus deutsche Werte) stellt der III Quadrant und der Betrag davon der IV Quadrant dar.

Aus diesem Zwischenergebnis lässt sich ableiten, dass die englische Übersetzung, für die Verarbeitung der Textdaten verwendet, bessere Ergebnisse liefert. Für die Implementierung in das Tool wird die native Schnittstelle zu dem von Google angebotenen Übersetzungsprogramm verwendet. Eine deutliche Verbesserung der Ergebnisse würde die Integration der von Deepl.com angebotenen Übersetzungssoftware bieten. Aus technischen Gründen ist diese Integration aber aktuell nicht möglich.

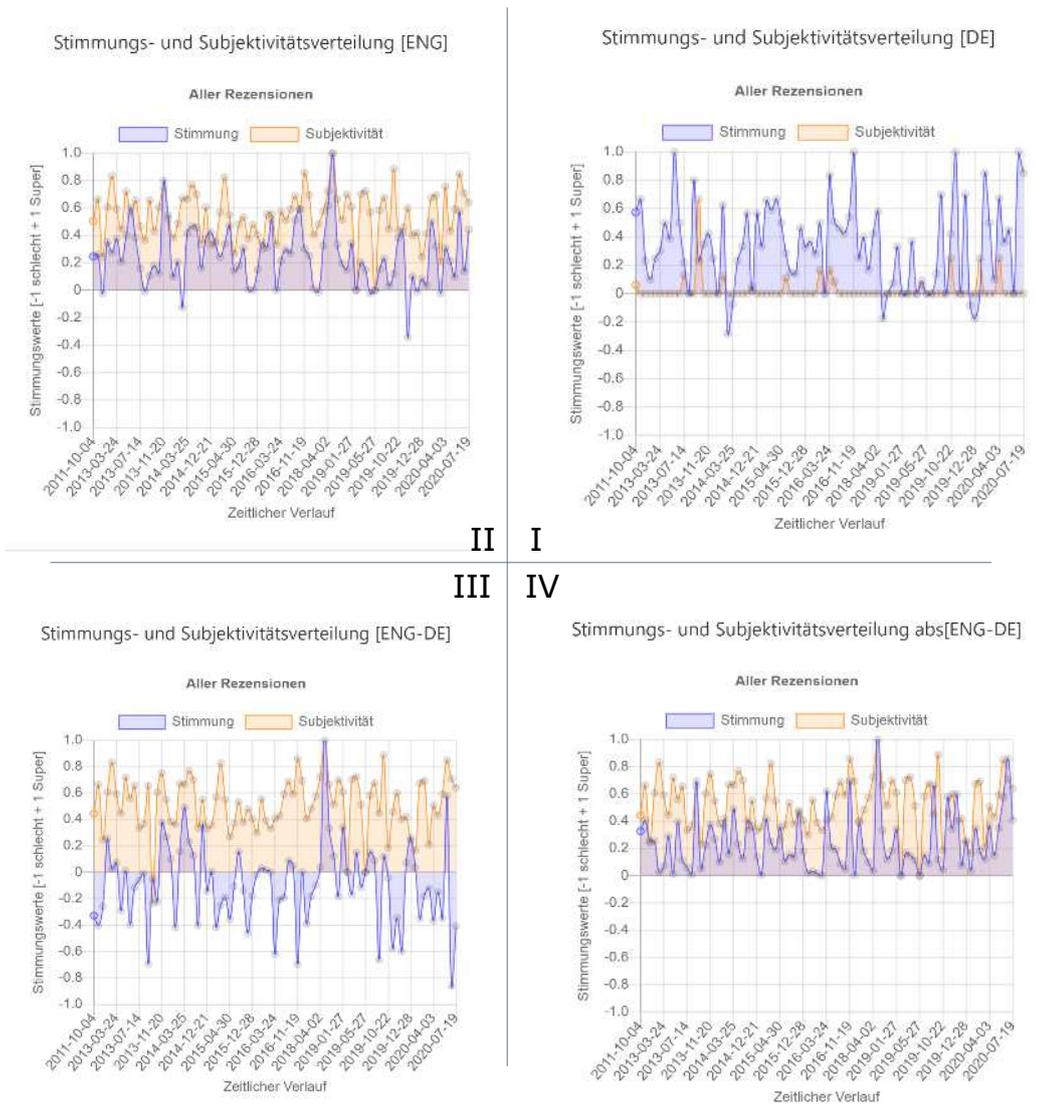


Abbildung 47: Vergleich der Ergebnisse deutscher und englischer Implementierung

7.5 Auswertung und Visualisierung

Die Auswertung mithilfe des Tools erfolgt automatisiert mithilfe eines Dashboards. Das Dashboard besteht aus den Bereichen: **Meta-Analyse**, **Label-Auswertung**, **Diagramm-Bereich** und **Textdetail-Bereich**.

Die **Meta-Analyse** wertet die „Nützlichkeit“ und die Größe des Datensatzes aus. Für den verwendeten Datensatz wurden 99 Rezensionen als potentiell nützlich identifiziert, wobei nur 81 Rezensionen inhaltlich nützliche Aussagen beinhalteten (vgl. Abbildung 48). Weiterhin wurden die 81 Rezensionen in 249 Textpassagen (hier als Text-Schnipsel bezeichnet) unterteilt und gelabelt. Pro Rezension wurden durchschnittlich 3 Textpassagen gelabelt. Wie oft welches Label verwendet wurde, stellt die Label-Auswertung dar.



Abbildung 48: Auswertung - Anzahl nützlicher Rezensionen und gelabelter Textpassagen (Toolausschnitt)

Die **Label-Auswertung** zeigt, dass 14 Label benötigt wurden, um alle Sätze in den Rezensionen zu kategorisieren (vgl. Abbildung 49). Die Reihenfolge der Label wurde absteigend nach Häufigkeit sortiert. Es war zu erwarten, dass das Produkt (**#Beitel**) und die Nutzung (**#Verwendung**) am häufigsten in den Rezensionen erwähnt werden. Bemerkenswert ist, dass die Marke (**#FirmaKirschen**) in 48 Textpassagen erwähnt wird und damit das dritthäufigste Label ist. Weiterhin ist aus dieser Analyse abzuleiten, dass die am meisten verwendeten Themen zum größten Teil positiv und subjektiv erwähnt wurden. Im Gegensatz dazu, werden die Themen, welche die Klinge betreffen (**#Klinge**, **#Phase_inBetriebNehmen**, **#Phase_Warten**, **#Griff**) weniger stark positiv bewertet. Noch weniger wurden die **#Kiste**, **#Spiegelseite** und **#Aufkleber** erwähnt. Die letzten 2 Label scheinen Einzelfälle zu sein und werden hier in der Untersuchung nicht näher erläutert.



Abbildung 49: Auswertung - Anzahl verwendeter Label, Stimmung und Subjektivität

Aus der Häufigkeitsverteilung der Label, lassen sich drei Gruppen von Kundenwahrnehmungen in Bezug auf das Produkt ermitteln (vgl. Abbildung 50):

1. Gruppe: Beitel, Verwendung, Markenauftritt, Weitere Produktmerkmale und Qualität werden besonders häufig genannt. Wenn diese Themen genannt werden, dann geschieht dies vermehrt sehr positiv und sehr subjektiv. Daraus lässt sich ableiten, dass dieser Merkmale für die Kunden dieses

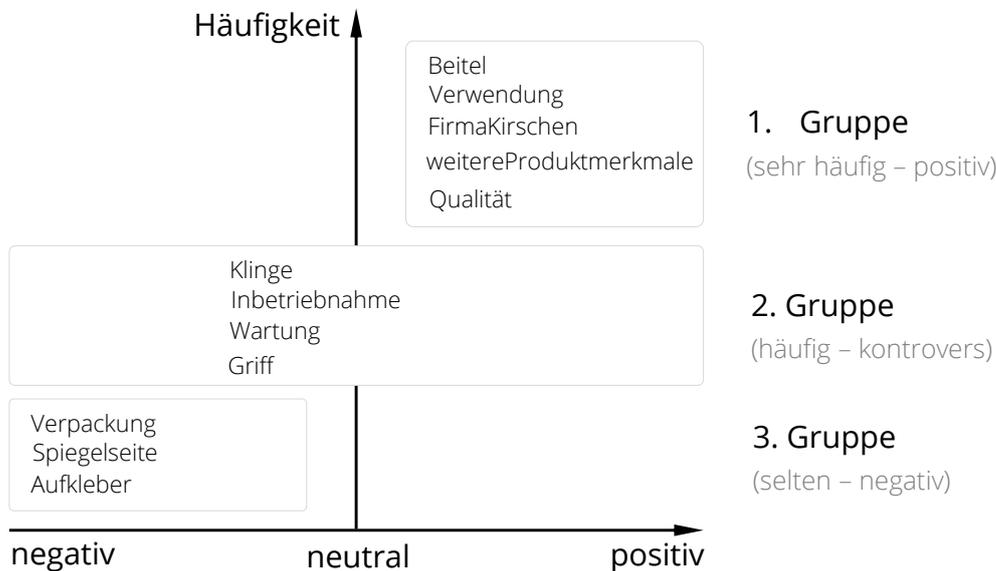


Abbildung 50: Auswertung der Label mittels Gruppenbildung

Produktes besonders wichtig sind und aktuell vom Hersteller gut erfüllt werden. Für die Produktentwicklung bedeutet es, dass diese Merkmale für zukünftige Produktgenerationen bleiben und mitunter sogar noch verbessert werden sollten. Bei der Kundenansprache in Werbung oder Verkaufsgesprächen sind das ebenfalls die zentralen Merkmale. Für Konkurrenzprodukte müssen diese Merkmale mindestens gleichwertig erfüllt sein um bei der gleichen Kundengruppe konkurrenzfähig zu sein (ungeachtet des Preises).

2. Gruppe: Klinge, Inbetriebnahme, Wartung und Griff sind für die Kunden wichtige Themen, die mitunter kontrovers wahrgenommen wurden. Daraus lässt sich ableiten, dass in diesem Bereich (interne) Verbesserungen möglich sind und dass hier Potentiale für Konkurrenzprodukte bestehen. Das Potential besteht sowohl für den Ingenieurbereich auf technologischer Ebene, als auch für den Marketingbereich. Durch eine direkte Ansprache dieser Themen wird eine adressatengerechtere Kommunikation möglich.

3. Gruppe: Die Aufbewahrungsbox, die Spiegelseite des Beitels und der Aufkleber sind am häufigsten negativ angesprochen worden. Daraus lässt sich ableiten, dass hier sowohl der größte als auch der dringlichste Handlungsbedarf für das Unternehmen besteht Verbesserungen am Produkt vorzunehmen. Für Konkurrenzprodukte besteht hier die größte Chance diese Kriterien besonders stark zu fokussieren (technologisch und kommunikativ).

Neben diesen praktischen und direkt nutzbaren Handlungsempfehlungen und Auswertungen der Rezensionen wird aus akademischen Zwecken die Auswertung noch detaillierter betrachtet. Diese Auswertung ist nicht in das Tool integriert, da es für den praktischen Anwender keine neuen Erkenntnisse liefert. Für die Bewertung der Funktionsweise des Tools und die datengetriebene Analyse des Datensatzes wird diese Analyse im Folgenden weiter ausgebaut.

Tabelle 11 zeigt die fünf häufigsten Label und deren prozentuale Verteilung der Stimmungs- und Subjektivitätswerte. Dabei werden die Prozentzahlen sowohl für das einzelne Label (relativ) und als auch in Bezug zu den gesamten Textpassagen (relativ zu allen - n=249) berechnet. Das folgende Beispiel für #Label soll diese zwei Berechnungen und die draus resultierenden Aussagen verdeutlichen:

Beitel wird insgesamt in 172 Rezensionen erwähnt. Davon 110-mal positiv (64%), 51 neutral (29,7%) und 11-mal negativ (6,4%). 133 Textpassagen waren subjektiv und 39 objektiv formuliert. Im Bezug zu dem gesamten Datensatz ergibt sich die folgende prozentuale Verteilung: in 44,2% aller gelabelten Rezensionen wurde der *Beitel* als positiv beschrieben, in 20,5% als neutral und in 4,4% als negativ. In 53,4% aller gelabelten Rezensionen wurde das Wort *Beitel* in einem subjektiv bewerteten Kontext beschrieben - in 15,7% aller Fälle objektiv. Die gesamte Auswertung dieser Gegenüberstellung ist dem Anhang (vgl. Anhang A.5- Tabelle 13) zu entnehmen.

Tabelle 11: Erweiterte Auswertung - Anzahl verwendeter Label und Häufigkeitsverteilung prozentual

n_gesamt = 249		Stimmung			Subjektivität	
		Positiv	Neutral	Negativ	Subjektiv	Objektiv
#Beitel	Absolut [n=172]	110	51	11	133	39
	Relativ	64,0%	29,7%	6,4%	77,3%	22,7%
	Relativ zu allen	44,2%	20,5%	4,4%	53,4%	15,7%
#Phase_Verwenden	Absolut [n=79]	47	29	3	60	19
	Relativ	59,5%	36,7%	3,8%	75,9%	24,1%
	Relativ zu allen	18,9%	11,6%	1,2%	24,1%	7,6%
#FirmaKirschen	Absolut [n=48]	26	18	4	34	14
	Relativ	54,2%	37,5%	8,3%	70,8%	29,2%
	Relativ zu allen	10,4%	7,2%	1,6%	13,7%	5,6%
#WeiterProduktmerkmale	Absolut [n=47]	31	12	4	41	6
	Relativ	66,0%	25,5%	8,5%	87,2%	12,8%
	Relativ zu allen	12,4%	4,8%	1,6%	16,5%	2,4%
#Qualität	Absolut [n=39]	28	8	3	32	7
	Relativ	71,8%	20,5%	7,7%	82,1%	17,9%
	Relativ zu allen	11,2%	3,2%	1,2%	12,9%	2,8%

Die vorletzte Auswertung stellt der *Diagramm-Bereich* dar. Der Diagramm-Bereich zeigt drei verschiedene Auswertungsmöglichkeiten mithilfe von zwei verschiedenen Diagrammtypen (vgl. Abbildung 51).

Das **Säulendiagramm** wertet den zeitlichen Verlauf der Anzahl der Rezensionen aus. Für den untersuchten Datensatz liegt das Maximum im Jahr 2019 mit 24 Rezensionen und das Minimum mit einer Rezension im Jahr 2011 (vgl. Abbildung 51 links). Die Aussage dieser Analyse ist für den vorliegenden Datensatz nicht eindeutig, da weder Trends noch Zyklen zu erkennen sind. Von besonderem

Interesse sind die Jahre 2017 und 2018, in denen die Zahl der Rezensionen von 11 auf 4 sank, um dann 2019 wieder auf ein Allzeithoch von 24 zu steigen. Diese Veränderungen können auf zusätzliche Mitbewerber auf dem Markt oder andere externe Einflussfaktoren hindeuten. Diese Annahmen bleiben auf der aktuellen Wissensbasis Vermutungen. Neben dem Säulendiagramm existieren noch zwei Liniendiagramme.

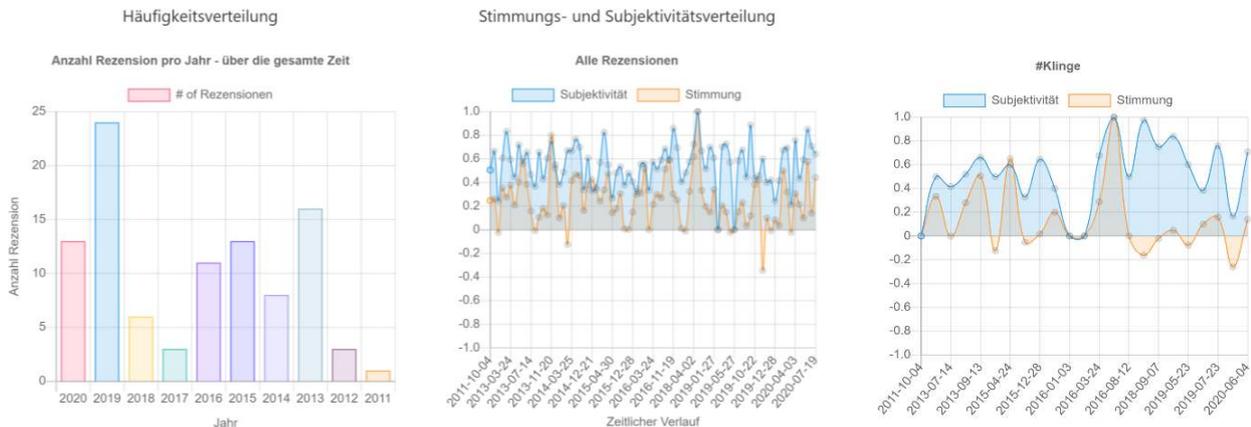


Abbildung 51: Verschiedene Auswertungen aus dem Diagramm-Bereich (links: Anzahl Rezensionen pro Jahr; mitte: Stimmungs- und Subjektivitätswerte für die Gesamtrezensionen; rechts: Stimmungs- und Subjektivitätswerte für das Label #Klinge

Das **erste Liniendiagramm** stellt die Stimmungs- und Subjektivitätswerte der gesamten Rezension in Abhängigkeit des Verfassungsdatums dar. Für den untersuchten Datensatz ergibt sich daher, dass die Stimmungs- und Subjektivitätswerte der Gesamtrezension über den gesamten Zeitraum fast durchweg über der Abszisse liegen (vgl. Abbildung 51 mitte). Weiterhin ist zu erkennen, dass die berechneten Subjektivitätswerte für die Gesamtrezensionen immer höher sind als die der Stimmungswerte. Von besonderem Interesse sind die lokalen positiven Extrema: 0,8 und 1,0 und das negative Extremum das bei - 0,35 liegt. Der Großteil der Rezensionen befindet sich in dem Wertebereich von 0,1-0,5. Diese Werte lassen vermuten, dass die Kunden die Gesamterfahrung mit dem Produkt fast durchweg positiv einschätzen. Die Extrema sollten besonders überprüft werden. Folgende Fragen sind dafür denkbar: Welche Merkmale wurden angesprochen? Können diese Informationen genutzt werden, um den Durchschnittswert zu erhöhen? Wie könnten diese Merkmale helfen, die Kundenwahrnehmung mit dem Produkt weiterhin verbessern

Das **zweite Liniendiagramm** stellt den Graph der Stimmungs- und Subjektivitätswerte über den gesamten Zeitraum für einzelne Label dar. Diese Auswertung ermöglicht eine detailliertere Auswertung: das Erkennen von produktspezifischen Schwachstellen und die Identifikation möglicher Verbesserungen. Für den untersuchten Datensatz ergibt sich für das Label #Klinge ein Graph, der in der ersten Hälfte stetig positiv ansteigt, dann schlagartig abfällt und sich langsam wieder aufbaut (vgl. Abbildung 51 rechts). Diese Interpretation des Graphen ist ablesbar, wenn der immer wieder auf null zurückfallende Graph nicht berücksichtigt wird. Der Verlauf des Graphs für die Stimmung zeigt ebenfalls, dass das negative Extremum in der frühen Vergangenheit liegt. Dieser Verlauf könnte auf Qualitätsmängel oder Veränderungen in den Produkteigenschaften der Klinge zurückzuführen sein und

sollte – vor allem durch die erst kürzlich verfasste Rezension – berücksichtigt und analysiert werden. Die Subjektivitätswerte für das Label #Klinge sind sehr hoch und lassen vermuten, dass die Klinge oftmals sehr emotional und subjektiv beschrieben wird.

Abbildung 52 zeigt die Auswertung der weiteren Label. Der Vergleich der verschiedenen Graphen zeigt, dass die Subjektivität auch bei anderen Labeln oftmals höher ist als die Stimmungswerte. Dadurch ist es möglich personenspezifische Kundenbedürfnisse zu bewerten. Dieser Zusammenhang sollte mithilfe einer quantitativen Studie erprobt werden. Darüber hinaus ist festzustellen, dass die Label zu den Produktlebenszyklusphasen eine geringere Stimmungsintensität als die Produktmerkmale zeigen. Bei den Phasen *Warten* und *In Betrieb nehmen* ist es nicht verwunderlich, dass diese Phasen eher negativ und weniger stark subjektiv beschrieben wurden. Dennoch zeigt sich, dass hier Verbesserungspotential in der User-Experience besteht. Das gewählte Visualisierungskonzept hat eine Schwäche in der gleichzeitigen Darstellung zu vieler Datenpunkte. Das tritt dann auf, wenn zu viele Textpassagen verschlagwortet wurden. Am Beispiel der Auswertung für das Label #Beitel soll das verdeutlicht werden (vgl. Abbildung 52 rechts unten). Die negativen Ausreißer sind klar zu erkennen. Der restliche Verlauf ist durch die Datenmenge jedoch nicht mehr klar zu identifizieren. Die vollständige graphische Auswertung ist im Anhang (vgl. Anhang A.5 - Abbildung 56) zu finden.



Abbildung 52: Auswertung der Stimmungs- und Subjektivitätsverteilung für verschiedene Label

Am Ende des Auswertungsbereichs der Tools schließt sich der *Textdetail-Bereich* an. Dieser Bereich stellt alle Label, die zugeordneten Textpassagen und die errechneten Werte für Stimmung und Subjektivität dar. Diese Zahlenwerte sind zwar den Textpassagen zugeordnet, sind aber auf Basis der

englischen Übersetzung (welche aus Übersichtlichkeitsgründen nicht erneut separat aufgeführt werden) berechnet worden. Exemplarisch wird an dieser Stelle das Label #Aufkleber beschrieben (vgl. Abbildung 53). Bei der zweiten Textpassage zeigen sich Schwachstellen des aktuellen Algorithmus. Für den menschlichen Verstand zeigt die Textpassage „Abzüge in der Verpackung [...] nicht rückstandsfrei entfernen [...]“ eine sehr stark negative Aussage. Obschon die Bewertung der Subjektivität (Wert = 0,5) als sinnvoll betrachtet werden kann, stellt die Bewertung als eine „neutrale“ Aussage eine fragwürdige Einteilung dar. Diese Erkenntnisse werden im nächsten Kapitel diskutiert.

The image shows a screenshot of a product page for 'ERIGMA' with a callout box highlighting a specific label and its evaluation. The callout box contains the following text:

#Aufkleber

Einziges Manko wie immer: dieser hässliche Aufkleber auf dem Holzkasten, der sich nur schwer entfernen lässt.
Polarity: -0.3 | Subjektivität: 0.75 |

Abzüge gibt es für die Verpackung: die Holzbox ist mit einem Aufkleber versehen, der sich nicht rückstandsfrei entfernen lässt - vom Hersteller (!).
Polarity: 0.0 | Subjektivität: 0.5 |

The screenshot also shows other labels like #Spiegelkante and #Spiegelglas, but they are not highlighted.

Abbildung 53: Auswertung - Textdetail-Bereich für die vergebenen Label

8 Ergebnisse und Diskussionen

Das nachfolgende Kapitel gliedert sich entsprechend der entwickelten Teilziele in folgende Unterkapitel: 8.1 Aufbereitung von nutzergenerierten Inhalten für die Produktentwicklung (für Teilziel 1), 8.2 Algorithmus-basierte Auswertung gelabelter Textdaten (für Teilziel 2) und 8.3 Praxisnahe Implementierung und praktische Implikationen (für die Teilziele 3-5). Für jedes Unterkapitel werden die Motivation - die hergeleitete Problemstellung - kurz umrissen und die erreichten Ergebnisse dargelegt. Anschließend werden die Ergebnisse in Bezug auf die ausgewertete Literatur im Stand der Technik interpretiert und bewertet. Die Auswirkungen auf die bestehende Literatur und Praxis werden abgeschätzt. Für jedes Unterkapitel werden ausführlich die Beschränkungen der Untersuchungen, die Limitationen und mögliche Verbesserungen skizziert.

8.1 Aufbereitung von nutzergenerierten Inhalten für die Produktentwicklung

Nutzergenerierte Inhalt stellen eine wichtige Informationsquelle für die Produktentwicklung dar. Die aktuelle Wissenschaft diskutiert intensiv und kontrovers über die Vorteile und Möglichkeiten nutzergenerierter Inhalte. Es besteht Einigkeit darüber, dass die Bewertung von nutzergenerierten Inhalten aufgrund der zunehmenden Menge an Informationen im Internet immer wichtiger und komplexer wird.

Für die Analyse von Textdaten werden entweder die Texte automatisiert klassifiziert oder händisch vorbereitet. Für die händische Aufbereitung werden oftmals fachfremde Personen ohne direkten Anwendungsbezug beauftragt. Die Textaufbereitung bestimmt sehr stark die Analyse. Um den Ingenieur:innen in der Produktentwicklung zu befähigen, nutzergenerierte Inhalte für ihren Anwendungsfall aufzubereiten, fehlen in der aktuellen Literatur methodische Vorgehensweisen und praxisnahe Softwarewerkzeuge.

Sowohl die in der vorliegenden Arbeit beschriebene Vorgehensweise als auch die Validierung am Datensatz konnten zeigen, dass die entwickelte Methode für Ingenieur:innen in der Produktentwicklung zur Beschriftung (labeln) von nutzergenerierten Textdaten funktioniert. Durch die Vorgehensweise werden die wichtigen Informationen des Produktdatenmodells schon zu Beginn fast vollständig identifiziert. Dadurch können zeitaufwendige Iterationen und das Risiko von unvollständigen oder nicht berücksichtigten Informationen reduziert werden. Darüber hinaus ist zu erwähnen, dass diese Methode im Bereich der NLP-Literatur neu ist. Sie bietet auch für die bestehenden Ansätze der automatisierten Kategorisierung einen Vorteil. Die hier entwickelte Methode liefert ein Rahmenwerk, welches Pro-

duktdatenmodelle rechnerverarbeitend (nach festen Regeln) extrahiert. Somit können Produktkategorien automatisiert für Produkte synthetisiert werden. Diese Produktgruppen können dann als Cluster verwendet werden, um Vorhersagen und Analysen von Textdaten kontextbezogener zu gestalten.

Die entwickelte Methode zur Beschriftung von unstrukturierten Textdaten ist einfach durchzuführen. Zu Beginn wurde die Komplexität des Beschriftungsvorgangs unterschätzt. Erst nach mehrmaligen Iterationen mit Testpersonen konnte die Komplexität der Methode analysiert, dann reduziert und anschließend bedarfsgerecht entwickelt werden. Für den untersuchten Datensatz lieferte die Methode sehr gute Ergebnisse, da alle Textpassagen in das Kategorisierungsschema passten und weil es möglich war, die geforderten Merkmalsattribute des Produktes zu identifizieren. Eine Einschränkung ist die Wahl der Fallstudie. Es muss berücksichtigt werden, dass die Aussage eine Verallgemeinerung der Wirksamkeit der Methode über diese Produktgruppe und den Datensatz (Produkt) hinaus nicht zulässt. Um die Wirksamkeit der entwickelten Methode zusätzlich noch quantitativ zu bewerten, wäre es sinnvoll, die entwickelten Kennzahlen der Methode in einer separaten Fallstudie mit den Kennzahlen zu vergleichen, die für einen beschrifteten Datensatz ermittelt wurden, der ohne die Methode erstellt wurde. Zusätzlich sollten Mehrfallstudien (multi case study) die Methode in Bezug auf Produktgruppen und Ursprünge der Textinformationen weiter analysieren und von anderen bestehenden Ansätzen abgrenzen.

Auch wenn die Iterationen durch die Vollständigkeit der berücksichtigten Informationen wegfallen, stellt der Ressourcenaufwand den größten Kritikpunkt der entwickelten Methode dar. Diese Schwachstelle muss durch bessere Algorithmen oder einfachere Anpassung bestehender Algorithmen optimiert werden. Obwohl der Initialaufwand hoch ist, ist die anschließende Pflege und das Hinzufügen neuer Rezensionen einfach.

8.2 Algorithmus-basierte Auswertung gelabelter Textdaten

Die Auswertung nutzergenerierter Inhalte für die Produktentwicklung mithilfe von NLP-Algorithmen ist sehr komplex. Viele Wissenschaftler beschäftigen sich damit, Kundenwahrnehmungen explizit zu machen und so auszuwerten, dass auf dieser Basis Entscheidungen für das Produktdesign, die Produktfunktionen, die Alleinstellungsmerkmale und vieles mehr abgeleitet werden können (Cantamessa et al. 2020; Franke und Schreier 2010; Gefen und Ridings 2002; Xiong et al. 2021). Die größte Herausforderung besteht aktuell in der automatisierten Kategorisierung von Texten, in dem erheblichen Aufwand in der Initialisierung der Software und in der Anpassung des Softwarecodes. Diese Anpassung ist auf andere Domänen schlecht bis gar nicht übertragbar, wodurch die Barrieren für Ingenieur:innen, in Praxis und Wissenschaft, bei der Nutzung von Algorithmus-basierten Auswertungen von Textdaten zu hoch sind.

Die Berechnungen der Stimmungs- und Subjektivitätswerte durch die, in dieser Arbeit verwendete, Software lieferten sehr brauchbare Ergebnisse. Es konnte gezeigt werden, dass die entwickelte Logik

zur Auswertung gelabelter Textdaten durch Kombination bestehender quantitativer und qualitativer linguistischer Algorithmen, anwendbar und nutzbringend ist. Die in der vorliegenden Arbeit entwickelte Logik stellt eine Softwarebasis für die Auswertung unstrukturierter Texte im Rahmen der Produktentwicklung dar. Die Softwarebasis ist leicht anpassbar und kann immer wieder um neue und bessere Algorithmen (z.B. NLP-Algorithmen mit neuronalen Netzen) und um anschließende Auswertungen erweitert werden. Besonders kritisch ist die vorgeschlagene Art der Übersetzung. Der Umweg, deutsche Rezensionen ins Englische zu übersetzen, erscheint auf den ersten Blick sehr umständlich, ist aber unter Berücksichtigung der aktuellen Entwicklungstrends von Software ein sehr guter Weg, um auch in Zukunft die leistungsfähigsten Softwarecodes nutzen zu können. Die Übersetzung mittels Google-API stellt hierbei die größte Schwachstelle dar, da die Qualität der Übersetzung das Ergebnis der Analyse stark beeinflusst. Der leistungsstärkste gegenwärtig zur Verfügung stehende Algorithmus in diesem Bereich ist die Software Deepl.com. Für eine kommerzielle Nutzung des hier entwickelten Tools, sollte hierfür eine professionelle API eingerichtet werden. Die dafür notwendigen Voraussetzungen sind in der aktuellen Umsetzung des Tools bereits berücksichtigt und können einfach durch Änderungen im Softwarecode erfüllt werden.

Im Besonderen ist hervorzuheben, dass die Analyse des zeitlichen Verlaufs in Bezug auf die Stimulationswerte der Label ein Novum in der NLP-Literatur darstellt. Kritisch zu betrachten ist, dass die Null-Werte und die starken Schwankungen in den Wertepaaren eine Aussage oder Interpretation des Graphen oftmals erschweren. Eine bessere Darstellung könnte durch zusätzliche mathematische Operationen (Graphen glätten, Ausreißer aussortieren etc.) erreicht werden.

Eine weitere sehr vielversprechende Analyse, die in dieser Arbeit nicht weiter untersucht wurde, stellt der Vergleich der Stimmungswerte der gelabelten Textpassagen in Bezug zu der dazugehörigen Gesamtrezension dar. Dieser Vergleich ermöglicht die Aussagen über die Wahrnehmung kundenspezifischer zu berechnen. Ein ähnlicher Ansatz wurde von Lim und Tucker (2017) vorgeschlagen. Dabei wurden die Rezensenten anhand ihrer gemachten Rezensionen, insbesondere der vergebenen Sterne, kategorisiert. Aufgrund dieser Kategorisierung war es dann möglich die unterschiedlichen Rezensionen zu normieren und somit ein besser vergleichbares Analyseergebnis zu erzielen.

8.3 Praxisnahe Implementierung und praktische Implikationen

Die Validierung des Tools hat gezeigt, dass sowohl das Tool, als auch das verwendete Framework fähig sind, die definierten Anforderungen effektiv und effizient zu erfüllen. Die Implementierung, und somit die Möglichkeit die Software-Algorithmen praxisnah zur Verfügung zu stellen, war ein sehr aufwendiger und durch mehrere Iterationen geprägter Prozess. Der gewählte Lösungsansatz und die daraus resultierende Softwarelösung spiegeln die Anforderungen an die praktische Arbeit im Unternehmen wider und geben den Ingenieur:innen ein modernes und innovatives Werkzeug an die Hand.

Es bleibt die Frage, ob es für software-affine Ingenieur:innen möglich ist, auf Basis der hier beschriebenen Informationen das Tool nachzubauen und zu erweitern. Der Softwarecode im Anhang ist hierfür die dienlichste Informationsquelle. Eine Verbesserung in Form eines Entwickler-Baukastens anzustreben, würde die Flexibilität der Softwareentwickler:innen zu sehr einschränken. Ziel sollte es sein, die Einstiegshürden in die Softwareentwicklung für Ingenieur:innen durch eine verbesserte Dokumentation zu senken und die Flexibilität nicht einzuschränken.

Weiterhin konnte durch die Analyse der Häufigkeitsverteilungen der Label eine Gewichtung abgeleitet werden. Die Annahme, dass die Häufigkeit der Erwähnung mit der Wichtigkeit des Themas korreliert, wird auch in den meisten wissenschaftlichen Publikationen vertreten. Diese Auswertung ermöglicht es, Kundenprofile zu entwickeln und sie miteinander zu vergleichen. Die größte Schwäche der statistischen Textanalyse ist, dass Informationen, die selten erwähnt werden, kaum berücksichtigt werden. Der tf-IDF-Wert (vgl. Kapitel 3.1.2) ist geeignet, diese Schwäche zu verbessern, da damit die Wörter identifiziert werden, die in einem Textkorpus weniger häufig vorkommen.

Der Transfer und die Umsetzung in die Praxis sowie die auf Basis der Analyse abgeleiteten Handlungsempfehlungen wurden stichprobenartig mit dem verantwortlichen Mitarbeiter der Firma der Stechbeitel diskutiert und konnten bestätigt werden. Wie das Tool in die Praxis integriert wird und welche Maßnahmen für die Produktentwicklung abgeleitet werden, wurde nicht im Detail besprochen und muss in späteren Arbeiten weiter untersucht werden.

Aktuelle Trends der NLP-Techniken basieren auf „guten“, vorbereiteten Datensätzen, deren Funktion darin besteht, die mathematischen Modelle für einen gewissen Anwendungsfall zu instanzieren. Die Herausforderung besteht dabei immer darin, die Trainingsdatensätze aufzubereiten. Das hier entwickelte Tool ist eine sehr nutzerfreundliche und einfache Lösung, um Datensätze für den eCommerce Bereich aufzubereiten. Der Autor der vorliegenden Arbeit geht davon aus, dass die von ihm vorgestellte Umsetzung des Label-Vorgangs eine Innovation in diesem Wirtschaftsbereich darstellt, da auf diese Weise den Ingenieur:innen erstmals möglich wird, ihre Bedürfnisse und Perspektiven auf reale Datensätze in Form von Labeln zu übertragen und NLP direkt für ihre Fragestellungen in der Produktentwicklung zu nutzen.

9 Fazit und Ausblick

Nutzerzentrierte Produktentwicklung ist für Firmen einerseits eine Notwendigkeit und andererseits eine Möglichkeit wettbewerbsfähige und innovative Produkte zu entwickeln. Die aktuelle Schwierigkeit besteht darin, dass die nutzerzentrierte Produktentwicklung unter den modernen Rahmenbedingungen der Märkte und des Arbeitsalltags von Ingenieur:innen und Konstrukteur:innen aufgrund der großen Datenmengen ohne softwarebasierte Lösungen nicht mehr effizient und effektiv realisiert werden kann.

Die öffentlich zugänglichen Informationen auf sozialen Netzwerken, Foren oder Online-Handelsplattformen sind eine fast unerschöpfliche Quelle von Vorschlägen, Kritiken und persönlicher Meinungen. Kundenrezensionen auf Online-Handelsplattformen bieten aufgrund des direkten Produktbezugs und der Tatsache, dass der Verfasser Teil der Kundschaft ist, einen besonderen Nutzen für die Produktentwicklung im Vergleich zu anderen Informationsquellen. Die Rezensionen thematisieren Kundenwahrnehmungen, -bewertungen und -erfahrungen von Produkten. Die softwarebasierte Analyse dieser Textdaten bietet ein großes Potenzial für die Entdeckung von Mustern, Trends und versteckten Informationen über Kunden und Produkte, sowohl die eigenen als auch die fremd konkurrierend und noch nicht realisierte aber erwünschte Produkt. Diese Informationen werden beim *manuellen Durchlesen* der Kundenrezensionen kaum erkannt und können schwerlich systematisch ausgewertet und gespeichert werden.

Eine Auswertung von Kundenrezensionen ermöglicht ein besseres Verständnis der Kunden und schafft damit einen erheblichen Wettbewerbsvorteil. Gleichzeitig besteht aber auch die Gefahr, dass andere (z.B. Konkurrenten) die Inhalte von Kundenrezensionen nutzen, um neue Produkte zu entwickeln, die den Bedürfnissen der Kunden und Nutzenden besser entsprechen. Daher ist es schon heute notwendig, Lösungen wie die in dieser Arbeit entwickelten zu nutzen, um Produkte in Zukunft erfolgreich zu vermarkten. Ziel dieser Arbeit ist es, ein Text-Mining-Modell zur Analyse von unstrukturierten und nutzergenerierten Textdaten für die Produktentwicklung zu entwickeln. Das Text-Mining-Modell dient der Entwicklung innovativer, kundenzentrierter Produkte und sichert damit die Wettbewerbsfähigkeit von Unternehmen.

Die Forschungsarbeit untersuchte wie nutzergenerierte Inhalte für die Produktentwicklung aufbereitet werden können. Aus einer vergleichenden Analyse ausgewählter Primärstudien der internationalen Ingenieurliteratur wurden drei relevante Themenbereiche abgeleitet: (1) Ingenieurwissenschaftliche Betrachtungsweise, (2) Sprachbasierte Auswertung und (3) Nutzung und Implementierung.

Die größten Schwachstellen der aktuellen Forschungsarbeiten in den drei Themenbereichen sind: (1) fehlende methodische Unterstützung für Ingenieur:innen bei der Aufbereitung unstrukturierter Text-

daten, (2) kein gesamtheitliches Konzept (fehlende Logik und keine anforderungsgerechte Visualisierung) zur Auswertung aufbereiteter Textdaten und (3) keine öffentlich zugängliche Umsetzung der bestehenden Modelle.

Deshalb wurde in vorliegender Arbeit die drei Themenbereiche adressiert:

1. eine Methode zur Aufbereitung nutzergenerierte Texte entwickelt,
2. eine Logik zur Auswertung gelabelter Textdaten hergeleitet und
3. ein kontextspezifisches und anforderungsgerechtes Visualisierungskonzept erarbeitet.

Diese einzelnen Elemente wurden dann zu einem Gesamtmodell zusammengeführt. Dieses Gesamtmodell wurde mit Hilfe eines webbasierten Frameworks (Django) in eine Webapplikation umgesetzt und anhand eines realen Datensatzes validiert.

Aus den Ergebnissen lässt sich schließen, dass sowohl das Tool, die Methode als auch die Logik fähig sind, nutzergenerierte Inhalte für die Produktentwicklung sinnvoll und nutzbringend aufzubereiten. Das Fallbeispiel konnte zeigen, dass die Auswertung der einzelnen Textabschnitte einen erheblich besser strukturierteren Überblick über die Kundenperspektiven und Wahrnehmungen ermöglicht. Allerdings ergaben die verwendeten Algorithmen oftmals noch optimierungsbedürftige Ergebnisse. Weiterhin zeigte das Fallbeispiel, dass sowohl die Methode der Aufbereitung den verwendeten Datensatz ganzheitlich zu kategorisieren vermag, als auch die Implementierung eine Möglichkeit darstellt, gelabelte Datensätze als Trainingsdatensätze für ingenieurgetriebene Fragestellungen zu erstellen.

Implikationen

Für das *Forschungsfeld* stellt die Entwicklung des Tools und die Methode erstmalig ein gesamtheitliches Rahmenwerk für die Textdatenauswertung mithilfe von NLP für die Produktentwicklung dar. Die Kombination und die Zusammenführung der einzelnen Teilschritte ist nach Wissensstand des Autors neu und soll den gegenwärtigen wissenschaftlichen Diskurs erweitern helfen.

Für die *Produktentwicklung* stellt diese Arbeit eine sehr individualisierte Software und Methodenbasis zur Verfügung um unstrukturierte Textdaten aufzubereiten. Die Implementierung ist auf Kundenrezensionen fokussiert und kann leicht auf andere Datenstrukturen angepasst werden.

Für die *Softwareentwicklung* stellt diese Arbeit einen Vorschlag bereit, auf einfache Weise verbesserte NLP-Algorithmen anhand ingenieurwissenschaftlicher Fragestellungen zu validieren. Weiterhin sind „gute“ gelabelte Datensätze notwendig um maschinelle Lernalgorithmen adäquat trainieren zu können. Das entwickelte Tool und die Methode zur Datenaufbereitung ermöglicht es, domänenspezifische gelabelte Datensätze zu erstellen. Diese aufbereiteten Datensätze können dann genutzt werden, um maschinelle Lernalgorithmen zu trainieren.

Ausblick

Aus heutiger Sicht ist ein Ende des Trends zu immer mehr verfügbaren Daten und deren Nutzung nicht in Sicht. Zusätzlich wird für die Unternehmen, die noch Produkte zugleich entwickeln und vertreiben, der Konkurrenzdruck durch eine wachsende wirtschaftliche Globalisierung noch größer. Um diesen Herausforderungen in der Zukunft weiterhin gewachsen zu sein, ist eine datenbasierte Entscheidungslogik in der nutzerzentrierten Produktentwicklung ein vielversprechender Lösungsweg. Weiterhin sollte dieser Ansatz weitere zukunftssträchtige und übergeordnete Ziele (z.B. Nachhaltigkeit) verstärkt fokussieren. Beispielsweise wäre es denkbar, anhand der nutzergenerierten Textdaten die Kundenakzeptanz und –wahrnehmung für nachhaltigere Produktmerkmale (z.B. Verpackung, Reparatur, Wartung etc.) zu extrahieren, Produkte zu vergleichen oder Alleinstellungsmerkmale (Unique Selling Point, USP) für Produkte zu konfigurieren oder gänzlich neu zu entwickeln. Aktuelle Cluster-Algorithmen haben das Potential diese Aufgaben zu lösen. Derzeit bieten sie jedoch noch keine praxistauglichen Ergebnisse, da die dafür notwendige Datengrundlage aufbereiteter Daten fehlt. Ein aufbereiteter Datensatz (Textkorpus) für Produktbewertungen, der anschließend als Wissensbasis dient, ist ein vielversprechender Lösungsansatz. Dieser Datensatz lässt sich auch mithilfe vorbereiteter Funktionen in die Python-Bibliothek (NLTK's Plain textCorpusReader) einbinden. Die entwickelte Webapplikation bietet die Möglichkeit aufbereitete Daten zu generieren, um damit State-of-the-Art Machine Learning Algorithmen zu trainieren.

Der Autor ist daher der Meinung, dass das hier entwickelte Tool das Kundenverständnis in der Produktentwicklung maßgeblich verbessert und somit das Wissensmanagement im Unternehmen nachhaltig stärkt. Mit der zunehmenden Verfügbarkeit von Informationen gewinnt das Wissensmanagement immer mehr an Bedeutung. Durch eine datenbasierte und skalierbare Lösung ist das hier vorgestellte Tool in der Lage, auch zukünftig große Datenmengen auszuwerten und damit einen erheblichen Informationszuwachs zu liefern. Mit diesen neuen Informationen können Ingenieur:innen, Konstrukteur:innen, Manager:innen und andere an der Produktentwicklung Beteiligte verbesserte und innovativere Produkte zielgerichteter und ressourcenschonender entwickeln.

Literaturverzeichnis

- Agile manifesto (2001). Online verfügbar unter <https://de-academic.com/dic.nsf/dewiki/37363>.
- Alavi, Maryam; Leidner, Dorothy E. (2001): Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. In: *MIS QUARTERLY* 25 (1), S. 107. DOI: 10.2307/3250961.
- Aljumah, Ahmad Ibrahim; Nuseir, Mohammed T.; Alam, Md Mahmudul (2021): Traditional marketing analytics, big data analytics and big data system quality and the success of new product development. In: *BUSINESS PROCESS MANAGEMENT JOURNAL* 27 (4, SI), S. 1108–1125. DOI: 10.1108/BPMJ-11-2020-0527.
- Bartl, Michael; Füller, Johann; Mühlbacher, Hans; Ernst, Holger (2012): A Manager's Perspective on Virtual Customer Integration for New Product Development. In: *J Prod Innov Manag* 29 (6), S. 1031–1046. DOI: 10.1111/j.1540-5885.2012.00946.x.
- Bender, Beate; Gericke, Kilian (Hg.) (2021): Pahl/Beitz Konstruktionslehre. Methoden und Anwendung erfolgreicher Produktentwicklung. 9. Auflage. Berlin, Heidelberg: Springer Berlin Heidelberg (Springer eBook Collection).
- Bird, Steven; Klein, Ewan; Loper, Edward (2009): Natural Language Processing with Python. Analyzing Text with the Natural Language Toolkit. Sebastopol: O'Reilly Media Inc (Safari Books Online).
- Bonta, Venkateswarlu; Kumaresh, Nandhini; Janardhan, N. (2019): A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis. In: *AJCSST* 8 (S2), S. 1–6. DOI: 10.51983/ajcst-2019.8.S2.2037.
- Camastra, Francesco; Vinciarelli, Alessandro (2015): Machine learning for audio, image and video analysis. Theory and applications. 2nd edition. London, Heidelberg, New York, Dordrecht: Springer (Advanced information and knowledge processing).
- Cambria, Erik; White, Bebo (2014): Jumping NLP Curves: A Review of Natural Language Processing Research [Review Article]. In: *IEEE Comput. Intell. Mag.* 9 (2), S. 48–57. DOI: 10.1109/MCI.2014.2307227.
- Cantamessa, Marco; Montagna, Francesca; Altavilla, Stefania; Casagrande-Seretti, Alessandro (2020): Data-driven design: the new challenges of digitalization on product design and development. In: *Des. Sci.* 6. DOI: 10.1017/dsj.2020.25.

- Chen, Min; Mao, Shiwen; Liu, Yunhao (2014): Big Data: A Survey. In: *Mobile Netw Appl* 19 (2), S. 171–209. DOI: 10.1007/s11036-013-0489-0.
- Django Software Foundation (2021): Django Documentation (Release 3.2.6.dev). Online verfügbar unter <https://buildmedia.readthedocs.org/media/pdf/django/3.2.x/django.pdf>.
- Dybå, Tore; Dingsøy, Torgeir (2008): Empirical studies of agile software development: A systematic review. In: *Information and Software Technology* 50 (9-10), S. 833–859. DOI: 10.1016/j.infsof.2008.01.006.
- El Dehaibi, Nasreddine; Goodman, Noah D.; MacDonald, Erin F. (2019): Extracting Customer Perceptions of Product Sustainability From Online Reviews. In: *J. Mech. Des* 141 (12, SI). DOI: 10.1115/1.4044522.
- DIN EN ISO 9241-210, 23.09.2021: Ergonomie der Mensch-System-Interaktion - Teil 210. Online verfügbar unter <https://www.beuth.de/de/norm/din-en-iso-9241-210/135399380>, zuletzt geprüft am 23.09.2021.
- Feldhusen, Jörg; Grote, Karl-Heinrich (Hg.) (2013): Pahl/Beitz Konstruktionslehre. Methoden und Anwendung erfolgreicher Produktentwicklung. 8., vollständig überarbeitete Auflage. Berlin, Heidelberg: Springer Vieweg. Online verfügbar unter <http://gbv.ebib.com/patron/FullRecord.aspx?p=1538461>.
- Franke, Nikolaus; Schreier, Martin (2010): Why Customers Value Self-Designed Products: The Importance of Process Effort and Enjoyment *. In: *J Prod Innov Manag* 27 (7), S. 1020–1031. DOI: 10.1111/j.1540-5885.2010.00768.x.
- Gefen, David; Ridings, Catherine M. (2002): Implementation Team Responsiveness and User Evaluation of Customer Relationship Management: A Quasi-Experimental Design Study of Social Exchange Theory. In: *Journal of Management Information Systems* 19 (1), S. 47–69. DOI: 10.1080/07421222.2002.11045717.
- George, Nigel (2016): Mastering Django core. The complete guide to Django 1.8 LTS : delivers absolutely everything you will ever need to know to become a master Django programmer. Birmingham, UK: Packt Publishing.
- George, Nigel (2020): Build a Website With Django 3. 4th Edition: Gnw Independent Publishing.
- Göpfert, Jan (1998): Modulare Produktentwicklung. Zur gemeinsamen Gestaltung von Technik und Organisation. Wiesbaden: Deutscher Universitätsverlag (Springer eBook Collection Humanities, Social Science).
- Hou, Tianjun; Yannou, Bernard; Leroy, Yann; Poirson, Emilie (2019): Mining Changes in User Expectation Over Time From Online Reviews. In: *J. Mech. Des* 141 (9). DOI: 10.1115/1.4042793.

- HU BERLIN: STTS-Tags gemäß Tiger-Annotationsschema. Online verfügbar unter https://www.linguistik.hu-berlin.de/de/institut/professuren/korpuslinguistik/mitarbeiterinnen/hagen/STTS_Tagset_Tiger, zuletzt geprüft am 09.08.2021.
- Hukal, Philipp (2018): Three Essays on Growth and Innovation of Digital Platforms. Online verfügbar unter http://wrap.warwick.ac.uk/125585/1/WRAP_Theses_Hukal_2018.pdf, zuletzt geprüft am 20.09.2021.
- IEEE Computer Society (1983): IEEE Guide for Software Requirements Specifications. In: *IEEE Std 830-1984*. DOI: 10.1109/IEEESTD.1984.119205.
- Jiang, Huimin; Kwong, C. K.; Park, W. Y.; Yu, K. M. (2018): A multi-objective PSO approach of mining association rules for affective design based on online customer reviews. In: *Journal of Engineering Design* 29 (7, SI), S. 381–403. DOI: 10.1080/09544828.2018.1475629.
- Jiang, Huimin; Kwong, C. K.; Yung, K. L. (2017): Predicting Future Importance of Product Features Based on Online Customer Reviews. In: *J. Mech. Des* 139 (11, SI). DOI: 10.1115/1.4037348.
- Jin, Jian; Ji, Ping; Liu, Ying (2014): Prioritising engineering characteristics based on customer online reviews for quality function deployment. In: *Journal of Engineering Design* 25 (7-9), S. 303–324. DOI: 10.1080/09544828.2014.984665.
- Johnson, Jeff S.; Friend, Scott B.; Lee, Hannah S. (2017): Big Data Facilitation, Utilization, and Monetization: Exploring the 3Vs in a New Product Development Process. In: *J Prod Innov Manag* 34 (5), S. 640–658. DOI: 10.1111/jpim.12397.
- Jones, Karen Sparck (1994): Natural Language Processing: A Historical Review. In: Antonio Zampolli, Nicoletta Calzolari und Martha Palmer (Hg.): *Current Issues in Computational Linguistics: In Honour of Don Walker*. Dordrecht: Springer (Linguistica Computazionale, 9), S. 3–16.
- Jurafsky, Daniel; Martin, James H. (2020a): *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Draft. Online verfügbar unter https://web.stanford.edu/~jurafsky/slp3/ed3book_dec302020.pdf, zuletzt geprüft am 30.08.2021.
- Jurafsky, Daniel; Martin, James H. (2020b): *Word Senses and WordNet*. Online verfügbar unter <http://web.stanford.edu/~jurafsky/slp3/18.pdf>, zuletzt geprüft am 07.08.2021.
- Kim, Jongbae; Wilemon, David (2009): An empirical investigation of complexity and its management in new product development. In: *Technology Analysis & Strategic Management* 21 (4), S. 547–564. DOI: 10.1080/09537320902819296.

- Korgaonkar, Pradeep; Silverblatt, Ronnie; Girard, Tulay (2006): Online retailing, product classifications, and consumer preferences. In: *Internet Research* 16 (3), S. 267–288. DOI: 10.1108/10662240610673691.
- Larman, C.; Basili, V. R. (2003): Iterative and incremental developments. a brief history. In: *Computer* 36 (6), S. 47–56. DOI: 10.1109/MC.2003.1204375.
- Lim, Sunghoon; Tucker, Conrad S. (2017): Mitigating Online Product Rating Biases Through the Discovery of Optimistic, Pessimistic, and Realistic Reviewers. In: *J. Mech. Des* 139 (11, SI). DOI: 10.1115/1.4037612.
- Loria, Steven (2021): TextBlob: Simplified Text Processing — TextBlob 0.16.0 documentation. Online verfügbar unter <https://textblob.readthedocs.io/en/dev/index.html>, zuletzt aktualisiert am 29.01.2021, zuletzt geprüft am 08.08.2021.
- Madhavan, Ravindranath; Grover, Rajiv (1998): From Embedded Knowledge to Embodied Knowledge: New Product Development as Knowledge Management. In: *Journal of Marketing* 62 (4), S. 1. DOI: 10.2307/1252283.
- Malinen, Sanna (2015): Understanding user participation in online communities: A systematic literature review of empirical studies. In: *Computers in Human Behavior* 46, S. 228–238. DOI: 10.1016/j.chb.2015.01.004.
- Melé, Antonio (2020): Django 3 By Example. Build powerful and reliable Python web applications from scratch, 3rd Edition: Packt Publishing Ltd.
- Mendes-Filho, Luiz Augusto Machado; Tan, Felix B. (2008): An overview on User-Generated Content and the empowerment of Online Travellers. In: *FARN Journal* 7 (2), S. 17–30.
- Moisl, Hermann L.; Somers, H. L.; Dale, Robert (2000): Handbook of natural language processing. New York: Marcel Dekker.
- Naab, Teresa K.; Sehl, Annika (2017): Studies of user-generated content: A systematic review. In: *Journalism* 18 (10), S. 1256–1273. DOI: 10.1177/1464884916673557.
- Narula, Rajneesh (2004): R&D collaboration by SMEs: new opportunities and limitations in the face of globalisation. In: *Technovation* 24 (2), S. 153–161. DOI: 10.1016/S0166-4972(02)00045-7.
- O'Connor, Gina Colarelli; DeMartino, Richard (2006): Organizing for Radical Innovation: An Exploratory Study of the Structural Aspects of RI Management Systems in Large Established Firms. In: *J Prod Innov Manag* 23 (6), S. 475–497. DOI: 10.1111/j.1540-5885.2006.00219.x.
- Pahl, Gerhard; Beitz, Wolfgang (1977): Konstruktionslehre. Handbuch für Studium und Praxis. Berlin, Heidelberg, s.l.: Springer Berlin Heidelberg.

- Pahl, Gerhard; Beitz, Wolfgang; Feldhusen, Jörg; Grote, Karl-Heinrich (2007): Konstruktionslehre. Grundlagen erfolgreicher Produktentwicklung ; Methoden und Anwendung. 7. Aufl. Berlin, Heidelberg: Springer.
- Pavanelli Stefanovitz, Juliano; Lopes de Sousa Jabbour, Ana Beatriz (2021): Product development management complexity: emerging challenges and the role of senior leadership. In: *JKM ahead-of-print* (ahead-of-print). DOI: 10.1108/JKM-04-2021-0298.
- Ramesh, Balasubramaniam; Tiwana, Amrit (1999): Supporting Collaborative Process Knowledge Management in New Product Development Teams. In: *Decision support systems* 27 (1-2), S. 213–235. DOI: 10.1016/S0167-9236(99)00045-7.
- Schiller, Anne; Teufel, Simone; Stöckert, Christine (1999): Guidelines für das Tagging deutscher Textcorpora mit STTS. Kleines und großes Tagset. Hg. v. Universität Stuttgart - Institut für maschinelle Sprachverarbeitung. Online verfügbar unter <http://www.sfs.uni-tuebingen.de/resources/stts-1999.pdf>, zuletzt geprüft am 13.09.2021.
- Schwaber, Ken; Beedle, Mike (2002): Agile software development with Scrum. Online verfügbar unter http://sutlib2.sut.ac.th/sut_contents/h129174.pdf.
- Schwaber, Ken; Sutherland, Jeff (2011): The scrum guide. Online verfügbar unter <https://key2agile.de/wp-content/uploads/2020/11/2020-scrum-guide-german.pdf>.
- Schweitzer, Fiona; van den Hende, Ellis A.; Hultink, Erik-Jan (2020): There's More Than One Perspective to Take Into Account for Successful Customer Integration Into Radical New Product Innovation: A Framework and Research Agenda. In: *IEEE Transactions on Engineering Management* 67 (3), S. 813–829. DOI: 10.1109/TEM.2019.2893060.
- DIN EN ISO 12100, 23.09.2021: Sicherheit von Maschinen. Online verfügbar unter <https://www.beuth.de/de/norm/din-en-iso-12100/128264334>, zuletzt geprüft am 23.09.2021.
- Song, Wenyan; Ming, Xinguo; Xu, Zhitao (2013): Risk evaluation of customer integration in new product development under uncertainty. In: *Computers & Industrial Engineering* 65 (3), S. 402–412. DOI: 10.1016/j.cie.2013.04.001.
- Statista (2021a): Internet users in the world 2021 | Statista. Online verfügbar unter <https://www.statista.com/statistics/617136/digital-population-worldwide/>, zuletzt aktualisiert am 10.09.2021, zuletzt geprüft am 10.09.2021.
- Statista (2021b): Prognose zum Volumen der jährlich generierten digitalen Datenmenge weltweit in den Jahren 2018 und 2025 (in Zettabyte). Online verfügbar unter <https://de.statista.com/statistik/daten/studie/267974/umfrage/prognose-zum-weltweit-generierten-datenvolumen/>, zuletzt aktualisiert am 10.09.2021, zuletzt geprüft am 10.09.2021.

- Statista (2021c): Top 20 Webseiten in Deutschland nach der Anzahl der Unique Visitors im Juni 2021 (in Millionen). Online verfügbar unter <https://de.statista.com/statistik/daten/studie/180570/umfrage/meistbesuchte-websites-in-deutschland-nach-anzahl-der-besucher/>, zuletzt aktualisiert am 10.09.2021, zuletzt geprüft am 10.09.2021.
- Su, Chao-Ton; Chen, Yung-Hsin; Sha, D. Y. (2006): Linking innovative product development with customer knowledge: a data-mining approach. In: *Technovation* 26 (7), S. 784–795. DOI: 10.1016/j.technovation.2005.05.005.
- Suryadi, Dedy; Kim, Harrison (2018): A Systematic Methodology Based on Word Embedding for Identifying the Relation Between Online Customer Reviews and Sales Rank. In: *J. Mech. Des* 140 (12). DOI: 10.1115/1.4040913.
- Tan, Kim Hua; Zhan, Yuanzhu (2017): Improving new product development using big data: a case study of an electronics company. In: *R&D Management* 47 (4, SI), S. 570–582. DOI: 10.1111/radm.12242.
- Taylor, Ann; Marcus, Mitchell; Santorini, Beatrice (2003): The Penn Treebank: An Overview. In: Anne Abeillé (Hg.): *Treebanks. Building and Using Parsed Corpora*, Bd. 20. Dordrecht, Boston: Kluwer Academic Publishers (Text, Speech and Language Technology, 20), S. 5–22.
- Tik (2021): TIGER Korpus | Institut für Maschinelle Sprachverarbeitung | Universität Stuttgart. Online verfügbar unter <https://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger/#id-cef5dc25>, zuletzt aktualisiert am 09.08.2021, zuletzt geprüft am 09.08.2021.
- Ulrich, Karl (1995): The role of product architecture in the manufacturing firm. In: *Research Policy* 24 (3), S. 419–440. DOI: 10.1016/0048-7333(94)00775-3.
- Vajjala, Sowmya; Majumder, Bodhisattwa; Gupta, Anuj; Surana, Harshit (2020): Practical Natural Language Processing. Explore a preview version of Practical Natural Language Processing right now.: O'Reilly Media, Inc.
- Veryzer, Robert W.; Borja de Mozota, Brigitte (2005): The Impact of User-Oriented Design on New Product Development: An Examination of Fundamental Relationships*. In: *J Prod Innov Manag* 22 (2), S. 128–143. DOI: 10.1111/j.0737-6782.2005.00110.x.
- Vossen, Piek; Maks, Isa; Segers, Roxane; van der Vliet, Hennie; van Zutphen, Hetty (2007): The Cornetto database: architecture and alignment issues of combining lexical units, synsets and an ontology.
- Xiong, Xianqing; Lu, Guozhen; Lu, Danting (2021): Research on Children's Customized Furniture Design Based on Group Technology. In: *Applied Sciences* 11 (23), S. 11371. DOI: 10.3390/app112311371.

- Zhan, Yuanzhu; Tan, Kim Hua; Huo, Baofeng (2019): Bridging customer knowledge to innovative product development: a data mining approach. In: *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 57 (20), S. 6335–6350. DOI: 10.1080/00207543.2019.1566662.
- Zhan, Yuanzhu; Tan, Kim Hua; Li, Yina; Tse, Ying Kei (2018): Unlocking the power of big data in new product development. In: *ANNALS OF OPERATIONS RESEARCH* 270 (1-2, SI), S. 577–595. DOI: 10.1007/s10479-016-2379-x.

Anhang

A.1 Übersicht existierender Textkorpora

Tabelle 12: Übersicht verschiedener Textkorpora für die NLP-Bibliothek NLTK (Bird et al. 2009)

Korpus-Name	Compiler	Inhalte
Brown Corpus	Francis, Kucera	15 Genres, 1,15 Mio. Wörter, getaggt, kategorisiert
CESS Treebanks	CLIC-UB	1 Mio. Wörter, getaggt und geparst (Katalanisch, Spanisch)
Chat-80 Data Files	Pereira & Warren	Weltgeografische Datenbank
CMU Pronouncing Dictionary	CMU	127k Einträge
CoNLL2000 Chunking Data	CoNLL	270k Wörter, getaggt und geschunked
CoNLL2002 Named Entity	CoNLL	700k Wörter, verschlagwortet und mit Namen versehen (Niederländisch, Spanisch)
CoNLL2007 Dependency Treebanks (sel)	CoNLL	150k Wörter, abhängigkeitsgeparst (Baskisch, Katalanisch)
Dependency Treebank	Narad	Auf Abhängigkeiten geprüfte Version der Penn Treebank-Probe
FrameNet	Fillmore, Baker et al.	10k Wortsinne, 170k manuell annotierte Sätze
Floresta Treebank	Diana Santos et al.	9k Sätze, getaggt und geparst (Portugiesisch)
Gazetteer Lists	Various	Listen von Städten und Ländern
Genesis Corpus	Misc web sources	6 Texte, 200k Wörter, 6 Sprachen
Gutenberg (selections)	Hart, Newby, et al.	18 Texte, 2 Mio. Wörter
Inaugural Address Corpus	CSPAN	Antrittsreden von US-Präsidenten (1789 bis heute)
Indian POS-Tagged Corpus	Kumaran et al.	60k Wörter, getaggt (Bangla, Hindi, Marathi, Teluga)
MacMorpho Corpus	NILC, USP, Brazil	1 Mio. Wörter, getaggt (brasilianisches Portugiesisch)
Movie Reviews	Pang, Lee	2k Filmbewertungen mit Klassifizierung der Stimmungspolarität
Names Corpus	Kantowitz, Ross	8k männliche und weibliche Namen
NIST 1999 Info Extr (selections)	Garofolo	63k Wörter, Newswire und SGML-Markup für benannte Einheiten (named-entity)
Nombank	Meyers	115k Propositionen, 1400 Substantivrahmen
NPS Chat Corpus	Forsyth, Martell	10k IM-Chat-Beiträge, POS-tagged und Dialog-Act-tagged
Open Multilingual WordNet	Bond et al.	15 Sprachen, an das englische WordNet angeglichen
PP Attachment Corpus	Ratnaparkhi	28k Präpositionalphrasen, getaggt als Substantiv- oder Verbmodifikatoren
Proposition Bank	Palmer	113k Propositionen, 3300 Verb-Frames
Question Classification	Li, Roth	6k Fragen, kategorisiert
Reuters Corpus	Reuters	1,3 Mio. Wörter, 10.000 Nachrichtendokumente, kategorisiert
Roget's Thesaurus	Project Gutenberg	200k Wörter, formatierter Text
RTE Textual Entailment	Dagan et al.	8k Satzpaare, kategorisiert
SEMCOR	Rus, Mihalcea	880k Wörter, Part-of-Speech und Sinnmarkierung
Senseval 2 Corpus	Pedersen	600k Wörter, Part-of-Speech und Sinnmarkierung
SentiWordNet	Esuli, Sebastiani	Stimmungsbewertungen für 145k WordNet-Synonymsätze
Shakespeare texts (selections)	Bosak	8 Bücher im XML-Format
State of the Union Corpus	CSPAN	485k Wörter, formatierter Text
Stopwords Corpus	Porter et al.	2.400 Stoppwörter für 11 Sprachen
Swadesh Corpus	Wiktionary	vergleichende Wortlisten in 24 Sprachen
Switchboard Corpus (selections)	LDC	36 Telefonanrufe, transkribiert, geparst
Univ Decl of Human Rights	United Nations	480k Wörter, 300+ Sprachen
Penn Treebank (selections)	LDC	40k Wörter, getaggt und geparst
TIMIT Corpus (selections)	NIST/LDC	Audiodateien und Transkripte für 16 Sprecher
VerbNet 2.1	Palmer et al.	5k Verben, hierarchisch organisiert, verknüpft mit WordNet
Wordlist Corpus	OpenOffice.org et al.	960k Wörter und 20k Affixe für 8 Sprachen
WordNet 3.0 (English)	Miller, Fellbaum	145k Synonymsätze

A.2 Nomenklatur zur Bezeichnung von Wortarten

pos-tag	Beschreibung	Beispiel(e)
ADJA	attributives Adjektiv	<i>der <u>schlaue</u>/ADJA Mitarbeiter</i>
ADJD	adverbiales ODER prädikatives Adjektiv	<i>er spricht <u>schnell</u>/ADJD</i> <i>Sein Sprechen ist <u>schnell</u>/ADJD</i>
ADV	Adverb	<i><u>Bald</u>/ADV <u>schon</u>/ADV kommt sie wohl/ADV</i>
APPR	Präposition; Zirkumposition links	<i><u>nach</u>/APPR Berlin; <u>ohne</u>/APPR Hund</i>
APPRART	Präposition mit Artikel	<i><u>zum</u>/APPRART Streichen; <u>zur</u>/APPRART Sache</i>
APPO	Postposition	<i>ihm <u>zuleibe</u>/APPO; der Sache <u>wegen</u>/APPO</i>
APZR	Zirkumposition rechts	<i>von mir <u>aus</u>/APZR</i>
ART	bestimmter ODER unbestimmter Artikel	<i><u>Der</u>/ART Mann schenkt <u>die</u>/ART Rose</i> <i><u>einer</u>/ART unerwarteten Frau</i>
CARD	Kardinalzahl	<i><u>zwei</u>/CARD Männer im Jahre 1994/CARD</i>
FM	Fremdsprachliches Material	<i>Er sagte: " <u>Hasta</u>/FM <u>luego</u>/FM, <u>amigos</u>/FM ."</i>
ITJ	Interjektion	<i><u>Mhm</u>/ITJ, <u>ach</u>/ITJ, <u>tja</u>/ITJ, dann halt nicht.</i>
KOUI	unterordnende Konjunktion mit (zu-)Infinitiv	<i>Sie kommt, <u>um</u>/KOUI zu arbeiten</i>
KOUS	unterordnende Konjunktion	<i><u>Anstatt</u>/KOUI anzufangen, geht sie wieder</i>
KON	nebenordnende Konjunktion und, oder, aber	<i>Emma wartet, <u>weil</u>/ob/so/longe/dass/KOUS sie stiehlt</i> <i>Sie <u>und/oder</u>/KON Emma kommen und/KON streichen</i>
KOKOM	Vergleichskonjunktion als, wie	<i>blauer <u>als</u>/KOKOM er; blau <u>wie</u>/KOKOM er</i>
NN	normales Nomen	<i>am <u>Tage</u>/NN dem <u>Mann</u>/NN den <u>Schlaf</u>/NN</i>
NE	Eigennamen	<i>die <u>Emma</u>/NE dem <u>Hans</u>/NE sein <u>HSV</u>/NE</i>
PDAT	attribuierendes Demonstrativpronomen	<i><u>Jene</u>/PDAT Männer sprachen <u>dieses</u>/PDAT lockere Spanisch</i>

Abbildung 54: STTS-Tags gemäß TIGER-Annotationsschemata (HU BERLIN)

A.3 Ablaufschema der Änderung eines Datensatzes

The image illustrates the workflow for updating a labeled text passage in the ENIGMA system. It is divided into three main sections:

- Review List:** A table of reviews with columns for ID, text, polarity, and subjectivity.

ID	Text	Polarity	Subjectivity
1596097402-66	Mein Fazit: Sie sind schon toll anzusehen, aber bei der Arbeit mit den Kirschen wird das ganze Potential dieser tollen Werkzeuge sichtbar.	0.4875	0.7625
1596097391-11	Gute Stemmeisen	0.7	0.6000000000000001
1596097391-11	Gute Beitel, am Anfang muss man noch einmal nachschleifen, aber insgesamt tolle Qualität, kann ich jedem nur empfehlen.	0.375	0.5875
1596097391-11	Abzüge gibt es für die Verpackung: die Holzbox ist mit einem Aufkleber versehen, der sich nicht rückstandsfrei entfernen lässt - vom Hersteller (!).	0.0	0.5
1596097393-21	Ich kenne keine besseren Beitel	0.5	0.5
- Analytics View:** A detailed view of a review with a text area and two buttons: 'Update' (highlighted) and 'Delete'.
- Create Labeled Review Form:** A form to create a new labeled review with the following fields:
 - Text*:** A large text area containing the review text.
 - Labels*:** A text input field containing the label '#FirmaKirschen#Beitel#Qualität'.
 - Review-Bezug*:** A dropdown menu showing the review ID '1596097402-66'.
 - Create Labeled Review:** A button to submit the form.

Abbildung 55: Änderungsprozedur für eine gelabelte Textpassage

A.4 Ausschnitte Software-Code

A.4.1 Django-Modelle

LabeledReview-Class im Blog-Model

```

class LabeledReview(models.Model):
    webScraperOrder = models.ForeignKey(Review, default=1,
    verbose_name="Review-Bezug", on_delete=models.SET_DEFAULT)
    text = models.TextField(null=True, max_length=8000) #Start, End
    labels = models.CharField(max_length=800, null=True,
    default="#wieckONE")

    date = models.DateField(verbose_name='DATE in Format (mm/dd/yyyy) oder
    (YYYY-MM-DD)', null=True)
    sentiment_polarity_DE = models.FloatField(default=-5)
    sentiment_subjectivity_DE = models.FloatField(default=-5)

    sentiment_polarity_related_review_DE = models.FloatField(default=-5)
    sentiment_subjectivity_related_review_DE = models.FloatField(default=-5)

    ###ENGLISH COUNTERPART
    text_eng = models.TextField(null=True, max_length=8000) #Start, End
    sentiment_polarity_ENG = models.FloatField(default=-5)
    sentiment_subjectivity_ENG = models.FloatField(default=-5)

    sentiment_polarity_related_review_ENG = models.FloatField(default=-5)
    sentiment_subjectivity_related_review_ENG = models.FloatField(default=-
5)

    def __str__(self):
        return '%s %s %s %s' %(self.webScraperOrder.web_scraper_order,
    self.text, self.labels, self.id)

```

Review-Model

```

class Review(models.Model):
    web_scraper_order = models.TextField(max_length=128, default='NULL')
    web_scraper_start_url = models.URLField(blank = True)
    author = models.CharField(max_length=128, null=True)
    title = models.TextField(blank = True)
    content = models.TextField(blank = True)
    rating = models.CharField(max_length=128, default='NULL')
    next_href = models.URLField(blank = True)
    date = models.TextField(blank = True)

    STATUS_CHOICES_FRANZ = (
        (1, 'neu'),
        (2, 'wip'),
        (3, 'done'),
    )
    status = models.IntegerField(choices=STATUS_CHOICES_FRANZ, default=1)

    def __str__(self):
        return self.web_scraper_order #stringformat:"s"

```

LabeledReview-Model:

```

class LabeledReview(models.Model):

    webScrapOrder = models.ForeignKey(Review, default=1,
verbose_name="Review-      Bezug", on_delete=models.SET_DEFAULT)

    text = models.TextField(null=True, max_length=8000) #Start, End
    labels = models.CharField(max_length=800, null=True,
default="#wieckONE")
    date = models.DateField(verbose_name='DATE in Format (mm/dd/yyyy) oder
(YYYY-MM-DD)', null=True)
    sentiment_polarity_DE = models.FloatField(default=-5)
    sentiment_subjectivity_DE = models.FloatField(default=-5)

    sentiment_polarity_related_review_DE = models.FloatField(default=-5)
    sentiment_subjectivity_related_review_DE = models.FloatField(default=-5)

    ###ENGLISH COUNTERPART
    text_eng = models.TextField(null=True, max_length=8000) #Start, End
    sentiment_polarity_ENG = models.FloatField(default=-5)
    sentiment_subjectivity_ENG = models.FloatField(default=-5)

    sentiment_polarity_related_review_ENG = models.FloatField(default=-5)
    sentiment_subjectivity_related_review_ENG = models.FloatField(default=-
5)

    def __str__(self):
        return '%s %s %s %s' %(self.webScrapOrder.web_scrap_order,
self.text,      self.labels, self.id)

```

GartenbaumAnalysis-Model

```

class GartenbaumAnalysis(models.Model):
    STATUS_CHOICES = (
        (1, 'imported'),
        (2, 'wip'),
        (3, 'done'),
    )
    webScrapOrder = models.ForeignKey(Review, default=1,
verbose_name="Review-Bezug", on_delete=models.SET_DEFAULT)
    status = models.IntegerField(choices=STATUS_CHOICES, default=1)
    time_range = models.CharField(max_length=200) #Zeitraum ~ Rezension
    review_length = models.CharField(max_length=5000) #Zeichenanzahl ~
Rezension

```

Label-Model

```

class Label(models.Model):
    label=models.CharField(max_length=200)
    color = models.CharField(max_length=120, default='primary')

    def __str__(self):
        return self.label

```

A.4.2 Django-Views*Import in views.py:*

```

from django.core.paginator import Paginator, PageNotAnInteger, EmptyPage
from django.db.models import Count
from django.forms import model_to_dict
from django.shortcuts import render, redirect
from django.http import HttpResponse, HttpResponseRedirect
from django.contrib.auth.mixins import LoginRequiredMixin,
UserPassesTestMixin
from rest_framework import authentication, permissions

from rest_framework.views import APIView
from rest_framework.response import Response

from .forms import *
from .models import *

from .forms import CreateProduktMasterForm, UploadcsvLabeledReviewDataForm
from .models import Review, ProduktMaster, csvLabeledReviewData, csvData

import csv, io
import sys, os
from django.contrib import messages
from django.contrib.auth.decorators import permission_required,
login_required
import pandas
import numpy as np

#####SENTIMENT
from textblob_de import TextBlobDE #as TextBlobDE
from textblob import TextBlob

#####TRANSLATE
import pydeepl #geht nicht
from googletrans import Translator

import datetime
from django.views.generic import ListView, DetailView, CreateView,
UpdateView, DeleteView

#####Globale Werte für die Grenzen
GRENZE_POLARITY_NEGATIV = -0.1
GRENZE_POLARITY_POSITIV = 0.01

```

Funktion `review_upload` in `views.py`:

```
@permission_required('admin:can_add_log_entry')
def review_upload(request):

    template = 'blog/review_upload.html'
    prompt = {
        'order' : "Order of the csv should be based on the requiremnt for
the webcrawler "
    }

    if 'upload_amazon_review_raw_csv' in request.POST:
    if not Review.objects.all():
        if request.method == "GET":
            return render(request,template,prompt)
        csv_file = request.FILES['file']
        if not csv_file.name.endswith('.csv'):
            messages.error(request,'This is not a .csv file')
        data_set = csv_file.read().decode('UTF-8')
        io_string_neu = io.StringIO(data_set)
        next(io_string_neu)
        dataset = pandas.read_csv(io_string_neu, encoding="UTF-8",
error_bad_lines=False, delimiter=',', engine='python', header=None)
        # Hier werden die Einträge für das Model aus der CSV extrahiert
        stopper = 0
        for row in dataset.itertuples():
            review_inter = Review(
                web_scraper_order=row[1],
                web_scraper_start_url=row[2],
                author=row[3],
                title=row[4],#.title,
                date=row[5],#.date,
                content=row[6],#.content,
                rating=row[7],#.rating,
                next_href=row[9]
            )
            review_inter.save()
            messages.success(request, 'CSV erfolgreich hochgeladen')
    else:
        print("datensatz bereits vorhanden")
        messages.warning(request, 'Datensatz bereits vorhanden')
    return redirect('/')
```

Software-Code der Funktion `home` in `views.py`:

```
def home(request):

    labeledReviews = LabeledReview.objects.all()
    labelForm = LabelForm()
    createForm = CreateLabeledReviewForm()
    reviewForm = ReviewForm()
    gartenbaumAnalysisForm = GartenbaumAnalysisForm()

    #reviews = Review.objects.all()
    reviews = Review.objects.filter(status = 1) # HIER NUR NOCH ALLE
    REZENSIONEN NUTZEN; DIE NICHT "ALS FERTIG MARKIERT SIND"

    #Darstellung nur eine Rezension pro Seite (Paginator)
    if not reviews.exists():
        print('nüscht zu paginatoren')
    else:
        paginator = Paginator(reviews, 1)
        page = request.GET.get('page')

        reviews = paginator.get_page(page)
        counts = {
            'reviews': Review.objects.all().count(),
            'completed_reviews' : (Review.objects.all().count()) -
(Review.objects.filter(status = 1).count()),
            'labels': Label.objects.all().count(),
            'labeled_review': LabeledReview.objects.all().count()
        }
        print('COUNT DER REVIEWS' + str(Review.objects.all().count()))
        print('COUNT DER LABELS' + str(Label.objects.all().count()))

    if request.method == 'POST':
        if 'btn_create_labeled_review' in request.POST:
            createLabeledText(request)
            return redirect (request.META['HTTP_REFERER'])
        elif 'Create_A_NEW_LABEL' in request.POST:
            createLabel(request)
            return redirect('./')
        elif 'update_review' in request.POST:
            updateReview(request)
            return

    context = {
        'reviews': reviews,
        'labels': Label.objects.all(),
        'labeledReviews' : labeledReviews,
        'labelform': labelForm,
        'createForm' : createForm,
        'counts': counts,
        'gartenbaumAnalysisForm': gartenbaumAnalysisForm,
        'reviewForm' : reviewForm,
    }
    return render(request, 'blog/home.html', context)
```

Software-Code der Funktion `createLabeledText` in `views.py`:

```
def createLabeledText(request):
    label_list = []
    createform = CreateLabeledReviewForm(request.POST)
    ##LABEL-LISTE NOCH HINZUFÜGEN#####
    current_Labels = Label.objects.all()
    for label in current_Labels:
        if str(label) in request.POST:
            label_list.append(str(label))
    label_string = ''.join(label_list)
    #####
    createform.labels = label_string # hinzufügen zur Form
    if createform.is_valid():
        try:
            current_review_object =
Review.objects.get(web_scraper_order=request.POST.get('getReviewID'))
            review_header = current_review_object.title #nimm den Header der
            review_content = current_review_object.content #nimm den Content
der
            typed_text_in_field = request.POST.get('text') #nimm die Eingabe
        except Exception:
            raise
            # Prüfen ob der zu speichernde Text Teil der Rezension ist
            if findString(typed_text_in_field, review_header, review_content,
review_object = current_review_object):
                #Wenn er den String gefunden hat, dann speichern
                instance = createform.save(commit=False) # Umweg, da Label erst
default gesetzt wird und dann aber angepasst (bevor gespeichert)
                instance.labels = label_string # Hier die Labels sauber
zuweisen
                #####Und hier noch die Referenz zu der Ausgangsrezension
                if 'getReviewID' in request.POST:
                    webScrapID = request.POST.get('getReviewID')
                    # webScrapID = '1591616466-277'
                    instance.webScrapOrder =
Review.objects.get(web_scraper_order=webScrapID)
                    #####englische Übersetzung gleich hinzufügen
                    instance.text_eng = myTranslate(typed_text_in_field)
                    instance.save()
                    messages.success(request, 'Text gespeichert!')
            else:
                messages.error(request, 'Oh, Oh, da ist wohl etwas schief
gelaufen... \n Markierter Text wurde nicht gespeichert.')
                return redirect('/hatta/')
            return

return
```

Funktion `deep_analytics` in `views.py` (Ausschnitt):

```

@login_required
def deep_analytics(request):
    [...]

    #####WIE OFT WURDE JEDES LABEL VERWENDET#####
    for label in labels:
        #gehe durch alle einträgen und such nach dem label
        count = LabeledReview.objects.filter(labels__contains=label).count()
        #list_entry = [(str(label),count)]
        label_count_list[label] = count
        unique_reviews_list = []

        if LabeledReview.objects.filter(labels__contains=label):
            for each in
LabeledReview.objects.filter(labels__contains=label):
                if each.sentiment_polarity_ENG > GRENZE_POLARITY_POSITIV:
                    sentiment_count_ENG_polarity_positiv += 1
                elif each.sentiment_polarity_ENG < GRENZE_POLARITY_NEGATIV:
                    sentiment_count_ENG_polarity_negativ += 1
                else:
                    sentiment_count_ENG_polarity_neutral += 1
                if each.sentiment_subjectivity_ENG > 0:
                    sentiment_count_ENG_subjectivity_positiv += 1
                else:
                    sentiment_count_ENG_subjectivity_neutral += 1
                if each.webScraperOrder not in unique_reviews_list:
                    unique_reviews_list.append(each.webScraperOrder)

            sentiment_label_count = {
                'Label': label,
                'all_count': int(sentiment_count_ENG_subjectivity_positiv +
sentiment_count_ENG_subjectivity_neutral),
                'unique_reviews': len(unique_reviews_list), #wenn mehrmal
pro review auftaucht - wird das nur einmal gezählt
                'polarity':{
                    "positiv": sentiment_count_ENG_polarity_positiv,
                    "negativ": sentiment_count_ENG_polarity_negativ,
                    "neutral": sentiment_count_ENG_polarity_neutral,
                    'time_list': 0, #[VALUE,TIME]
                },
                "subjectivity": {
                    "positiv": sentiment_count_ENG_subjectivity_positiv,
                    "neutral": sentiment_count_ENG_subjectivity_neutral
                }
            }
    [...]

    fakten = {
        'rezensionen_gesamt' : Review.objects.all().count(),
        'labels':labels,
        'labeled_texte_gesamt' : LabeledReview.objects.all().count(),
        'zahl': zahl,
        'zahl_zahl': zahl_zahl,
        'anzahl_textabschnitte_label':anzahl_textabschnitte_label,
        'durchschnitt_textabschnitt' : durchschnitt_textabschnitt,
        'label_count_list':label_count_list,
        'context_meta_data': context_meta_data,
        'list_year_count': context_meta_data['list_year_count'],
        'unique_list_year': context_meta_data['unique_list_year'],
        'sentiment_dict_count': sentiment_dict_count,
        'sentiment_label_count': sentiment_label_count,
        'sentiment_label_count_df': sentiment_label_count_df
    }

```

```

context={
  'reviews': Review.objects.all(),
  'LabeledReviews' : LabeledReview.objects.all(),
  'allLabels': allLabels,
  'fakten': fakten,
  'label_count_list_direct': label_count_list,
  'science_table': science_table,
  'statistic_kpi': science_context['statistic_kpi']
}
return render (request, 'blog/deep-analytics.html', context)

```

A.5 Label-Auswertung

Tabelle 13: Erweiterte Auswertung - Anzahl verwendeter Label und Häufigkeitsverteilung

		Stimmung			Subjektivität	
		Positiv	Neutral	Negativ	Subjektiv	Objektiv
#Beitel	Absolut [n=172]	110	51	11	133	39
	Relativ	64,0%	29,7%	6,4%	77,3%	22,7%
	Relativ zu allen	44,2%	20,5%	4,4%	53,4%	15,7%
#Phase_Verwenden	Absolut [n=79]	47	29	3	60	19
	Relativ	59,5%	36,7%	3,8%	75,9%	24,1%
	Relativ zu allen	18,9%	11,6%	1,2%	24,1%	7,6%
#FirmaKirschen	Absolut [n=48]	26	18	4	34	14
	Relativ	54,2%	37,5%	8,3%	70,8%	29,2%
	Relativ zu allen	10,4%	7,2%	1,6%	13,7%	5,6%
#WeitereProduktmerkmale	Absolut [n=47]	31	12	4	41	6
	Relativ	66,0%	25,5%	8,5%	87,2%	12,8%
	Relativ zu allen	12,4%	4,8%	1,6%	16,5%	2,4%
#Qualität	Absolut [n=39]	28	8	3	32	7
	Relativ	71,8%	20,5%	7,7%	82,1%	17,9%
	Relativ zu allen	11,2%	3,2%	1,2%	12,9%	2,8%
#Klinge	Absolut [n=23]	12	8	3	20	3
	Relativ	52,2%	34,8%	13,0%	87,0%	13,0%
	Relativ zu allen	4,8%	3,2%	1,2%	8,0%	1,2%
#Phase_InBetriebNehmen	Absolut [n=16]	9	7	0	11	5
	Relativ	56,3%	43,8%	0,0%	68,8%	31,3%
	Relativ zu allen	3,6%	2,8%	0,0%	4,4%	2,0%
#Phase_Warten	Absolut [n=13]	6	7	0	12	1
	Relativ	46,2%	53,8%	0,0%	92,3%	7,7%
	Relativ zu allen	2,4%	2,8%	0,0%	4,8%	0,4%
#Griff	Absolut [n=12]	8	2	2	12	0
	Relativ	66,7%	16,7%	16,7%	100,0%	0,0%
	Relativ zu allen	3,2%	0,8%	0,8%	4,8%	0,0%
#Kiste	Absolut [n=12]	8	2	2	12	0
	Relativ	66,7%	16,7%	16,7%	100,0%	0,0%
	Relativ zu allen	3,2%	0,8%	0,8%	4,8%	0,0%
#Spiegelseite	Absolut [n=4]	0	4	0	3	1
	Relativ	0,0%	100,0%	0,0%	75,0%	25,0%
	Relativ zu allen	0,0%	1,6%	0,0%	1,2%	0,4%
#Aufkleber	Absolut [n=2]	0	1	1	2	0
	Relativ	0,0%	50,0%	50,0%	100,0%	0,0%
	Relativ zu allen	0,0%	0,4%	0,4%	0,8%	0,0%
#Sales	Absolut [n=1]	1	0	0	1	19
	Relativ	100,0%	0,0%	0,0%	5,0%	95,0%
	Relativ zu allen	0,4%	0,0%	0,0%	0,4%	7,6%
#Intensivgebrauch	Absolut [n=1]	0	1	0	0	1
	Relativ	0,0%	100,0%	0,0%	0,0%	100,0%
	Relativ zu allen	0,0%	0,4%	0,0%	0,0%	0,4%

Graphische Auswertung der NLP-basierten Analyse

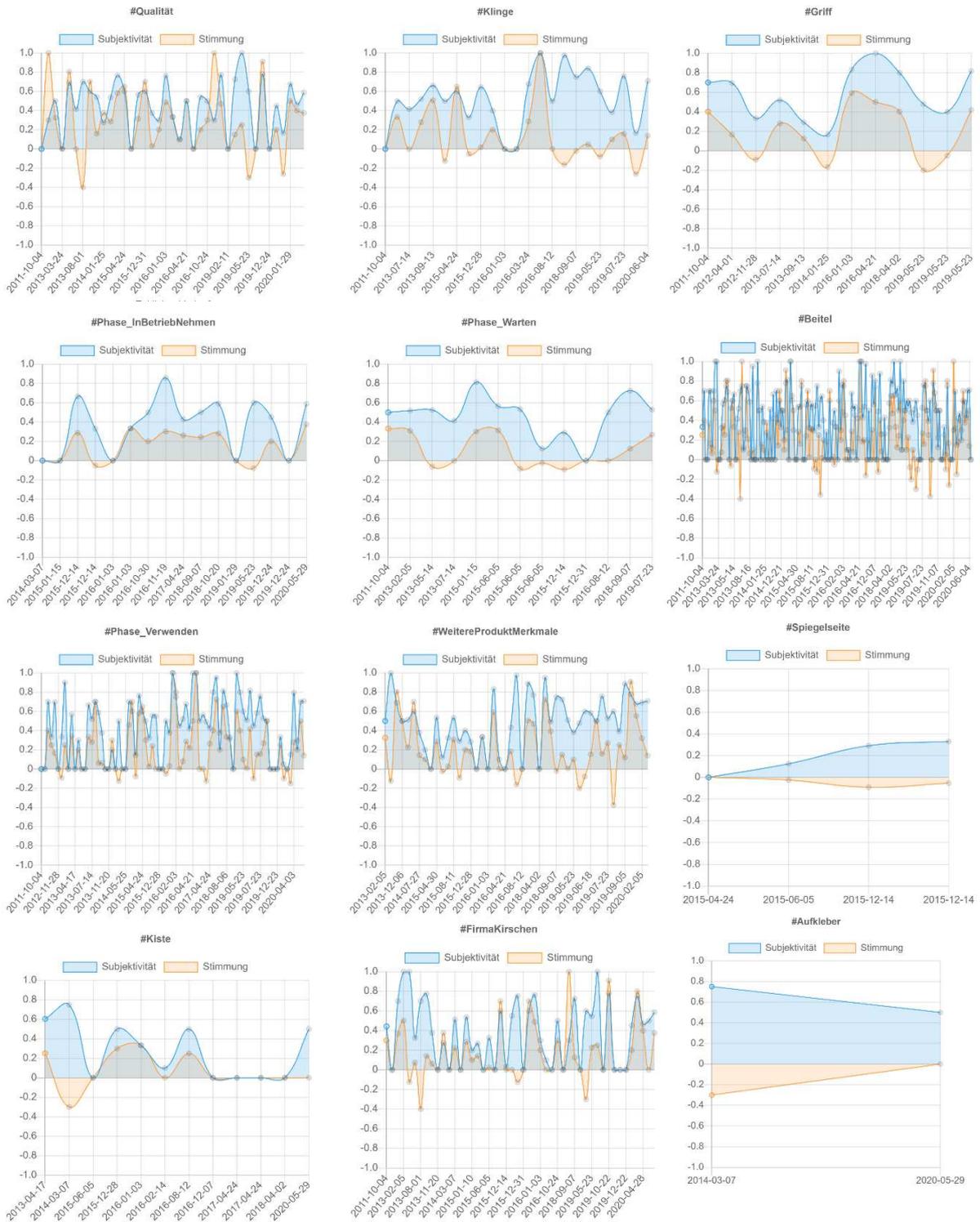


Abbildung 56: Auswertung der Stimmungs- und Subjektivitätsverteilung



**PRODUKT
SICHERHEIT
QUALITÄT**



**BERGISCHE
UNIVERSITÄT
WUPPERTAL**