Universität Bremen

DISSERTATION

zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

Optimierung des semantischen Informationszugriffs auf Technische Dokumentation

Jan Oevermann

vorgelegt dem

Fachbereich 3 Mathematik/Informatik
der Universität Bremen

GUTACHTER

Prof. Dr. Wolfgang Ziegler
Prof. Dr. Christoph Lüth

ZEITRAUM

Oktober 2015 - April 2019

Dissertation

Optimierung des semantischen Informationszugriffs
auf Technische Dokumentation

zur Veröffentlichung in: tekom Hochschulschriften (Bd. 25), tcworld: Stuttgart

Fassung vom 25.06.2019 10:42

Vorgelegt von
Jan Oevermann, M.Sc.
geboren am 13.12.1988 in Baden-Baden

Betreuer

1. Prof. Dr. Wolfgang Ziegler, Hochschule Karlsruhe
2. Prof. Dr. Christoph Lüth, Universität Bremen

Weitere Angaben

Anmeldung als Doktorand am 27.07.2015 unter dem Arbeitstitel: „Semantische Such- und Delivery-Plattformen für Multi-Source Produktinformationen“ am Fachbereich 3 Mathematik/Informatik der Universität Bremen

Abstract

Technische Dokumentation wird oft in Formaten bereitgestellt, die keinen granularen und semantischen Zugriff auf die benötigten Informationen zulassen. Dadurch können viele Inhalte nicht in moderne Digitalisierungsszenarien integriert werden und sind aus Sicht des Information Retrieval schwer zugänglich. In dieser Arbeit werden vier Optimierungen vorgestellt, die durch den Einsatz von Maschinellem Lernen und semantischen Technologien die notwendige Aufbereitung automatisieren. Die Ergebnisse der durchgeführten Versuche zeigen, dass eine domänenspezifische Anpassung verbreiteter Verfahren eine wesentliche Verbesserung des Informationszugriffs bewirken.

Auf Basis etablierter Konzepte in der Technischen Dokumentation wird zunächst ein Optimierungsmodell für den semantischen Informationszugriff entwickelt, welches Methoden und ihre softwaregestützte Anwendung in ein einheitliches Schema einordnet. Vier konkrete Untersuchungen gehen jeweils im Detail auf die speziellen Herausforderungen im Bereich der Technischen Dokumentation ein: die semantische Ähnlichkeitsanalyse zur Identifikation von unkontrollierten Varianten, die automatisierte Klassifizierung von Modulen im Content Management, die formatierungsunabhängige Segmentierung von PDF-Dokumenten und die einheitliche Annotation heterogener Datenquellen mit semantischen Konzepten.

Inhaltsverzeichnis

1 Einleitung 11

1.1 Motivation 11
1.2 Forschungsziel 12
1.3 Umfeld 12
1.4 Abgrenzung 13
1.5 Aufbau dieser Arbeit 14
1.6 Verwandte Arbeiten 14

2 Konzepte 17

2.1 Standardisierte Texterstellung 17
2.2 Strukturierte Technische Dokumentation 18
2.3 Metadaten und Klassifikation 18
2.4 Intelligente Informationen 20
2.5 Ontologien und semantische Netze 21
2.6 Information Retrieval 22
2.7 Semantischer Informationszugriff 24
2.8 Künstliche Intelligenz 27

3 Das CoSMOS-Modell 29

3.1 Grundlagen 29
3.1.1 Methoden 29
3.1.2 Ebenen 30
3.1.3 Softwaregestützte Anwendung 31

3.2 Ebenen 31
3.2.1 Content 31
3.2.2 Struktur 34
3.2.3 Metadaten 36
3.2.4 Ontologien und semantische Netze 39

3.3 Anwendung 41
3.3.1 Optimierung und Empfehlungen 41
3.3.2 Forschung 42
3.3.3 Reifegrade 42

4 Untersuchungen 45

4.1 Überblick 45
4.1.1 Zusammenhang 45
4.1.2 Reihenfolge 46

4.2 Semantische Ähnlichkeitsanalyse 47
4.2.1 Zusammenfassung 47
4.2.2 Einleitung 47
4.2.3 Verwandte Arbeiten 49
4.2.4 Methodologie und Versuchsaufbau 50
4.2.5 Semantische Gewichtung 52
4.2.6 Implementierung 53
4.2.7 Ausblick 57
4.2.8 Fazit 57

4.3 Automatisierte Klassifizierung 59
4.3.1 Zusammenfassung 59
4.3.2 Einleitung 59
4.3.3 Verwandte Arbeiten 61
4.3.4 Methodologie 63
4.3.5 Charakteristiken 67
4.3.6 Auswirkungen und Anpassungen 71
4.3.7 Ergebnisse und Diskussion 77
4.3.8 Implementierung 80
4.3.9 Anwendungen 81
4.3.10 Ausblick 82
4.3.11 Fazit 85

4.4 Automatisierte Segmentierung 87
4.4.1 Zusammenfassung 87
4.4.2 Einleitung 87
4.4.3 Verwandte Arbeiten 90
4.4.4 Methodologie 91
4.4.5 Evaluierung 99
4.4.6 Implementierung 103
4.4.7 Anwendung 103
4.4.8 Ausblick 104
4.4.9 Fazit 104
4.4.10 Weitere Ergebnisse 105

4.5 Informationsintegration ... 109
4.5.1 Zusammenfassung ... 109
4.5.2 Einleitung ... 109
4.5.3 Verwandte Arbeiten ... 113
4.5.4 Unterstützte Berichtserstellung ... 114
4.5.5 Annotation von Technischer Dokumentation ... 118
4.5.6 Standardisierte semantische Annotation ... 123
4.5.7 Anwendung ... 126
4.5.8 Implementierung ... 128
4.5.9 Evaluierung ... 128
4.5.10 Fazit ... 132
4.5.11 Ausblick ... 132

5 Zusammenfassung und Ausblick ... 133

5.1 Zusammenfassung ... 133
5.1.1 Zusammenfassung der Untersuchungen ... 133
5.1.2 Übergreifende Ergebnisse ... 135
5.1.3 Kritische Betrachtung ... 136

5.2 Ausblick ... 137

6 Literaturverzeichnis ... 138

6.1 Publikationen im Rahmen der Dissertation ... 138
6.2 Vollständige Bibliographie ... 140

7 Anhang ... 153

7.1 Übersicht Quellcode und Demos ... 153
7.2 Codebeispielverzeichnis ... 153
7.3 Tabellenverzeichnis ... 153
7.4 Abbildungsverzeichnis ... 154

Abkürzungsverzeichnis

AMS	Authoring-Memory-System
AWS	Amazon Web Services
API	Application Programming Interface
CDP	Content-Delivery-Portal
CLC	Controlled Language Checker
CM	(Component) Content Management
CMS	Content-Management-System
DIN	Deutsches Institut für Normung
DITA	Darwin Information Typing Architecture
DTD	Document Type Definition
EU	Europäische Union
GUI	Graphical User Interface (*Benutzeroberfläche*)
HTML	Hypertext Markup Language
IEC	International Electrotechnical Commission
iiRDS	intelligent information – Request and Delivery Standard
IR	Information Retrieval (*Informationszugriff*)
ISO	International Organization for Standardization
JS	JavaScript
JSON	JavaScript Object Notation
KI	Künstliche Intelligenz
ML	Machine Learning (*Maschinelles Lernen*)
OASIS	Organization for the Advancement of Structured Information Standards
OCR	Optical Character Recognition (*optische Zeichenerkennung*)
OWL	Web Ontology Language
PDF	Portable Document Format
PI	Produkt/Information
RDF	Resource Description Framework
RDFS	RDF Schema
SEO	Search Engine Optimization (*Suchmaschinenoptimierung*)
SPARQL	SPARQL Protocol And RDF Query Language
SVM	Support Vector Machines
TD	Technische Dokumentation
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VSM	Vector Space Model (*Vektorraummodell*)
W3C	World Wide Web Consortium
XML	Extensible Markup Language

Übliche Abkürzungen nach DIN 5008:2011

Danksagung

Besonderer Dank gilt meinem Doktorvater Prof. Dr. Wolfgang Ziegler, der mich immer auf meinem akademischen Weg unterstützt hat und ohne den diese Arbeit nicht möglich gewesen wäre. Meinem zweiten Doktorvater Prof. Dr. Christoph Lüth danke ich für inspirierende Gespräche und seinen fachlichen Beistand. Außerdem möchte ich Prof. Dr. Jörg Hennig für seinen motivierenden Ansporn und Toni Zieger für sein gründliches Lektorat danken.

Für die bedingungslose Unterstützung meiner Forschung bin ich Stephan Steurer gegenüber zu außerordentlichem Dank verpflichtet. Meinen Kollegen bei der ICMS GmbH möchte ich für ihre Unterstützung und Geduld ebenso danken, wie den Firmen und Kunden, die für zahlreiche Versuche ihre Daten bereitgestellt haben.

Ein herzliches Dankschön gilt meinen beiden Eltern, meinem Bruder und meiner Verlobten Kim, die immer für mich da waren.

Diese Arbeit ist meinem Opa – *Günther Kölmel (1933-2015)* – gewidmet,
der mir beigebracht hat, immer neugierig zu sein.

Karlsruhe, im April 2019 Jan Oevermann

1 Einleitung

1.1 Motivation

Die digitale Transformation in der Industrie, neue Zielmedien und Endgeräte sowie immer größer werdende Datenmengen machen die gezielte Bereitstellung von Informationen zu einer Herausforderung. Das größte Hindernis bei der Informationsrecherche stellt mittlerweile nicht mehr der Mangel an möglichen Datenquellen, sondern die Fülle an potenziell relevanten Inhalten dar (CARLSON, 2003; LEASE, 2018; WADHWA, PATERIYA & SHRIVASTAVA, 2019). Spezifische Informationen schnell und kontextabhängig zur Verfügung zu stellen ist demnach eine der großen Herausforderungen moderner Informationssysteme (ALLAN u. a., 2003; BELKIN, 2008).

Etablierte Methoden zur Erfassung, Strukturierung und Annotation von Technischer Dokumentation können bereits heute einen granularen semantischen Zugriff auf Inhalte über mehrere Quellen hinweg ermöglichen (ZIEGLER, 2018a); ihre Umsetzung ist in der Regel jedoch mit einem erheblichen Arbeitsaufwand verbunden. Die nachträgliche „Semantifizierung" von Bestandsdaten ist deshalb für die Industrie von großem Interesse, bringt aber auch viele Herausforderungen mit sich (FURTH, 2018). Durch Innovationen im Bereich der Künstlichen Intelligenz und durch die Methoden des semantischen Webs ist es möglich, Teile dieser Aufbereitung zu automatisieren und so personelle Arbeitsaufwände zu verringern (OEVERMANN, 2017a). Zu diesen Optimierungen gehören die Analyse der Inhalte auf Redundanzen, die automatisierte Vergabe von klassifizierenden Metadaten sowie die Segmentierung von Dokumenten in geeignete Informationseinheiten und die standardisierte Annotation mit semantischen Konzepten.

Ein effektiver semantischer Zugriff auf technische Inhalte kann in Content-Delivery-Portalen oder Information-Retrieval-Anwendungen dazu verwendet werden, den Endkunden-Support zu entlasten, Serviceeinsätze zu optimieren oder neue *Industrie-4.0*-Szenarien zu realisieren (SCHAFFNER, 2017). Neben dem erhöhten Komfort für den Anwender sind auch Zeit- und Kostenersparnisse entscheidende Gründe für den größer werdenden Bedarf an „intelligenten Informationen".

Dem gegenüber steht eine große Menge an Technischer Dokumentation in Formaten, die für solche Szenarien nicht geeignet sind und damit oft von neuen Verwendungsmöglichkeiten ausgeschlossen bleiben. Um diese Bestandsdaten zu integrieren, wird ein ganzheitliches Vorgehensmodell benötigt, das Methoden für verschiedene Content-Ebenen zusammenfasst und auf die Besonderheiten der Textsorte *Technische Dokumentation* Rücksicht nimmt.

1.2 Forschungsziel

Ziel der Arbeit ist die Erarbeitung und Erforschung von Optimierungen für einen semantischen Informationszugriff auf Technische Dokumentation, z. B. über Content-Delivery-Portale, Suchmaschinen oder Linked-Data-Anwendungen. Der Fokus liegt dabei auf automatisierten Verfahren, die mit Hilfe von Maschinellem Lernen und semantischen Technologien umgesetzt werden können. Diese Werkzeuge sollen dann als Ergänzung zu manueller methodischer Arbeit im redaktionellen Alltag eingesetzt werden können. Als Grundlage für die Einordnung und Bewertung dieser Optimierungen soll ein Modell entwickelt werden, das die relevanten Ebenen für einen semantischen Informationszugriff definiert und die dazugehörigen Methoden sowie maschinelle Kontroll-, Analyse- oder Automatisierungsmöglichkeiten zuordnet. Im Gegensatz zu der in der Künstlichen Intelligenz oft angestrebten vollständigen Automatisierung von Aufgaben, soll eine sogenannte „Intelligence Amplification" (ROGERS, KABRISKY, BAUER & OXLEY, 2003) verfolgt werden.[1] Menschen werden dabei von Maschinen intellektuell unterstützt, um ihre Aufgaben besser und effizienter ausführen zu können.

Die Übertragung von bekannten Methoden auf den domänenspezifischen Einsatz in der Technischen Dokumentation bildet den Kernaspekt der Arbeit. Als Ergebnis sollen konkrete Analyse- und Automatisierungsverfahren untersucht werden, um Inhalte für die Verwendung in semantischen Information-Retrieval-Szenarien vor- oder aufzubereiten – vom Text bis zum semantischen Netz. Die praxisnahe Anwendung soll durch lauffähige Implementierungen und Evaluierungen mit Echtdaten untermauert werden.

1.3 Umfeld

In der verarbeitenden Industrie wurde mit der „High-Tech Strategie 2020" (BUNDESMINISTERIUM FÜR BILDUNG UND FORSCHUNG, 2014) bereits der Kurs für Industrie 4.0 vorgegeben: „Losgröße 1", umfassende Automatisierung und die Vernetzung von cyber-physikalischen Systemen zur optimalen Produktionsauslastung. Wichtigster Treiber ist dabei der Einsatz von Künstlicher Intelligenz (BRYNJOLFSSON & MCAFEE, 2016), um damit z. B. Maschinenausfälle vorherzusagen („Predictive Maintenance") oder Fabriken intelligent zu machen („Smart Factories"). Von diesen Entwicklungen bleibt auch die Technische Dokumentation nicht unberührt (vgl. dazu auch OEVERMANN, 2018a; SCHAFFNER, 2017). Ein Wandel in Produktion und Produkten wirkt sich immer auch auf die begleitende Dokumentation aus.

[1] Die Idee, den Menschen nicht nur rein mechanisch sondern auch intellektuell zu verstärken, wurde zuerst von BUSH (1945) aufgegriffen.

Durch sogenanntes *Mass Customization* steigt nicht nur die Variantenvielfalt der Produkte, sondern auch der zugehörigen Dokumentation, was wiederum ein angepasstes Informationsmanagement erfordert (ZIEGLER, 2016). Diese Entwicklungen treiben innerhalb der Technischen Dokumentation die Entstehung von „digitalen Informationsservices" in industriellen Anwendungen voran (ZIEGLER, 2018b). Mit dem „Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0)" existiert bereits eine übergeordnete Methode zur Einordnung und Koordinierung von verschiedenen Standards aus den Bereichen Maschinenkommunikation und Informationsaustausch, die auch Technische Informationen in einer „Verwaltungsschale" mit einschließt (vgl. u. a. HEIDEL, HOFFMEISTER, HANKEL & DÖBRICH, 2017 und ZVEI, 2016).

Die Wandlung von Produkten zu Dienstleistungen („Servitization") birgt für Hersteller neue Anreize, um z. B. den eigenen Service effizienter zu gestalten, da mit jedem Stillstand einer Maschine erhebliche Verdienstausfälle einhergehen (BAINES, LIGHTFOOT, BENEDETTINI & KAY, 2009). Um die Einsatzzeit im Feld zu reduzieren, kann hier insbesondere der Aufwand bei der Informationsrecherche reduziert werden (YAMAUCHI, WHALEN & BOBROW, 2003). Dazu geeignet sind etablierte Methoden aus dem Information Retrieval, wie z. B. facettierte Suchen, XML-Strukturen oder semantische Wissensgraphen (HJØRLAND, 2013; LALMAS, 2009; ZHENG, ZHANG & FENG, 2013).

Auch die Rolle des Technischen Redakteur verändert sich dahingehend, dass sich die Arbeit mehr mit der Modellierung von semantischen Metadaten und Ontologien beschäftigen wird (SCHAFFNER, 2019) und die Erstellung neuer Inhalte mit Hilfe von unterstützenden Systemen erfolgen wird (siehe z. B. DREWER & ZIEGLER (2011), S. 428f und OEVERMANN (2017a)).

1.4 Abgrenzung

Neben grundlegenden Konzepten in der Technischen Dokumentation und einem übergeordneten Optimierungsmodell für den semantischen Zugriff, wird im Hauptteil der Arbeit vor allem auf deren softwaregestützte Anwendung eingegangen. Die vier vorgestellten Untersuchungen sind als Beispiele der vorgestellten Optimierungsmethoden in den Bereichen der Automatisierung und Autorenunterstützung zu verstehen und stellen keine vollständige oder umfassende Auflistung der bestehenden Möglichkeiten dar, sondern nur einen Ausschnitt daraus.

Die im Modell genannten Vorgehensweisen beschränken sich meist auf textuelle Inhalte aus der Technischen Dokumentation und angrenzenden Gebieten mit ähnlichen Anforderungen. Konkrete Einschränkungen hinsichtlich der Methodologie finden sich direkt bei den entsprechenden Untersuchungen.

1.5 Aufbau dieser Arbeit

Diese Arbeit ist als kumulative Dissertation konzipiert. Neben dem vorliegenden Text wurden wesentliche Ergebnisse in peer-reviewten Publikationen im Rahmen der Promotion veröffentlicht. Die dazugehörigen Artikel und Konferenzbeiträge sind im Abschnitt 6.1 aufgelistet. Diese Veröffentlichungen behandeln jeweils Teilaspekte des übergeordneten Themas und werden durch das vorliegende Dokument in Zusammenhang gebracht. Die ursprünglich auf Englisch publizierten Texte wurden im Rahmen dieser Arbeit ins Deutsche übersetzt, der Inhalt mit entsprechender Quellenkennzeichnung übernommen und in die Gesamtstruktur eingefügt. An geeigneten Stellen wurden Hinweise, Fußnoten, Abbildungen oder Abschnitte ergänzt, die in den originalen Veröffentlichungen nicht vorkommen. Grafiken wurden – soweit möglich – ins Deutsche übertragen.

Nachdem in der Einleitung auf Problemstellung, Ziele und Aufbau der Arbeit eingegangen wurde, wird zunächst der Begriff des semantischen Zugriffs auf Technische Dokumentation näher beleuchtet und verwandte Konzepte vorgestellt. Darauf basierend wird ein Übersichtsmodell skizziert, das die wichtigsten Optimierungen, Methoden und softwaregestützten Anwendungen einordnet und erklärt. Basierend auf dem entwickelten Modell werden im Hauptteil der Arbeit (Kapitel 4) vier konkrete Untersuchungen zur Verbesserung des semantischen Zugriffs auf Technische Dokumentation vorgestellt, im Detail diskutiert und evaluiert. Dazu wird immer eine mögliche Implementierung der jeweiligen Methode betrachtet. Diese Abschnitte können unabhängig voneinander gelesen werden und enthalten jeweils eigene Einleitungen, verwandte Arbeiten, Methodologien und Ergebnisse.

Abschließend werden die einzelnen Resultate in Zusammenhang zueinander gebracht, die Erkenntnisse zusammengefasst und ein Ausblick auf weiterführende Forschungsansätze gegeben. Den Schlussteil der Arbeit bilden eine ausführliche Bibliographie, Verzeichnisse sowie Verweise auf Quellcode und Prototypen.

1.6 Verwandte Arbeiten

Die akademische Forschung zu automatisierter semantischer Inhaltserschließung in der Technischen Dokumentation und deren Auswirkungen auf den Informationszugriff ist noch wenig ausgeprägt und wird vor allem durch interdisziplinäre Arbeiten beeinflusst. Die Bereiche *Information Retrieval* (vor allem Portale und Suchmaschinen), *Document Engineering* (mit Fokus auf Maschinellem Lernen) und *Semantische Systeme* bzw. Technologien befassen sich teilweise mit Anwendungen in der Technischen Kommunikation, oft auch in Zusammenhang mit industriellen Forschungsprojekten.

FURTH (2018) befasst sich in seiner Arbeit ausführlich mit der (teilweise automatisierten) „Semantifizierung" von Technischer Dokumentation und stellt neben verschiedenen Ansätzen und Beispielen von Systemen auch ein Reifegradmodell[2] zur Bewertung von Dokumentationen vor. Durch die parallele Entstehung der Arbeiten werden auch einige der hier vorgestellten Methoden genannt und eingeordnet (FURTH, 2018, S. 107, 124, 233). Unterschiede zur vorliegenden Arbeit liegen im Ziel der semantischen Anreicherung (Verknüpfung von Inhalten vs. Aufbereitung für Zugriff) und den betrachteten Informationsebenen (verschiedene „strukturelle Komponenten" bis auf Wortebene vs. Module, vgl. FURTH, 2018, S. 39).

REUßNER (2018) befasst sich in seiner Arbeit mit der Klassifizierung von Technischer Dokumentation und fokussiert sich dabei auf den Vergleich zwischen manuellen und automatisierten Methoden zur domänenspezifischen Modulklassifikation sowie der Konzeption einer Facettensuche.

Weitere Forschung zur automatischen Analyse und Verarbeitung von Technischer Dokumentation finden sich insbesondere bei CAPONI, DI IORIO, VITALI, ALBERTI & SCATÁ (2018), CASCINI, FANTECHI & SPINICCI (2004) und SOTO u. a. (2015). Ähnliche Ansätze aus anderen Domänen finden sich u. a. bei CALDAS, SOIBELMAN & HAN (2002); SOFEAN, ARAS & ALRIFAI (2018); UREN u. a. (2006).

Verfahren zur Dokumentsegmentierung finden sich u. a. bei DÉJEAN & MEUNIER (2006); FANG, TANG & GAO (2011); MEUNIER (2010); NAZEMI, MURRAY & MCMEEKIN (2014) und PAAß & KONYA (2011). Die Problematik der immer komplexer werdenden Informationsrecherche wird speziell im Bereich des industriellen Service untersucht (SCHWEITZER & AURICH, 2010; YAMAUCHI u. a., 2003).

Gut erforscht ist das Information Retrieval im Allgemeinen (BAEZA-YATES & RIBEIRO-NETO, 1999; MANNING, RAGHAVAN & SCHÜTZE, 2008; SINGHAL, 2001) und die zu bewältigenden Herausforderungen in modernen Einsatzszenarien im Speziellen (ALLAN u. a., 2003; KUHLTHAU, 2005).

Methodisches (Enterprise) Content Management wird in der Literatur ausführlich behandelt (ANDERSEN, 2011; DREWER & ZIEGLER, 2011; GRAHLMANN, HILHORST, VAN AMERONGEN, HELMS & BRINKKEMPER, 2010; ROCKLEY & COOPER, 2012; ZIEGLER, 2013a), ebenso wie das verwandte Konzept des Content Delivery (STEURER, 2017; ZIEGLER, 2018a; ZIEGLER & BEIER, 2015).

[2] Das „5-STARs maturity schema for technical documentation", vgl. auch Abschnitt 3.3.3.

Die für die spezifischen Untersuchungen relevanten Arbeiten und kurze Zusammenfassungen dieser finden sich in den jeweiligen Abschnitten 4.3.3 (automatisierte Klassifizierung), 4.4.3 (automatisierte Segmentierung), 4.2.3 (Ähnlichkeitsanalyse) und 4.5.3 (Service-Informationssysteme, Informationsintegration, semantische Annotation).

2 Konzepte

2.1 Standardisierte Texterstellung

Die standardisierte Texterstellung ist im Bereich der Technischen Dokumentation tief verankert und wird durch etablierte Methoden umgesetzt. Bei Betrachtung der speziellen Anforderungen, die an die Textsorte gestellt werden (Verständlichkeit, Übersetzbarkeit, Wiederverwendbarkeit, etc.) wird deutlich, welche Bedeutung einheitlich erstellte Inhalte haben. Oft arbeiten mehrere Autoren innerhalb einer Technischen Redaktion, die Informationen aus unterschiedlichen Unternehmensbereichen zusammentragen, um sie dann in verschiedenen Publikationen zusammenzufassen und anschließend in zahlreiche Zielmedien zu publizieren. Der methodische Unterbau für die Standardisierung reicht dabei von der Wortebene bis zu ganzen Kapiteln. Die in einer Technischen Redaktion entstandenen „qualifizierten Inhalte" werden dabei in der Regel als Content bezeichnet (DUDEN, 2018). Dies beinhaltet sowohl textuellen als auch visuellen Inhalt (DREWER & ZIEGLER, 2011, S. 297).

In der *Terminologiearbeit*[3] werden Erkenntnisse aus der *Terminologielehre* praktisch angewandt, um Fachausdrücke zu definieren, zu strukturieren und festzulegen; grammatische *Schreibregeln*, die in Redaktionsleitfäden formuliert werden oder der Einsatz von *kontrollierten Sprachen* stellen sicher, dass Texte verständlich, übersetzbar und einheitlich sind (DREWER & ZIEGLER, 2011, S. 191f). Über die Satzebene hinaus kommen spezielle *Standardisierungsmethoden* (wie z. B. Funktionsdesign® oder Information Mapping®) zum Einsatz, welche wiederkehrende sprachliche Muster definieren und ihnen ihre sprechakttheoretische Funktion zuweisen (vgl. Beiträge bei MUTHIG, 2008). Die Nano- und Mikrostrukturen in semantischen Informationsmodellen (FURTH, 2018; KRÜGER & ZIEGLER, 2008) können ebenfalls zur Standardisierung beitragen, in dem z. B. vorgegebene Elemente die inhaltlichen Muster einer Methode widerspiegeln. Auf der inhaltlichen Ebene ist die Vermeidung von *unkontrollierten Redundanzen* für eine effiziente Verwaltung und Wiederverwendung sowie Qualität und Effizienz bei der Erstellung unerlässlich (DREWER & ZIEGLER, 2011, S. 271). Diese doppelten oder sehr ähnlichen Texte treten z. B. durch „Copy&Paste" oder eine doppelte Inhaltserfassung auf und können sich auch im Information Retrieval negativ auswirken (BERNSTEIN & ZOBEL, 2005). Zu den Autorenunterstützungen für eine standardisierte Texterstellung gehören u. a. Sprachkontrollwerkzeuge (z. B. CLC) oder Authoring-Memory-Systeme.

[3] Wichtige Aspekte der Terminologiearbeit finden sich auch in semantischen Netzen wieder (vgl. Abschnitt 2.5): Definitionen, eindeutige Bezeichner und Begriffsysteme, die Terme in Beziehung setzen. Terminologie kann demnach auch Grundlage für eine einfache Ontologie sein.

2.2 Strukturierte Technische Dokumentation

Neben der rein sprachlichen Ebenen des Inhalts spielt auch die Strukturierung im Content Management von Technischer Dokumentation eine große Rolle (KRÜGER & ZIEGLER, 2008). Die methodische Aufteilung eines monolithischen Dokuments in kleinere Informationseinheiten unterstützt eine einheitliche Struktur von Dokumentationen, die Ablage von sowie die Suche nach Inhalt und die übergreifende Wiederverwendung (DREWER & ZIEGLER, 2011, S. 306). Im Umfeld von XML-basierten Redaktionssystemen spricht man im deutschsprachigen Raum von Modulen, die als in sich abgeschlossene Informationseinheiten erstellt werden und deren äußere Form in der Regel durch die Makrostrukturen von Informationsmodellen vorgegeben wird. Alternativ werden diese Inhaltsfragmente auch als Topics, Content Components oder Textbausteine bezeichnet. Da in der Technischen Dokumentation die Modularisierbarkeit eine wichtige Rolle spielt, ist eine Klassifizierung nach Strukturierungsgrad sinnvoll. Prinzipiell kann Content in drei Stufen eingeteilt werden (DREWER & ZIEGLER, 2011, S. 297): strukturierter Content (z. B. XML-Daten), schwach strukturierter Content (z. B. Word-Dateien) und unstrukturierter Content (z. B. die meisten PDF-Varianten).

2.3 Metadaten und Klassifikation

Metadaten enthalten weiterführende Informationen über Inhalt. In der Technischen Dokumentation sind das typischerweise vom Redaktionssystem vergebene (Bearbeiter, Erstellungsdatum, ID), Angaben über Art und Bezug des Contents (Inhaltstyp, Zielgruppe, beschriebenes Produkt) sowie prozessrelevante Informationen (Notizen, Status, Priorität). Je nach System und Zweck des Metadatums können die Werte per Freitext, Checkbox, Liste oder Taxonomie-Baum vergeben werden. Des Weiteren unterscheidet man zwischen Metadaten, die vom System vergeben werden („generativ“) und Angaben, die der Anwender pflegt. Speziell bei inhaltsbezogenen Metadaten ist es von Vorteil, einer zuvor erarbeiteten Systematik zu folgen, um sowohl die Vergabe als auch den Informationszugriff so effizient und verständlich wie möglich zu gestalten. Ein festgelegtes Metadatensystem hilft Technischen Redakteuren bei der eindeutigen Kategorisierung von Informationen und ermöglich eine automatisierte Verarbeitung, z. B. in einem zielgruppenspezifischen Publikationsprozess oder bei der Dokumentaggregation. In einem Klassifikationssystem bilden die Klassen Achsen in einem mehrdimensionalen Informationsraum und die möglichen Klassifikationswerte Abschnitte darauf (vgl. Abbildung 1). Eine Informationseinheit kann dann über ihre Position im Informationsraum (Kombination aus Klassifikationen) eindeutig identifiziert werden (DREWER & ZIEGLER, 2011, S. 328; VON LEPEL & FINKLER, 2010). Dadurch ist ein sehr effizientes Information Retrieval möglich.

Eine der verbreitetsten Methoden zur systematischen Klassifikation von Technischer Dokumentation ist PI-Class® (z. B. DREWER & ZIEGLER, 2011). PI-Klassifikationen werden als Taxonomien definiert und können systemunabhängig eingesetzt werden. Intrinsische Klassifikationen kategorisieren eindeutig die Informationsart des Inhalts (sog. „Informationsklasse“) und verknüpfen ihn mit der beschriebenen Produktkomponente (sog. „Produktklasse“). Extrinsische Klassifikationen ergänzen die Methode um die geplante Verwendung des Moduls für Produktmodelle und Dokumenttypen oder Zielgruppen (OEVERMANN, 2017a).

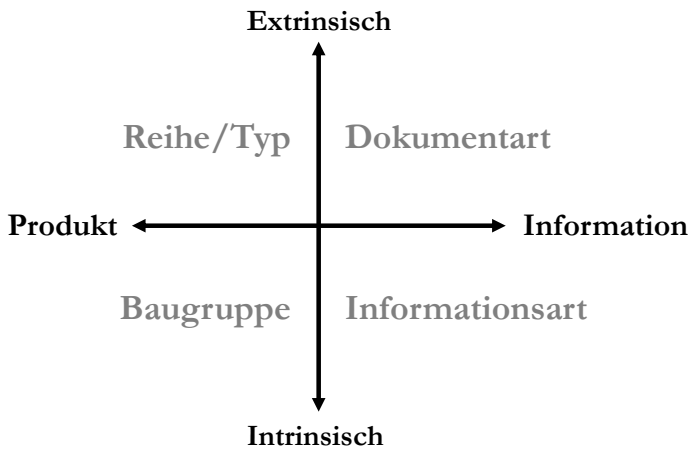

Abbildung 1: PI-Klassifikationsraum nach Ziegler
Mit beispielhaften Klassen für Dokumentation im Maschinen- und Anlagenbau.
Darstellung übernommen aus Oevermann (2016a)

Aufbauend auf der Einteilung in produkt- und informationsbezogene Metadaten wurde auch die übergeordnete Modellierung von Konzepten in iiRDS vorgenommen (PARSON, SAPARA & ZIEGLER, 2017). Das Schema unterscheidet Produktmetadaten (**`iirds:ProductMetadata`**) und Informationstypen (**`iirds:InformationType`**) als Oberklassen (BECKER u. a., 2018). Bei den in dieser Arbeit angewendeten Methoden zur automatisierten Klassifizierung spielt hingegen die Einteilung in intrinsische und extrinsische Metadaten eine große Rolle. So können z. B. in der automatisierten Auswertung eines Textes nur die Merkmale betrachtet werden, die sich direkt über den Inhalt erschließen (intrinsisch) und nicht solche, die sich rein über externe Bezüge ergeben (extrinsisch). Zu den intrinsische Konzepten in iiRDS zählen u. a. Informationsthema (**`iirds:InformationSubject`**), Topictyp (**`iirds:TopicType`**), Produktlebenszyklusphase (**`iirds:ProductLifeCyclePhase`**) und Baugruppe bzw. Komponente (**`iirds:Component`**) (OEVERMANN, 2017b).

2.4 Intelligente Informationen

Unter der Bezeichnung „intelligente Informationen“ (oder auch „intelligenter Content“) werden verschiedene Eigenschaften von Inhalten zusammengefasst, die speziell in Verbindung mit Information-Retrieval- oder Content-Delivery-Anwendungen von Relevanz sind.[4] Übergeordnetes Ziel ist dabei, einen granularen und kontextabhängigen Informationszugriff zu ermöglichen:

> „Intelligent content is content, that is structurally rich and semantically categorized, and is therefore automatically discoverable, reusable, reconfigurable and adaptable.“ (ROCKLEY & COOPER, 2012)

In dieser Arbeit werden dabei vor allem die zwei Grundvoraussetzungen für Content betrachtet, um als intelligente Information zu gelten: Struktur und klassifizierende Metadaten. Beide Prinzipien sind bereits durch modulbasierte Informationsmodelle und standardisierte Klassifikationsmethoden im Content Management etabliert und werden bei Inhaltserstellung und -verwaltung in XML-basierten Redaktionssystemen auch umgesetzt. Entscheidend für den Nutzen von intelligenten Informationen ist jedoch die Bereitstellung und Auswertung der Inhalte. Technische Dokumentation wird in den meisten Fällen noch über monolithische PDF-Dokumente publiziert (STRAUB, 2016). Dabei gehen jedoch in der Regel beide Eigenschaften von intelligenten Informationen – Struktur und Metadaten – durch die Dokumentform verloren.[5]

Eine gezielte Bereitstellung der Inhalte, die meist unter der Bezeichnung „Content Delivery“ verstanden wird, will dem entgegenwirken, indem Such- und Filtermechanismen direkt auf den strukturierten und mit Metadaten versehenen Content zugreifen und somit alle Vorteile von intelligenten Informationen nutzen (ZIEGLER & BEIER, 2015).[6] Content-Delivery-Portale bieten entsprechend „den Zugriff durch unterschiedliche Zielgruppen mit Hilfe von content-bezogenen Suchmechanismen“ (ZIEGLER, 2013b). Für die effektive Informationsbereitstellung innerhalb einer Organisation können intelligente Informationen erhebliche Vorteile bringen; die zugrundeliegenden Methoden stoßen jedoch an ihre Grenzen, sobald Datenquellen aus Unternehmen und Systemen mit unterschiedlichen Metadatenmodellen integriert werden sollen oder der Anwendungsfall ein Beziehungswissen erfordert, das über taxonomische Relationen hinausgeht.

---

[4] vgl. diverse Beiträge zu „Intelligente Information“ bei HENNIG & TJARKS-SOBHANI (2017).

[5] Es existieren auch PDF-Varianten, die Strukturinformationen beinhalten oder editierbar sind, diese sind jedoch wenig verbreitet (HASSAN, 2018).

[6] Zuerst als „Content-Delivery-Lücke“ zwischen CMS und Portalen bei ZIEGLER (2013b)

2.5 Ontologien und semantische Netze

Neben dem ursprünglichen Begriff der Semantik als Teilgebiet der Linguistik, das sich mit der Bedeutung von Wörter und Sätzen auseinandersetzt (DUDEN, 2019), hat sich im Umfeld der Technischen Dokumentation vor allem die Verwendung im Sinne des *Semantic Web* durchgesetzt: standardisierte Metadaten, welche die Bedeutung einer Information genauer definieren und mit Hilfe von festgelegten Beziehungen miteinander in Verbindung gebracht werden (LEY, 2018). Ziel dieser Vorgehensweise ist es, Wissen in geeigneter Form zu repräsentieren und für den Informationszugriff verfügbar zu machen.

Semantische Netze können in ihrer Grundform als Erweiterung der klassischen Taxonomie verstanden werden und repräsentierten Wissen in Form von Aussagen (REICHENBERGER, 2010). Diese Aussagen bestehen in der Regel aus Subjekt, Prädikat und Objekt, den sogenannten *Tripeln*, wobei das Prädikat der Beziehung zwischen den beiden Ressourcen Subjekt und Objekt entspricht (LEY, 2018). Diese Methodik ist auch die Grundlage von semantischen Technologien wie RDF (W3C, 2014a). Im Gegensatz zu relationalen oder taxonomischen Metadaten können komplexere Beziehungen abgebildet und somit neue Informationsanwendungen ermöglicht werden.

Die komplexeste und mächtigste Form der Wissensrepräsentation stellt die *Ontologie* dar, bei der es sich um eine „formale Konzeptualisierung eines Wissensbereichs handelt“ (GRUBER, 1993). Die Idee der semantischen Netze wird bei einer Ontologie um eine Hierarchie der Generalisierung bzw. Spezialisierung von Konzepten erweitert (STAAB & STUDER, 2009). Als Technische Umsetzung haben sich u. a. die Standards RDFS und OWL etabliert. Eine entscheidende Voraussetzung für den praktischen Einsatz von Ontologien wurde erst im Laufe der Zeit betrachtet: die Konzeptualisierung sollte eine von mehreren Parteien geteilte Sicht auf das Wissen darstellen.

> „[…] the conceptualization should express a *shared* view between several parties, a consensus rather than an individual view.“ (STAAB & STUDER, 2009)

Diese geteilte Sicht erfordert festgelegte Definitionen von Eigenschaften und Beziehungen, um eine semantische Interpretation der Informationen über System- und Organisationsgrenzen hinweg zu ermöglichen. Eine Standardisierung, die sowohl auf technischer Ebene als auch in Form von präzise definierten Vokabularen und ontologischen Beziehungen erfolgt, ermöglicht eine sogenannte *Informationsintegration* (vgl. z. B. WELLER, 2009), bei der aus beliebigen (auch heterogenen) Datenquellen Inhalte zusammengeführt und durchsucht werden können.

Eine möglichst umfassende Informationsintegration ist daher die ideale Ausgangssituation für ein effizientes Information Retrieval und die große Vision des Semantic Web (BERNERS-LEE, HENDLER & LASSILA, 2001). In dieser semantischen Variante des Internets sollen Daten möglichst fünf Eigenschaften haben, um sich *Linked Open Data* nennen zu dürfen: frei verfügbar im Netz, maschinenlesbar, in einem nicht-proprietären Format, standardisiert in der Art wie Ressourcen adressiert werden und mit anderen Datensätzen verknüpft (BERNERS-LEE, 2009). Durch die Anwendung von Linked-Data-Prinzipien erhalten Inhalte nicht nur einen verbesserten Informationszugriff, sondern es entsteht auch die Möglichkeit der Dezentralisierung von Informationen (VERBORGH, 2019). Dies ist vor allem dann relevant, wenn Daten unabhängig von Plattformen oder Systemen verarbeitet werden sollen.

Für den standardisierten Austausch von digitaler Dokumentation und den dazugehörigen Metadaten in digitaler Form wurde von einem Konsortium aus Firmen, Verbänden und Hochschulen der Standard „iiRDS" entwickelt (BECKER u. a., 2018). iiRDS steht für „intelligent information – Request and Delivery Standard" und kombiniert ein Containerformat zur Auslieferung der Inhaltsdateien mit einer standardisierten Domänenontologie für Technische Dokumentation. Bereitgestellt wird der Standard als freie Spezifikation und RDFS-Datei.[7] Im Rahmen dieser Arbeit wurde iiRDS sowohl konzeptionell mitentwickelt als auch im Rahmen von Untersuchungen angewandt (vgl. die Abschnitte 4.4.4.7, 4.5.6.2 und 5.1.1).

2.6 Information Retrieval

Unter der Bezeichnung *Information Retrieval* (dt. auch *Informationszugriff*) werden Konzepte und Methoden für die Informationsrückgewinnung gesammelt, die darauf abzielen, in einer Menge von komplexen Informationen die für den Anwender relevanten Ergebnisse zu finden. Bekanntestes Beispiel ist die Volltextsuche nach Website-Inhalten mit einer Internetsuchmaschine. Davon abzugrenzen ist das *Data Retrieval*, bei dem man von strukturierten und eindeutige Daten ausgeht, z. B. bei Datenbankabfragen (BAEZA-YATES & RIBEIRO-NETO, 1999). Ausgangsbasis für einen Informationszugriff über eine Suche oder eine Filterung ist in der Regel ein bestimmtes *Informationsbedürfnis*, das einer konkreten Frage oder einem Wissensbedarf eines Anwenders entspricht. Ziel des Information Retrievals ist demnach, dem Anwender (nur) relevante Ergebnisse zurückzuliefern. Die Ergebnisse können alle Arten komplexer Informationen sein, wie z. B. Texte, Bilder oder Videos.[8]

[7] Die Dateien können nach Anmeldung auf der Seite http://iirds.org heruntergeladen werden.

[8] Im Laufe dieser Arbeit werden nur textbasierte Inhalte betrachtet.

Zur Beurteilung der Leistungsfähigkeit eines Information-Retrieval-Systems werden in der Regel zwei Metriken ausgewertet: *Precision* und *Recall*, die wie folgt definiert werden (vgl. u. a. MANNING, RAGHAVAN & SCHÜTZE, 2008; SINGHAL, 2001):

$$Recall = \frac{(relevant\ \&\ gefunden)}{(relevant\ \&\ gefunden) + (relevant\ \&\ nicht\ gefunden)}$$

$$Precision = \frac{(relevant\ \&\ gefunden)}{(relevant\ \&\ gefunden) + (nicht\ relevant\ \&\ gefunden)}$$

Ein gutes Information-Retrieval-System liefert demnach vollständige Ergebnisse zurück (hoher Recall) und enthält dabei so wenig nicht-relevante Informationen wie möglich (hohe Precision). Durch die generell steigende Menge an Inhalten wird die Precision eines Systems immer wichtiger, in manchen Fällen muss jedoch auch sichergestellt werden, dass alle relevanten Informationen gesichtet werden (MANNING u. a., 2008).

Die Größe der zurückgelieferten Informationen hängt meist von der Datenbasis ab und kann von ganzen Dokumenten bis hin zu einzelnen Sätzen reichen.[9] Im Bereich der Technischen Dokumentation hat sich die Größe eines Moduls oder Topics als geeignet erwiesen, da diese per Definition in-sich-abgeschlossene Informationseinheiten darstellen (und damit im Idealfall genau ein Informationsbedürfnis abdecken). Die meisten modernen Information-Retrieval-Systeme können verschieden große Bezugseinheiten bei einer Suche verarbeiten.

Anwender formulieren bei einer Recherche ihr *Informationsbedürfnis* als Anfrage (engl.: „query") an das System. Bei einer einfachen Volltextsuche können das z. B. bestimmte Wörter sein, die im Ergebnis enthalten sein sollen. Aber auch komplexere Abfragen sind möglich, die z. B. eine Wortsuche mit Filtern verbinden (sog. facettierte Suche) oder den Nutzer bei der Formulierung der Anfrage unterstützen (z. B. durch eine Autovervollständigung). Dabei sollte immer beachtet werden, dass der Nutzen und die Relevanz der Information für den Anwender im Vordergrund stehen:

> „A document[10] is relevant if it addresses the stated information need, not because it just happens to contain all the words in the query."
> (MANNING u. a., 2008)

[9] Hier spricht man auch von *Dokumentarischen Bezugseinheiten* bzw. *Dokumentationseinheiten*. Da diese Bezeichnungen im Kontext von Modulen und Dokumenten verwirren können, werden sie in dieser Arbeit nicht verwendet. Die Einheiten sind hier entweder Module oder Dokumente.

[10] „Document" ist in diesem Kontext das zurückgelieferte Ergebnis einer Suchanfrage.

Content-Delivery-Portale, die zu den spezialisierten Information-Retrieval-Systemen gehören, setzen oft auf eine Kombination verschiedener Such- und Filteroptionen, die teilweise vom System anhand von Kontextinformationen vorbelegt werden. Auch in CMS ist eine leistungsfähige Suche nach Informationen von großem Vorteil für eine Steigerung der Wiederverwendungsrate.

2.7 Semantischer Informationszugriff

Diese Arbeit widmet sich dem *semantischen Informationszugriff* im Bereich der Technischen Dokumentation. Im Gegensatz zum „klassischen" Information Retrieval ist dabei die Semantik von Informationen bzw. deren Einordnung in das Domänenwissen für die Effektivität des Zugriffs in besonderer Weise relevant. Grund dafür ist ein grundlegender Wandel in der Art, wie Anwender nach Informationen suchen. Wo zuvor noch lange Dokumente als Ergebnis einer Recherche ausreichend waren, erwarten moderne Anwender heute, dass ihr Informationsbedürfnis direkt mit einem Fakt oder einer vorgeschlagenen Handlung beantwortet wird. Die Semantik spielt hierbei eine große Rolle.

> „[…] the emphasis will shift from finding documents to finding facts, actionable information, and insights." (SHETH, ARPINAR & KASHYAP, 2004)

Große Suchmaschinenanbieter wie Google haben bereits vor einigen Jahren darauf reagiert, indem die Volltextsuchen von Anwendern mit einer darunter liegenden Ontologie abgeglichen werden, um die eingegebene Zeichenkette („String") zu einer vernetzten Entität („Thing") in einem Wissensgraphen umzuwandeln und die Suche damit intelligenter zu machen.

> „It's why we've been working on an intelligent model – in geek-speak, a "graph" – that understands real-world entities and their relationships to one another: things, not strings." (SINGHAL, 2012)

Die Grundlage dafür ist eine sog. semantische oder konzeptionelle Suche[11], bei der sich Anfrage aus Termen oder Entitäten einer Ontologie zusammensetzen (siehe u. a. CASTELLS, FERNÁNDEZ SÁNCHEZ & VALLET WEADON, 2007; FERNÁNDEZ u. a., 2011). Semantische Suchen sind im Bereich des Information Retrieval bereits seit den 80er Jahren bekannt (CROFT, 1986), haben aber erst in den letzten Jahren ihren Weg in populäre Software gefunden. Auch Misch- und Zwischenformen existieren, bei denen konventionelle Volltextsuchen mit semantischen Netzen kombiniert werden (z. B. Entitätssuchen oder Struktursuchen, FREITAS, CURRY, OLIVEIRA & O'RIAIN (2012)).

[11] Im Laufe der Arbeit werden „semantische" und „konzeptionelle" Suche synonym verwendet.

Eines der größten Hindernisse für die Verbreitung von semantischen Suchtechnologien ist jedoch weiterhin die Schnittstelle zum Anwender. Dort entsteht ein Spannungsfeld zwischen guter Benutzerfreundlichkeit (einfache Texteingabe) und hoher Genauigkeit (Eingabe formaler Abfragesprachen[12], vgl. FERNÁNDEZ u. a. (2011); FREITAS u. a. (2012)). Im ersten Fall muss das verarbeitende System aus den Eingaben die semantischen Konzepte automatisiert ableiten können, im letzten Fall übernimmt diese Aufgabe der Anwender selbst. Dazwischen existieren zahlreiche Kompromisse, die dem Anwender z. B. während der Texteingabe semantische Konzepte aus einer Ontologie vorschlagen (vgl. „Semantische Autovervollständigung" in Abschnitt 4.5.4.1). In Abbildung 2 wird das Spannungsfeld mit beispielhaften Benutzerschnittstellen dargestellt.

Abbildung 2: Spannungsfeld des semantisch Informationszugriffs. Notwendiges Abwägen zwischen Benutzerfreundlichkeit und Ausdrucksstärke. Basierend auf der Darstellung bei Freitas, Curry, Oliveira & O'Riain (2012).

Google verwendet bei der Verarbeitung von Texteingaben im Hintergrund einen *Knowledge Graph*[13] (SINGHAL, 2012) zur Disambiguierung (z. B. Golf als Sport vs. Golf als Auto), Konzeptualisierung (z. B. Golf ist eine Art von Sport) und Traversierung (z. B. weitere Sportarten). In Abbildung 3 wird die aktuelle Ergebnisanzeige unter Anwendung der genannten Technologien für den Beispielterm „Golf" als sog. „Rich Snippet" (eine Art Infobox) gezeigt. Diese Art von semantischer Suche funktioniert allerdings nur dann, wenn eine entsprechende Ontologie zur Verfügung steht. Bei Anfragen zum Weltwissen kann diese Ontologie zum Teil über manuell gepflegte Wissensdatenbanken (wie z. B. Kategorien bei Wikipedia, DBpedia, CIA World Factbook etc.) aufgebaut werden, welche grundlegende Beziehungen oder Fakten definieren.

[12] Die Eingabe von SPARQL-Abfragen ist zwar sehr genau, erfordert jedoch ein erhebliche Vorwissen der Anwender.

[13] „Knowledge Graph" ist ein von Google verwendeter Marketingbegriff für eine automatisiert aggregierte Ontologie bzw. Wissensdatenbank (EHRLINGER & WÖß, 2016).

In spezialisierten Wissensdomänen wie der Technischen Dokumentation von komplexen Maschinen oder Software ist es jedoch erheblich schwieriger diese Grundlage zu schaffen, da wesentlich weniger Daten öffentlich zugänglich sind. Deshalb stehen hier vor allem die semantische Aufbereitung der Inhalte und ihrer Metadaten im Vordergrund.

Abbildung 3: Ausschnitt Screenshot, sog. Rich Snippet.
Ergebnis einer Google-Suche nach „Golf“.[14] Zu erkennen sind die Disambiguierung (Pos. 3: alternative Bedeutung als Auto), die Konzeptualisierung (Pos. 1: „Sport“) und die Traversierungsmöglichkeiten (Pos. 2: weitere Arten von „Sport“).

Die Bezeichnung „semantischer Informationszugriff“ soll hierbei als Spektrum verschiedener Aspekte des Information Retrievals verstanden werden, das bei der Textqualität beginnt und mit der semantischen Suche ihr Optimum findet. Das im nächsten Kapitel vorgestellte Modell zeigt, dass diese verschiedenen Informationsebenen jeweils unterschiedlich zu einem effizienteren Zugriff beitragen und aufeinander aufbauen.

[14] Zu erreichen unter: https://www.google.com/search?q=golf [29.01.19]

2.8 Künstliche Intelligenz

Durch die steigende Rechenleistung von Computern, die wachsende Menge an auswertbaren Daten und der Entwicklung neuartiger Algorithmen hat die Künstliche Intelligenz (KI) in den letzten Jahren eine Renaissance erfahren, die auch von der breiten Masse wahrgenommen wird (BRYNJOLFSSON & MCAFEE, 2016; HARARI, 2018). Als ein Teilgebiet der Informatik befasst sich die KI-Forschung schon seit den 50er Jahren mit der Imitation und Automatisierung von menschlicher Intelligenz (ERTEL, 2016). Dies beinhaltet die verschiedensten Anwendungen – von einfachen regel- oder fallbasierten Systemen bis hin zu selbstlernenden neuronalen Netzen. Für die Industrie sind vor allem spezialisierte Anwendungen von Bedeutung, die einen Teilbereich des menschlichen Denkens unterstützen oder sogar übernehmen können. Unter der Bezeichnung *Expertensysteme* wurden die ersten praktikablen Anwendungen schon seit den Siebzigerjahren entwickelt[15] (PUPPE, 2013) und mittlerweile als *Intelligence Amplification* (ROGERS u. a., 2003) betitelt. Das *Machine Learning* (dt.: Maschinelles Lernen) ist ein Teilgebiet der KI, welches auf Basis von Erfahrungen (also Beispielen oder vorhandenen Daten) neues Wissen generiert, das dann auf neue, dem System unbekannte Daten angewandt werden kann („Lerntransfer"). In der Regel wertet ein spezieller Algorithmus dabei Muster oder Gesetzmäßigkeiten aus, mit deren Hilfe dann Vorhersagen getroffen werden können. Das noch junge *Deep Learning* (dt. „tiefes Lernen) setzt dafür künstliche neuronale Netze ein, die über eine Vielzahl von Neuronenschichten verschiedene Eingangssignale zu Aussagen verarbeiten.

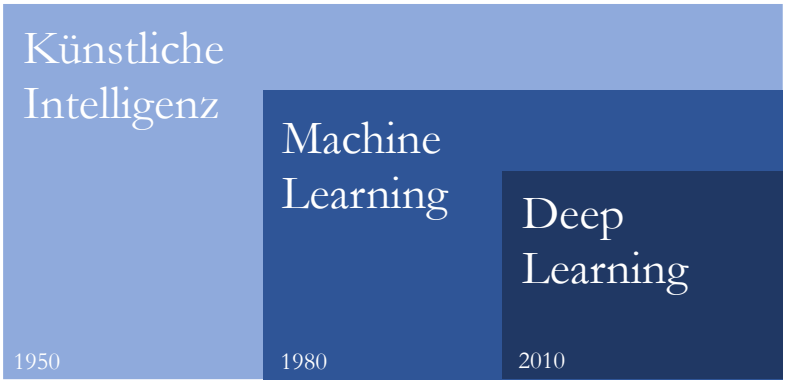

Abbildung 4: Machine Learning als Teilgebiet der Künstlichen Intelligenz. Grafik basiert auf Oevermann (2017a).

[15] Expertensysteme sind wiederum aus dem Scheitern von allgemeineren Problemlösungsansätzen in den Sechzigerjahren entstanden (PUPPE, 2013).

Die Technische Dokumentation kann mit ihren immer größer werdenden Mengen an Content und den sich stetig verkürzenden Produktlebenszyklen in besonderer Weise von KI profitieren (OEVERMANN, 2017a). Denn immer dann, wenn wiederkehrende Aufgaben für große Informationsmengen automatisiert werden sollen, lohnt sich meist der Einsatz von KI-Technologien aus wirtschaftlicher Sicht für Unternehmen (BRYNJOLFSSON & MCAFEE, 2016). Dazu zählt auch die Analyse und semantische Aufbereitung von Bestandsdaten, welche erhebliche personelle Aufwände verursachen kann und deshalb in den meisten Fällen unterlassen wird.

Die in dieser Arbeit vorgestellten Untersuchungen basieren in der Regel auf Methoden des angewandten maschinellen Lernens (*Applied Machine Learning*). Insbesondere bei den vorgestellten Verfahren zur automatisierten Klassifizierung (Abschnitt 4.2) und Segmentierung (Abschnitt 4.4) wird aus vorhandenen Daten neues Wissen in Form von Metadaten oder Annotationen generiert (überwachtes Lernen). Da die zugewiesenen Klassifikationen bzw. semantischen Konzepten intrinsischer Natur und daher einwertig sind, handelt es sich hierbei um eine Multiklassen-Klassifikation.[16] Bei der Ähnlichkeitsanalyse handelt es sich nicht direkt um ein KI-Verfahren, sondern um eine Automatisierung.

Die semantische Aufbereitung des Contents dient als Datengrundlage für die Verwendung in anderen Systemen, wie z. B. Content-Delivery-Portalen oder Assistenzsystemen im industriellen Service. Die prototypischen Implementierungen, wie z. B. die grafische Benutzeroberfläche zur Identifikation von unkontrollierten Varianten, kann den Expertensystemen zugeordnet werden, da domänenspezifisches Wissen zur Problemlösung eingesetzt wird.

Eine weitere kategorische Einordnung von „Intelligenz“ insbesondere für den Bereich der Technischen Dokumentation definiert ZIEGLER (2017):

- *Native Intelligenz* spiegelt sich z. B. in manuell vergebenen Metadaten wider, die in der Regel aber flach oder taxonomisch modelliert sind;
- *Erweiterte Intelligenz* ordnet diese Metadaten in einem größeren Beziehungsnetz ein (manuell oder softwareseitig unterstützt) und ermöglicht dadurch neuartige Anwendungen;
- *Künstliche Intelligenz* ist letztendlich die Automatisierung der Vergabe- und Modellierungsprozesse.

[16] Bei einer Multilabel-Klassifikation können mehrere Werte pro Vorhersage zugewiesen werden, bei einer Multiklassen-Klassifikation jeweils nur ein Wert.

3 Das CoSMOS-Modell

3.1 Grundlagen

Zur Optimierung des semantischen Informationszugriffs auf Technische Dokumentation müssen zunächst mögliche Ansatzpunkte gefunden werden, die einen positiven Einfluss auf die Leistungsfähigkeit von Information-Retrieval-Systemen haben.

3.1.1 Methoden

Die Einordung der Optimierungen erfolgt auf Basis der in Abschnitt 2 erläuterten methodischen Konzepte der Technischen Dokumentation in Form von vier Ebenen:

- Eine sprachliche Inhaltsebene befasst sich mit dem Text – unabhängig von seiner Makrostruktur: dem Content. Neben der Standardisierung einzelner Informationseinheiten ist auch die Konsistenz innerhalb des Inhaltsbestands wichtig, wozu z. B. die Datenqualität zählt.
- Die Strukturierung Technischer Dokumentation ist speziell in Verbindung mit Content-Delivery-Konzepten von enormer Bedeutung für den granularen Zugriff auf relevante Inhalte. Neben der strukturierten Erstellung in Informationsmodellen ist hier besonders die nachträgliche Aufbereitung monolithischer Dokumente von Interesse.
- Metadaten für Content sind in der Technischen Kommunikation immer schon dazu verwendet worden, Content besser zu organisieren und über eine gezieltes Information Retrieval (z. B. in einem CMS) zugänglich zu machen. Eine systematisierte Entwicklung dieser Klassifikationen oder Metadatenmodelle sowie die korrekte und effiziente Zuweisung sind die größten Herausforderungen in diesem Bereich.
- Die logische Fortführung dieser zusätzlichen Annotation von Inhalten sind die Prinzipien von Ontologien und semantischen Netzen: Metadaten können zueinander und mit Inhalten in verschiedenen Beziehungen stehen und sind universal definiert, um einen einfachen Austausch zu ermöglichen. Durch die netzartigen Relationen kann z. B. neues Wissen gewonnen werden und alternative Zugriffsmöglichkeiten auf die Informationen sind durch strukturierte Abfragen möglich. Größte Herausforderung bleibt ein benutzerfreundlicher und ganzheitlicher Ansatz zur „Semantifizierung“.

Daraus ergeben sich vier grundlegende Ebenen für die Optimierung:

Content, Struktur, Metadaten, Ontologien & Semantische Netze

oder kurz: **CoSMOS**.

3.1.2 Ebenen

Die Ebenen innerhalb des Modells bauen aufeinander auf und steigen in der Komplexität des erforderlichen Informationsmanagements an. Die einzelnen Stufen werden immer zusammen mit den darunter liegenden Ebenen betrachtet, weshalb CoSMOS auch als Reifegradmodell eingesetzt werden kann, um die Optimierungsstufe Technischer Dokumentation im Kontext des semantischen Informationszugriffs zu beurteilen (siehe auch Abschnitt 3.3.3).

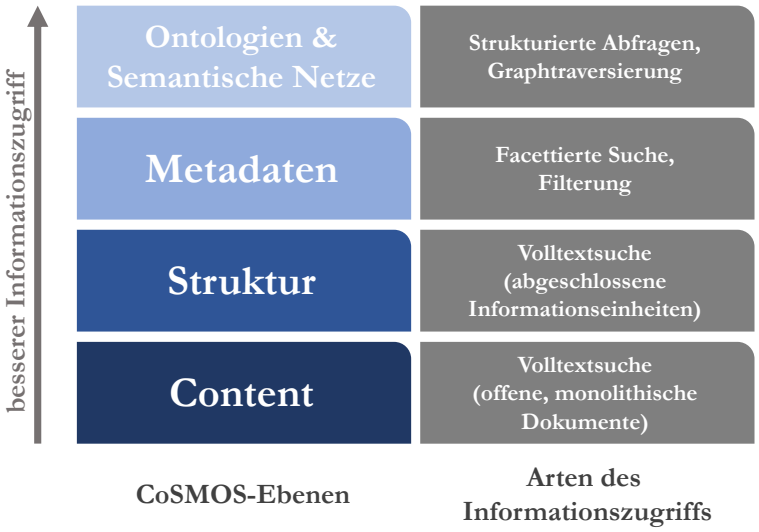

Abbildung 5: Arten des möglichen Informationszugriffs (CoSMOS-Ebenen). Die Effektivität des Zugriffs steigt mit jeder Ebene an.

Wie in Abbildung 5 zu sehen ist, hat die CoSMOS-Ebene, auf der die Informationen einzuordnen sind, eine direkte Auswirkung auf die möglichen Zugriffsarten und damit auch auf die Effektivität einer Suche. Eine mangelnde Strukturierung kann die Auswertung von Suchergebnissen erschweren; fehlende Metadaten verhindern die Einschränkung mittels nicht-sprachlicher Merkmale. Über erweitertes Beziehungswissen werden auch strukturierte Abfragen auf die Datenbasis möglich, z. B. mit SPARQL.

Mit einem verbesserten Informationszugriff steigt allerding auch die Komplexität der Umsetzung an, weshalb nicht für jeden Anwendungsfall alle Ebenen ausgenutzt werden müssen. Bei der Anwendung des Modells muss deshalb auch immer der individuelle Nutzen und die jeweilige Ausgangssituation betrachtet werden (siehe auch Abschnitt 3.3).

3.1.3 Softwaregestützte Anwendung

Die Übersicht konzentriert sich neben der Einordnung von existierenden Methoden oder Modellen auch auf die Gegenüberstellung mit der jeweiligen softwaregestützten Anwendung. Mit Hilfe der Ebenen lassen sich Automatisierungspotenziale erkennen und die Umsetzung einer Methodik durch den Einsatz von Software effektiver gestalten. Die softwaregestützte Anwendung kann hierbei eine (Teil-)Automatisierung der Methode bedeuten (z. B. durch den Einsatz Künstlicher Intelligenz) oder auch eine Assistenz bei der Umsetzung einer Methode sein (z. B. durch bestimmte Regeleinschränkungen oder Kontrollen). Eine weitere Möglichkeit stellt die nachträgliche Analyse des Inhalts dar, um die optimale Umsetzung der Methode zu prüfen oder zu verbessern. Diese drei Anwendungsarten werden in den Übersichten als „Assistenz", „Automatisierung" und „Analyse" bezeichnet. In manchen Fällen können die Übergänge zwischen diesen Arten jedoch auch fließend sein.

3.2 Ebenen

Die im letzten Abschnitt ermittelten Ebenen werden im Folgenden näher beleuchtet und Beispiele für zugehörige Methoden sowie ihre optimierte Umsetzung genannt. Die Ebenen bauen in der hier angegebenen Reihenfolge aufeinander auf und sollten deshalb immer im Zusammenhang zueinander betrachtet werden. Die angegebenen Auflistungen erheben keinen Anspruch auf Vollständigkeit, sondern sollen existierende Methoden sowie deren Anwendung einordnen und damit die Funktionsweise des Modells veranschaulichen.

3.2.1 Content

Bei einer Volltextsuche ist vor allem die Übereinstimmung der Anfrage mit dem indexierten Text ein entscheidender Faktor, ob eine Information gefunden werden kann. Stimmt die Terminologie des Nutzers nicht mit der im durchsuchten Inhalt verwendeten überein, kann das zu unbefriedigenden Ergebnissen führen (ZENG, KOGAN, ASH, GREENES & BOXWALA, 2002). In der Technischen Dokumentation steht hierbei vor allem die Konsistenz der Benennungen im Vordergrund, die mit Hilfe von diversen Terminologie-Kontrollwerkzeugen überwacht werden kann.

Zur Beurteilung der Relevanz von Ergebnissen spielt die Verständlichkeit des Inhalts eine große Rolle. Diese kann durch eine standardisierte oder sogar kontrollierte Sprache gesteigert werden (DREWER & ZIEGLER, 2011, S. 200). Mikrostrukturen[17] in Informationsmodellen können ebenfalls zum besseren

[17] Dazu gehören Elemente auf Satz- und Blockebene innerhalb eines Moduls, wie Listen oder Absätze, siehe auch „Technical Knowledge Ontology" bei FURTH (2018) oder bei KRÜGER & ZIEGLER (2008) als „standardisierte Informationsstrukturen auf Mikroebene".

Verständnis von Informationen beitragen, indem bestimmte textliche Muster in ihrer Struktur vorgegeben werden. Typische Beispiele in der Technischen Dokumentation sind Warnhinweise (Gefahr, Folgen, Maßnahmen, vgl. z. B. ANSI Z535.6, 2006) oder Handlungsanweisungen (Voraussetzung, Schritte, Ergebnis). Wiederkehrende Muster in Texten („Surface Text Patterns") können auch die Performance von bestimmten *Question-Answering-Systemen* erheblich verbessern (RAVICHANDRAN & HOVY, 2002).

Werden Benutzer nach einer Informationsabfrage mit einer Reihe von redundanten oder sehr ähnlichen Inhalten konfrontiert, steigt die Unsicherheit und die Effektivität der Suche sinkt (BERNSTEIN & ZOBEL, 2005). Die in der Technischen Dokumentation häufig vorkommenden Varianten können das weiter negativ beeinflussen, besonders wenn es sich dabei um „nicht-kontrollierte Varianten" handelt. Ungewollte Redundanzen können durch semantische Ähnlichkeitsanalysen verhindert werden (OEVERMANN & LÜTH, 2018). Kontrollierte Bezeichnungen sowie standardisierte Formulierungsmuster beeinflussen die Ergebnisse hingegen positiv und können mit Controlled Language Checkern bei der Erstellung forciert werden. Der funktionale Aufbau des Inhalts samt innerer Struktur und Ausgestaltung kann über Methoden wie Funktionsdesign® oder Information Mapping® definiert werden.

Als Suchmaschinenoptimierung (*Search Engine Optimization* oder kurz *SEO*) bezeichnet man verschiedene Methoden, die darauf abzielen, die in Suchmaschinen verwendeten Ranking-Algorithmen so zu beeinflussen, dass die gewünschten Ergebnisse für eine bestimmte Suchabfrage weiter oben platziert werden (EVANS, 2007). Dazu gehören u. a. die Verwendung der richtigen Terminologie für die Zielgruppe des Texts (s.o.), die Optimierung des Satzbaus auf bestimmte Wortmuster oder das bewusste Hinzufügen von Termen, die bei einer Volltextsuche ausgewertet werden sollen (KILLORAN, 2013). In der Technischen Dokumentation sind SEO-Maßnahmen vor allem dann von Interesse, wenn der Content im Web veröffentlicht wird und Anwender ihn z. B. bei Google finden sollen. Der CLC-Hersteller „Congree" nennt für den Bereich der Technischen Kommunikation explizit die Methode „linguistisches SEO" und argumentiert u. a. mit der Steigerung der Textverständlichkeit nach dem Karlsruher Verständlichkeitsmodell (MERZ, 2019).

Weitere Faktoren zur Optimierung der Content-Ebene lassen sich aus den *Information Quality Dimensions* (STVILIA, GASSER, TWIDALE & SMITH, 2007) für das Information Retrieval in der Technischen Dokumentation ableiten. Hier kommen insbesondere die intrinsischen Dimensionen in Frage, da diese sich direkt aus dem Content heraus ergeben. Hervorzuheben sind die Dimensionen „Semantic Consistency" (vgl. Standardisierung/Schreibregeln) und „Informativeness/Redundancy" (vgl. Vermeidung von Redundanzen).

Tabelle 1: CoSMOS-Ebene Content

CONTENT			
Optimierung	**Methode / Modell**	**Softwaregestützte Anwendung**	**Art**
Verwendung von Terminologiearbeit, Vermeidung von Synonymen	ISO 704:2009 (ISO 704, 2009)	Terminologie-Kontrolltools, CLC[18]	Assistenz
Einsatz von Schreibregeln und Redaktionsleitfäden	Funktionsdesign® (MUTHIG, 2008) oder InformationMapping®	CLC, angepasste DTD[19]	Assistenz
Vermeidung unkontrollierter Redundanzen	u. a. das *Uncertainty Principle* (BERNSTEIN & ZOBEL, 2005; CLARKE u. a., 2008; KUHLTHAU, 2005)	Semantische Ähnlichkeitsanalyse (OEVERMANN & LÜTH, 2018)	Analyse
Einsatz von kontrollierter Sprache	Simplified Technical English (ASD-STE10, 2017)	CLC	Assistenz
Verwendung von Informationsmodellen (Mikrostrukturen)	Teile von PI-Mod (ZIEGLER & STEURER, 2010)	DTD des Informationsmodells	Assistenz
Einsatz von wiederkehrenden Textmustern (Mikrostrukturen)	z. B. ANSI Z535 (ANSI Z535.6, 2006), u. a. *Surface Text Patterns* (RAVICHANDRAN & HOVY, 2002)	Fragmente im CMS, DTD des Informationsmodells, AMS	Assistenz
Verwendung von Search Engine Optimization (SEO)	z. B. Keyword insertion (KILLORAN, 2013), „Linguistisches SEO“ (s.o.)	z. B. Web Analytics	Analyse

[18] Controlled Language Checker (CLC) prüfen neben Rechtschreibung und Grammatik auch Formulierungsmuster anhand von hinterlegten Regeln sowie Terminologie auf Basis einer Terminologiedatenbank.

[19] DTDs schreiben vor, welche (XML-)Elementstruktur einem Inhalt zugrunde liegt. Sie können durch ein Informationsmodell vorgegeben sein oder selbst entwickelt werden.

3.2.2 Struktur

Im Information Retrieval spielt die *Granularität* der durchsuchten Inhalte eine große Rolle für die Effektivität einer Abfrage (MANNING u. a., 2008, S. 20f). Aus diesem Grund ist die in der Technischen Dokumentation gängige Modularisierung von Content ein genereller Optimierungsfaktor beim semantischen Informationszugriff. Darüber hinaus bietet die inhaltliche Abgeschlossenheit von Modulen weitere Vorteile bei Präsentation der Ergebnisse bzw. deren Verarbeitung durch den Anwender.

Die Modulbildung im Content Management orientiert sich oft am verwendeten Informationsmodell (bzw. dessen Makrostrukturen) oder der Unternehmensstrategie und wird mit Hilfe von DTDs oder XML Schemas kontrolliert (*Top-Down-Modulbildung* – vgl. DREWER & ZIEGLER (2011), S. 319). Alternativ wird die Strukturierung auf Basis eines Modularisierungskonzepts individuell entwickelt. In diesem Fall orientiert sich die Modulbildung direkt am vorhandenen Content, dessen Varianten und an Vorgaben aus inhaltlichen Standardisierungsmethoden (sog: *Bottom-Up-Modulbildung* – vgl. DREWER & ZIEGLER (2011), S. 317).

Ob die richtige Modulgröße gewählt wurde, lässt sich z. B. mit Hilfe von Kennzahlen zur Wiederverwendung und Verwendung ermitteln (z. B. *Content Intelligence*, vgl. OBERLE & ZIEGLER (2012)). Bei der klassifikationsbasierten Modulbildung kann eine automatisierte Segmentierung auf Basis intrinsischer Merkmale (z. B. der Inhaltsart oder der Produktlebenszyklusphase) eine Strukturierung des Contents vornehmen (OEVERMANN, 2018b). Gliederungsvorgaben, die sich z. B. aus Normen heraus ergeben, können bei der Strukturierung von Content ebenfalls hilfreich sein, da sie Auswirkungen auf die Granularität der Modularisierung haben (ebenfalls *Top-Down*). In einem CMS werden diese Vorgaben meist mit Hilfe von Dokumentstrukturen umgesetzt, an deren Knoten dann Module zugeordnet werden.

Die Effektivität einer Suche ist auch abhängig von der Informationsqualität, bei der vor allem zwei Faktoren der Struktur entscheidend sind: „Structural Consistency" (Präsentation und Formatierung des Inhalts) und „Precision/Completeness" (Granularität der Information) (STVILIA u. a., 2007). Darüber hinaus können strukturelle Muster (engl.: „Structural Patterns")[20] beim Lesen von Inhalten nützlich sein, um innerhalb eines Dokuments den relevanten Inhalt zu finden (DI IORIO, PERONI, POGGI & VITALI, 2014).

[20] Die Idee stammt ursprünglich aus der Softwareentwicklung. Dort werden schon seit den 90er Jahren *Design Patterns* verwendet, um Teile des Quellcodes besser wiederverwendbar zu machen. (GAMMA, HELM, JOHNSON & VLISSIDES, 1995).

Tabelle 2: CoSMOS-Ebene Struktur

STRUKTUR			
Optimierung	**Methode / Modell**	**Softwaregestützte Anwendung**	**Art**
Verwendung eines Modularisierungs-konzeptes	Modularisierungsmatrix (vgl. DREWER & ZIEGLER (2011), S. 330)	Content Intelligence, REx[21], Kennzahlen (OBERLE & ZIEGLER, 2012)	Analyse
Einsatz einer klassifikationsbasierten Modularisierung	z. B. bei PI-Mod (ZIEGLER & STEURER, 2010)	Automatisierte Segmentierung (OEVERMANN, 2018b)	Automati-sierung
Einsatz einer konstruktions-orientierten Modularisierung	z. B. S1000D (S1000D, 2017)	*Common Source Data Base* (bzw. S1000D-fähige CMS wie „STAR GRIPS“ oder „CDG immedius“)	Assistenz
Verwendung einer konsistenten visuellen Strukturierung (Formatierung)	*Structural Consistency* (STVILIA u. a., 2007)	Formatierungsbasierte Segmentierung (z. B. bei FANG, TANG & GAO, 2011; NAZEMI, MURRAY & MCMEEKIN, 2014)	Automati-sierung
Verwendung von Informationsmodellen (Makrostrukturen)	Topic-Konzept von DITA (OASIS, 2010);	DTD, XML Schema	Assistenz
Abgleich mit Gliederungsvorgaben	IEC 82079-1 (IEC 82079-1, 2012); EU-Maschinenrichtlinie (2006/42/EG, 2006)	Dokumentstrukturen im CMS	Assistenz
Verwendung von strukturellen Mustern (z. B. in Dokumenten)	*Structural Patterns* (DATTOLO, DI IORIO, DUCA, FELIZIANI & VITALI, 2007)	*ToC-based segmentation* (DÉJEAN & MEUNIER, 2006) *Autom. Pattern recognition* (DI IORIO, PERONI, POGGI & VITALI, 2012)	Automati-sierung

[21] *REx* steht für „Report Exchange Format“ und definiert die technische Realisierung von Kennzahlen im Content Management (OBERLE & ZIEGLER, 2013).

3.2.3 Metadaten

Metadaten und Klassifikationen haben eine lange Tradition in der Technischen Dokumentation und wurden bisher vor allem für das interne Informationsmanagement verwendet. So ist bei der Inhaltsrecherche in einem CMS eine reine Volltextsuche nicht sinnvoll (DREWER & ZIEGLER, 2011, S. 395). Wenn Metadaten öffentlich ausgewertet werden können (z. B. als Filter in einem Content-Delivery-Portal), ist die Kombination zwischen Volltextsuche und Metadaten ebenfalls sinnvoll und führt zu besseren Ergebnissen (ZIEGLER, 2018a, S. 16). Diese Vorteile einer facettierten Suche werden auch in anderen Bereichen evaluiert (BROUGHTON, 2006; HJØRLAND, 2013; ZHENG u. a., 2013). Insbesondere standardisierte Metadatenkonzepte ermöglichen einen systematischen Informationszugriff und dienen als Hilfsmittel in der Modellierung von Klassifikationen (DREWER & ZIEGLER, 2011, S. 374).

Die automatisierte Metadatenvergabe für textuelle Inhalte ist ein relativ altes Forschungsfeld, dessen Ergebnisse meist in domänenspezifischen Anwendungen zum Einsatz kommen (ein Beispiel für *e-Learning*-Ressourcen z. B. bei CARDINAELS, MEIRE & DUVAL (2005)). Darüber hinaus existieren diverse Spezialisierungen, wie z. B. die automatisierte Klassifizierungen von Modulen im Content Management, anhand von Art oder Thema des Inhalts (OEVERMANN & ZIEGLER, 2018). Diese ML-Methoden bauen meist auf zuvor festgelegten Kategoriesystemen auf, aus denen dann Werte vergeben werden.

In der Regel müssen Metadaten- oder Klassifikationsmodelle bzw. Taxonomien von Hand erstellt werden, da diese Arbeit ein umfassendes Produkt- und Domänenwissen erfordert, das noch nicht in den Trainingsdaten steckt und deshalb für einen überwachten Lernprozess ungeeignet ist. Ansätze zur automatisierten Taxonomiebildung existieren bereits, funktionieren aber am besten in allgemeinen Wissensdomänen (HOSNY, EL-BELTAGY & ALLAM, 2015). Alternativ können unüberwachte ML-Methoden auf Basis sprachstatistischer Merkmale die Menge an Metadaten in Gruppen (sog. *Cluster*) gruppieren, die für eine Vorstrukturierung verwendet werden können.

Eine weitere Methode für die Zuweisung von semantischen Metadaten, die nicht explizit im Inhalt erwähnt werden, ist die sog. *Latent Semantic Analysis*[22] (manchmal auch *Latent Semantic Indexing*). Dabei handelt es sich um ein Verfahren, welches dazu genutzt wird, die in einem Text vorkommenden Terme automatisiert in (semantische) Überbegriffe oder Synonyme zusammenzufassen (DEERWESTER, DUMAIS, FURNAS, LANDAUER & HARSHMAN, 1990).

[22] Übersetzt etwa: „Analyse verborgener Semantik". Im Deutschen manchmal auch „latentsemantische Analyse" (LEMNITZER & ZINSMEISTER, 2006, S. 35).

Die Besonderheit der Methode besteht darin, dass dabei keine Ontologie im Hintergrund die Begriffshierarchie vorgibt, sondern dass sich die „verborgene Semantik“ über gemeinsam auftretende Terme in ähnlichen Dokumenten statistisch herleiten lässt.

Im Bereich des maschinellen Lernens existieren zahlreiche andere Methoden und Werkzeuge zur Metadatengenerierung auf der Basis von Texten. Diese basieren z. B. auf einer Entitätenerkennung[23] oder automatischer Verschlagwortung (*Auto Tagging*). Im Gegensatz zu einer automatisieren Klassifikation sind hierbei die Wertemengen prinzipiell offen und wenig systematisiert. Neben diesen Content-bezogenen Verfahren gibt es auch die Möglichkeit, Metadaten über den Kontext, die Verwendung oder die Dokumentstruktur abzuleiten (CARDINAELS u. a., 2005). Eine Vergabe von Klassifikationen basierend auf anderen Metadaten ist ebenso möglich.

Bei der Zuweisung von Metadaten[24], die bei einer Suche vom Anwender ausgewertet werden sollen, wie etwa bei Tags oder Konzepten für einen Index, ist es daher wichtig, dass die gewählten Benennungen mit denen übereinstimmen, die der Anwender auch wählen würde (siehe auch bei ZENG u. a. (2002)):

> „If both the indexer and the searcher are guided to choose the same term for the same concept, then relevant documents will be retrieved.“ (ISO 25964-1, 2011)

Generell ist bei der Metadatenmodellierung ein Use-Case-orientiertes Vorgehen zu empfehlen, das sich auf die Bedürfnisse der Zielgruppe und typische Recherchetätigkeiten stützt. Dafür kann z. B. eine Metadatenanforderungsanalyse gemacht werden (HAYNES, 2004) oder mit Hilfe von gesammelten User Stories gearbeitet werden, von denen dann die relevanten Metadaten abgeleitet werden (PARSON u. a., 2017).[25] Beim herstellerübergreifenden Austausch von Daten ist es darüber hinaus von Vorteil, wenn die verwendete Metadatensystematik samt ihrer Werte standardisiert ist und somit ohne weiteres Mapping „verstanden“ werden kann. Diese Standards existieren z. B. im Bereich der Dokumentenkategorien (VDI 2770, 2018) oder bei Produktklassifikationen (HEPP, LEUKEL & SCHMITZ, 2007).

---

[23] Im Englischen „Named Entity Recognition“. Erkennt Entitäten wie Personen oder Orte.

[24] Hier handelt es sich hauptsächlich um nicht-klassifizierende und nicht-taxonomische Metadaten, wie etwas Keywords, Tags, Indexterme oder Freitext. Die Grundprinzipien gelten aber auch für klassifizierende Metadaten.

[25] Die semantischen Konzepte von iiRDS wurden z. B. auf Basis von 59 User Stories entwickelt (PARSON, SAPARA & ZIEGLER, 2017).

Tabelle 3: CoSMOS-Ebene Metadaten

METADATEN			
Optimierung	**Methode / Modell**	**Softwaregestützte Anwendung**	**Art**
Verwendung von Klassifikations-methoden (Module)	PI-Class® (ZIEGLER, 2015)	Automatisierte Klassifizierung (OEVERMANN & ZIEGLER, 2018)	Automati-sierung
Verwendung von Klassifikations-systemen (Produkte)	u. a. e-Cl@ass (HEPP u. a., 2007)	Automatisierte Taxonomiebildung (HOSNY u. a., 2015)	Automati-sierung
Abgleich der benutzerrelevanten Metadaten	ISO 25964 (ISO 25964-1, 2011)	Knowledge Organization System	Assistenz
Verwendung von Use-Case-orientierten Metadaten	u. a. eine Metadaten-Anforderungsanalyse (HAYNES, 2004, S. 148)	Content Relevance Analytics (DORFHUBER & ZIEGLER, 2017)	Analyse
Annotation mit Tags, Indextermen und Schlagworten	Diverse Normen, z. B. DIN 31623 (DIN 31623-1, 1988)	Auto-Tagger (z. B. „IntraFind Tagging Service“)[26], Automatic indexing (LAHTINEN, 2000)	Automati-sierung
Domänen-agnostische Klassifizierung	z. B. BBC Resort-Kategorisierung	z. B. „Google Cloud Platform” (KACHKACH, 2018), AWS, „Microsoft Azure“ etc.	Automati-sierung
Annotation mit standardisierten Kategorisierungen	VDI 2770 (VDI 2770, 2018), DIN EN 61355 (DIN EN 61355-1, 2009)	Systematisches Mapping von existierender Datenbasis (IIRDS CONSORTIUM, 2018)	Automati-sierung
Regelbasierte Annotation	Domänenspezifische oder individuelle Klassifikation (z. B. Maße, Farben, etc.)	z. B. „Empolis Smart Cloud“ mit „Knowledge Packs“ [27] (ADRIAN, 2018)	Automati-sierung

[26] IntraFind Tagging Service: https://www.intrafind.de/produkte/tagging-service [28.01.19]

[27] Bei den *Knowledge Packs* handelt es sich um eine Mischform verschiedener (ML-)Techniken.

3.2.4 Ontologien und semantische Netze

Die Einordnung von Inhalten in ein semantisches Netz durch die Zuweisung eines semantischen Konzepts (*Annotation*) geht noch über die reine Metadatenvergabe (siehe vorheriger Abschnitt) hinaus, da Beziehungs- und Klassenwissen sowie Definitionen daraus abgeleitet werden können. Die Vorteile der dadurch möglichen *semantischen Suche* und *Informationsintegration* liegen auf der Hand (siehe auch Abschnitte 2.5 und 2.7). Über das Netz aus Relationen kann vorhandenes Wissen extrahiert und neue Erkenntnisse gewonnen werden. Die Kombination aus Volltextsuche und semantischen Metadaten verbessert die Ergebnisse[28] bei der Suche nach speziellen Webinhalten (SHAH, FININ, JOSHI, COST & MATFIELD, 2002). In Entscheidungssystemen kann durch semantische Netze die sinnvolle Verkettung von Informationen wesentliche Vorteile gegenüber „linearen" Metadatensystemen haben (PODDIG, 1992). Durch das Beziehungswissen entstehen beim Informationszugriff neue Darstellungs- und Interaktionsmöglichkeiten (SEELING & BECKS, 2005).

Durch die übergreifende Semantik der zugewiesenen Konzepte (die durch Definitionen sichergestellt wird) ist eine organisationsübergreifende Informationsintegration möglich, die es erlaubt, über mehrere, auch heterogene Datenquellen hinweg zu suchen. Als Basis dafür kann z. B. eine umfassende, aber oberflächliche Weltwissen-Ontologie dienen (MASCARDI, CORDÌ & ROSSO, 2007). Daneben existieren semantische Standards zum Beschreiben von Ressourcen, die oft auch als Vokabulare bezeichnet werden. Beispiele dafür sind der bibliographische Beschreibungsstandard „Dublin Core"[29] oder das im Web sehr verbreitete Metadatenmodell „Schema.org" (GUHA, BRICKLEY & MACBETH, 2016).[30] Beide beschränken sich auf die Modellierung von relativ simplen Aussagen zugunsten einer höheren Adaptionsrate. Um die Ausdrucksmöglichkeit einer Ontologie voll auszunutzen, kommen durch die benötigte Detailtiefe in der Praxis jedoch nur eng begrenzte Themenbereiche in Frage (WELLER, 2009). Diese Domänenontologien decken z. B. die speziellen Anforderungen in einer Branche oder einer Firma ab und können als Basis für die semantische Modellierung des Contents dienen (im Bereich der Technischen Dokumentation mit iiRDS). Eine Kombination mit existierenden Standards für allgemeinere Teile einer Ontologie ist aber möglich und üblich.

---

[28] In diesem Fall konnte die „Mean Average Precision" gesteigert werden (vgl. Erklärung zu Recall und Precision in Abschnitt 2.6).

[29] Teile von Dublin Core wurden auch als ISO-Norm veröffentlicht (ISO 15836-1, 2017).

[30] Google befüllt seinen „Knowledge Graph" z. B. mit den von Websites bereitgestellten Schema.org-Metadaten (GUHA, BRICKLEY & MACBETH, 2016).

Tabelle 4: CoSMOS-Ebene Ontologien und semantische Netze

ONTOLOGIEN & SEMANTISCHE NETZE			
Optimierung	**Methode / Modell**	**Softwaregestützte Anwendung**	**Art**
Annotation mit Konzepten aus Domänenontologie	iiRDS (BECKER u. a., 2018)	Teil: Autom. Annotation (Klassifizierung/Mapping) (BADER & OEVERMANN, 2017)	Automatisierung
Abgleich mit allgemeinen Wissensmodellen	Google Knowledge Graph (SINGHAL, 2012), Cyc (MASCARDI u. a., 2007)	„Google Search“ (als IR-Anwendung)	Automatisierung
Annotation mit semantischen Metadaten-standards	u. a. Dublin Core (ISO 15836-1, 2017), Schema.org (vgl. z. B. GUHA u. a., 2016)	z. B. diverse Adobe-Software (eingebettete Metadaten in PDF-Dateien[31])	Automatisierung
Einhaltung von Linked-Data-Prinzipien	Linked Data (BERNERS-LEE, 2009)	Linked Data Platform (W3C, 2015)	Assistenz
Verwendung einer Autoren-unterstützung	z. B. eine semantische Autovervollständigung (HYVÖNEN & MÄKELÄ, 2006)	Teil: Unterstützte Berichterstellung (BADER & OEVERMANN, 2017; OEVERMANN, 2018b)	Assistenz
Einsatz von strukturierten Abfragen und Auswertungen	SPARQL (W3C, 2013)	z. B. „Apache Jena“ oder „Apache Marmotta“	Analyse
Verknüpfung verschiedener Ontologien und Vokabularien	Informationsintegration (vgl. WACHE u. a. (2001)), 5-Star Linked (Open) Data (BERNERS-LEE, 2009)	Teil: Informationsintegration (BADER & OEVERMANN, 2017)	Automatisierung / Assistenz

[31] Dublin-Core-Metadaten werden z. B. in PDF/A-Dateien verwendet: https://www.pdfa.org/pdfa-metadaten-xmp-rdf-dublin-core/?lang=de [05.02.2019]

3.3 Anwendung

Die vorgestellte Übersicht kann – wir im vorherigen Abschnitt gezeigt – zur systematischen Einordnung von Methoden und Anwendungen verwendet werden. Darüber hinaus lassen sich mit den vier CoSMOS-Ebenen Reifegrade der Technischen Dokumentation beurteilen oder Optimierungspotenziale und Forschungslücken aufzeigen.

3.3.1 Optimierung und Empfehlungen

Üblicherweise beginnt die Optimierung des semantischen Zugriffs beim Content (unterste Ebene) und wird dann bei Struktur und Metadaten fortgesetzt. Der Einsatz und Nutzen von Ontologien und semantischen Netzen hängt stark vom Anwendungsfalls ab und ist in vielen Fällen nicht unbedingt notwendig. Hier kann bereits mit standardisierten Metadaten und Klassifikationssystemen ein verbessertes Information Retrieval erzielt werden.

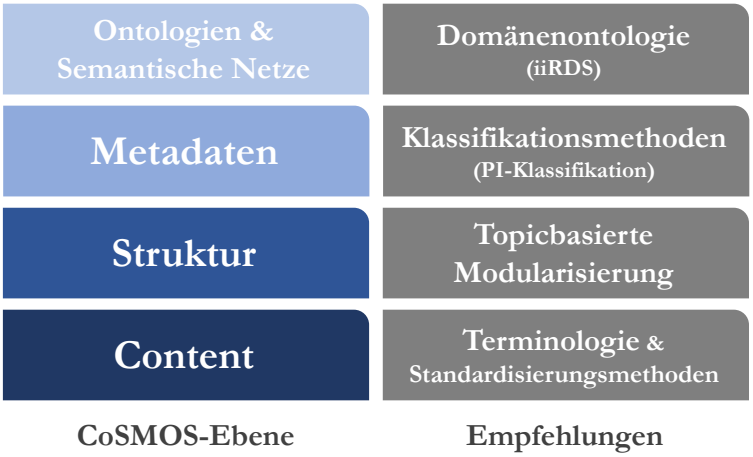

Abbildung 6: Empfohlene Optimierungen im CoSMOS-Modell.

Aus Sicht einer Technischen Redaktion kann so zunächst ein Abgleich der eingesetzten Methoden „von unten nach oben“ im Modell erfolgen. Für jede der Optimierungen kann evaluiert werden, ob die Verwendung im individuellen Kontext sinnvoll ist und ob eine softwaregestützte Anwendung in Frage kommt. Die konkrete Auswahl hängt in der Regel auch von den geplanten Anwendungsszenarien (z. B. Content-Delivery-Portal oder Mobile App) und den dort verwendeten Technologien ab.

In der Technischen Dokumentation sollten zunächst die empfohlenen Optimierungen („low hanging fruits“) über alle Ebenen hinweg umgesetzt werden, um schnell zu einem optimalen semantischen Informationszugriff zu gelangen (vgl. Abbildung 6).

Das sind beim Content eine benutzerorientierte Terminologie und standardisierte Formulierungsmuster mit Hilfe von TD-Standardisierungsmethoden und SEO, bei der Struktur eine topicbasierte Modularisierung (PI-Mod, DITA etc.), bei den Metadaten eine systematische Klassifizierung (z. B. basierend auf der PI-Klassifikation) und auf der Ontologie-Ebene die Verwendung einer Domänenontologie zur Verknüpfung und Annotation der Inhalte (z. B. iiRDS) und entsprechender Systeme.

In der Praxis empfiehlt sich deshalb neben den etablierten Standardisierungsmethoden für Inhalte und Struktur auf offene und verbreitete Standards im Bereich der semantischen Technologien zu setzen, da mit einer konsequenten Umsetzung viele Punkte für einen optimalen semantischen Informationszugriff bereits erfüllt werden.

3.3.2 Forschung

Auch wenn die gelisteten Methoden keinen Anspruch auf Vollständigkeit erhebt, können anhand der Übersichten weitere Forschungsthemen abgeleitet werden. So sind insb. in Bezug auf die Technische Dokumentation die Auswirkungen von Standardisierung und Modularisierung auf Information-Retrieval-Anwendungen noch näher zu untersuchen. Die methodischen Metadatenvergabe ist stark durch Informationsmodelle und Indexierungsverfahren geprägt. Die Auswirkungen dieser Methoden auf die Verbesserung von IR-Anwendung in der Technischen Dokumentation sind ein weiterer Ansatzpunkt für Untersuchungen.

Speziell in der Ebene „Struktur“ existiert noch wenig Literatur, die sich mit Modulen im Content Management und deren Auswirkungen auf die Effizienz von Suchanwendungen auseinandersetzt. Ebenfalls wird deutlich, dass in vielen methodischen Bereichen noch Automatisierungspotenzial steckt. Insbesondere die Content-Ebene wird hier in Zukunft von neuen technologischen Entwicklungen profitieren.

3.3.3 Reifegrade

Das CoSMOS-Modell kann auch zur qualitativen Einordnung von Technischer Dokumentation verwendet werden, um so Firmen eine Möglichkeit zu geben, die eigenen Informationsprozesse zu beurteilen. Anhand eines zuvor ermittelten Reifegrades und der angestrebten Zielebene im Modell können die Optimierungen dazwischen über die jeweils zugeordneten Methoden oder softwaregestützten Anwendungen umgesetzt werden.

Die Reifegrade für die Optimierung des semantischen Informationszugriffs orientieren sich auch an den Standardisierungsgraden von Technischer Dokumentation (KRÜGER & ZIEGLER, 2008). Neben Daten- und Zeichenformaten, welche in dieser Arbeit als gemeinsame Basis vorausgesetzt werden,

unterscheidet sich die Reihenfolge der Ebene, da bei der Optimierung für den Informationszugriff teilweise andere Kriterien wichtiger sind als bei der allgemeinen Standardisierung von Technischer Dokumentation (für Vergleich siehe Abbildung 7).

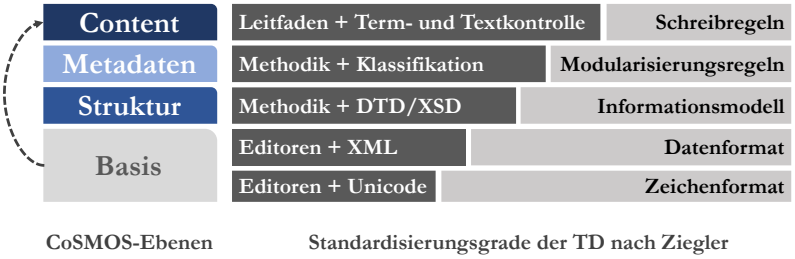

Abbildung 7: Standardisierungsgrade der TD nach Ziegler.
CoSMOS-Ebenen im Vergleich. Daten- und Zeichenformate dienen als Basis für Content im CoSMOS-Modell.
Standardisierungsebenen übernommen aus Krüger & Ziegler (2008).

Alternativ lassen sich Informationskonzepte, wie z. B. „intelligente Informationen" ein oder mehreren Ebenen zuordnen. Im klassischen Sinne sind intelligente Informationen dem Reifegrad *Content+Struktur+Metadaten* zuzuordnen. Beim Einsatz von iiRDS kommt (je nach Verwendung) die Ebene „Ontologien und semantische Netze" dazu. In dieser Arbeit wird dann von *vernetzten Informationen*[32] gesprochen (vgl. siehe Abbildung 8).

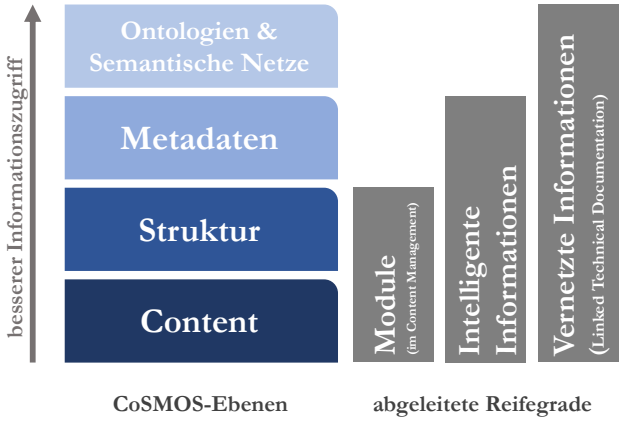

Abbildung 8: CoSMOS-Modell mit abgeleiteten Reifegraden von TD

[32] Alternativ auch „Linked Technical Documentation" nach FURTH (2018).

Ein weiterer Vergleich kann zum 5-STAR-Reifegradmodell für Technische Dokumentation gezogen werden, welches eine Sternebewertung zuweist (FURTH, 2018; FURTH & BAUMEISTER, 2015). Dort können Methoden und softwaregestützte Anwendungen angelehnt an die Qualitätskriterien von Linked Open Data (BERNERS-LEE, 2009) in fünf Kategorien beurteilt werden: Electronic Format, Structured Content, Atomic Modules, Typed Modules und Linked Information Units. Diese Kategorien decken sich sehr gut mit dem CoSMOS-Modell, wobei dieses an einigen Stellen andere Schwerpunkte setzt (z. B. Content-Qualität oder erweiterte Metadaten). Für einen Vergleich siehe Abbildung 9.

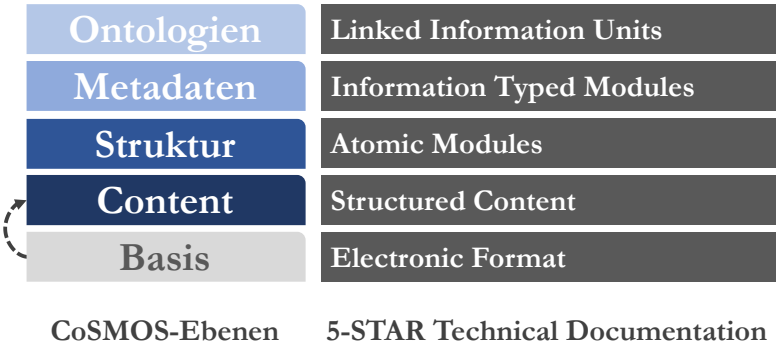

Abbildung 9: 5-STAR Technical Documentation nach Furth.
CoSMOS-Ebenen im Vergleich. Ein elektronisches Format für den Content wird im CoSMOS-Modell vorausgesetzt.
Ebenen/Kategorien übernommen aus Furth (2018).

4 Untersuchungen

4.1 Überblick

In den folgenden Abschnitten werden anhand durchgeführter Untersuchungen verschiedene Optimierungen des semantischen Informationszugriff vorgestellt. Die angewandten Methoden basieren zum größten Teil auf Verfahren des maschinellen Lernens. Als Datenbasis für die Evaluierung der Ergebnisse dienten dabei immer Echtdaten aus Technischen Redaktionen verschiedener Firmen. Forschungsziel war die domänenspezifische Anpassung von existierenden Verfahren zur Automatisierung von redaktionellen Prozessen.

4.1.1 Zusammenhang

Die Untersuchungen stehen jeweils für eine konkrete Optimierung innerhalb einer Ebene im vorgestellten CoSMOS-Modell. Die in Abbildung 10 dargestellte Zuordnung ist beispielhaft zu verstehen. In der Praxis sind die Übergänge zwischen den Ebenen fließend oder aufeinander aufbauend.

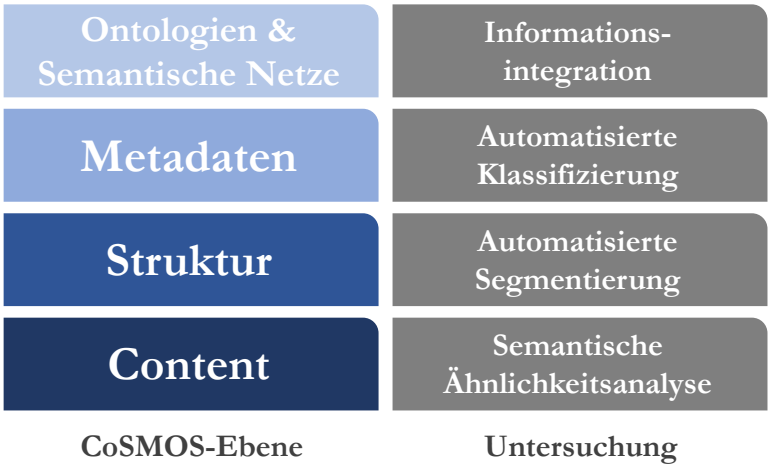

Abbildung 10: Zuordnung der Untersuchungen zu CoSMOS-Ebenen.

Da in den Untersuchungen jeweils auch angrenzende Bereiche mit abgedeckt werden und auf vorherige Arbeiten aufgebaut wird, fallen manche Teile auch in andere Ebenen. So behandelt z. B. die Untersuchung zu semantischer Segmentierung auch die Annotation mit semantischen Konzepten oder die Arbeit zur Informationsintegration auch die Zuweisung von Metadaten als Grundlage für ein weiterführendes Mapping.

4.1.2 Reihenfolge

Die hier gewählte Reihenfolge entspricht weder der chronologischen Entstehung noch dem Aufbau nach CoSMOS-Ebenen, sondern soll bei einem linearen Lesefluss dem Verständnis dienen, da die verwendeten Methoden und Techniken aufeinander aufbauen: Die automatisierte Segmentierung basiert auf den Ergebnissen einer modifizierten Klassifizierung und die Informationsintegration verwendet unter anderem die Verfahren aus Segmentierung und Klassifizierung zu einer einheitlichen semantischen Annotation und Gesamt-Evaluation. Durch die Abweichung von der chronologischen Reihenfolge wurde an manchen Stellen, an denen aktuellere Erkenntnisse existieren, entsprechende Anmerkungen zur Information eingefügt (siehe auch Abschnitt 1.5 zum Aufbau der Arbeit).

4.2 Semantische Ähnlichkeitsanalyse

Der folgende Inhalt wurde zu großen Teilen aus dem Konferenzbeitrag „Semantically Weighted Similarity Analysis for XML-based Content Components“ übernommen (OEVERMANN & LÜTH, 2018). An geeigneten Stellen wurden Querverweise gesetzt oder Anmerkungen eingefügt.

4.2.1 Zusammenfassung

Unkontrollierte Varianten und redundanter Content (Duplikate) erzeugen wiederkehrende Probleme im Content Management, wie eine sinkende Wiederverwendung, höhere Übersetzungskosten und weniger Standardisierung. Automatisierte Ähnlichkeitsanalysen können dabei helfen, bestehende Datenbestände in Content-Management-Systemen aufzuräumen und problematische Texte zu identifizieren. Konventionelle Verfahren scheitern jedoch oft an den großen Datenmengen und den in der Technischen Dokumentation häufigen (gewollten) Modulvarianten.

Eine Lösung dieses Problems ist die Verwendung eines effizienten Algorithmus auf Basis der Kosinusähnlichkeit, der semantische Informationen aus XML-basierten Informationsmodellen mit einbezieht. Zur Verifizierung des Ansatzes wurde ein browserbasierter Prototyp entwickelt, der gewollte Varianten mit hoher Performanz identifizieren kann, indem bestimmte semantische Texteigenschaften besonders gewichtet werden. Der Prototyp wurde in einem Industrieprojekt mit großem Inhaltskorpus erfolgreich getestet.[33]

4.2.2 Einleitung

Technische Dokumentation wird oft aus zahlreichen Textbausteinen, den sogenannten Modulen, zusammengestellt. Diese Textfragmente können wiederverwendet und über verschiedene Dokumente hinweg kombiniert und von unterschiedlichen Autoren erstellt werden. Hohe Wiederverwendungsraten führen zu einem effizienten Publikationsprozess, einem konsistenten Änderungsmanagement und reduzierten Übersetzungskosten (vgl. DREWER & ZIEGLER, 2011). Speziell in Multiautoren-Umgebungen erlauben sie ein paralleles Arbeiten und eine schnelle Erstellung des Inhalts.

Referenzierende Wiederverwendung und Variantenmanagement funktionieren jedoch nur dann zuverlässig, wenn sie kontrolliert sind, was bedeutet, dass Wiederverwendungen und Varianten zu ihrem ursprünglichen Inhalt zurückverfolgt werden können. Sehr ähnliche oder nahezu identische Inhalte verhindern ein eindeutiges Suchen und Referenzieren. Dies tritt insbesondere bei unkontrollierten Varianten häufig auf. Typische Beispiele sind Inhalte, die

[33] Die Ergebnisse aus diesem Industrieprojekt mit „SIEMENS Energy Sector“ wurden im Rahmen eines Tagungsbeitrages vorgestellt (OEVERMANN & FLESCHUTZ-BALAREZO, 2018).

zwar eine hohe textlicher Ähnlichkeit haben aber gleichzeitig signifikante semantische Unterschiede aufweisen, z. B. Module, die Produktvarianten mit gleicher Funktionalität aber unterschiedlichen technischen Daten beschreiben, „gespiegelte" Tätigkeitsbeschreibungen (bspw. Ein- und Ausbau) oder variable Produkt- und Markennamen. Die Zahl der unkontrollierten Varianten steht in Beziehung zur Gesamtkonsistenz des Contents und kann abhängig von externen Faktoren sein, wie der Zahl der Autoren (z. B. in einer Multiautoren-Umgebung), der typischen Lebensdauer des Content und der Anwendung eines Redaktionsleitfadens (DREWER & ZIEGLER, 2011). Automatisierte Datenmigrationen können beim Aufsplitten von Dokumenten zu Modulen weitere ungewollte Varianten verursachen, soweit diese beim Import nicht mit dem Datenbestand abgeglichen werden. In den meisten Fällen sind aber weiterhin menschliche Fehler (z. B. durch Copy/Paste oder Neuerstellung eines Inhalts statt Wiederverwendung) für Dubletten oder sehr ähnliche Inhalte verantwortlich.

Für komplexe Maschinen und Anlagen kann die Menge an zugehöriger Technischer Dokumentation von hundert bis mehrere tausend Modulen reichen, weshalb in der Regel Content-Management-Systeme zum Einsatz kommen, die Technische Redakteure bei der Erstellung, Verwaltung und Zusammenstellung der Textfragmente unterstützen. Mit Hilfe von semantischen Informationsmodellen, die oft als Dokumenttypdefinitionen (DTDs) oder XML-Schema-Definitionen (XSD) umgesetzt werden, können Autoren ihre Texte strukturieren und relevante Informationen mit den entsprechenden semantischen Funktionen bis auf die Wortebene auszeichnen. Beispiele für diese Art von Informationsmodell sind DITA (OASIS, 2010), PI-Mod (ZIEGLER & STEURER, 2010) oder S1000D (S1000D, 2017).

Die meisten existierenden Methoden zur Ähnlichkeitsanalyse werden auf reinen Text angewendet und nutzen keine Semantik der verwendeten Informationsmodelle, weshalb sie anfällig für die Identifizierung gewollter Varianten als potenzielle Duplikate sind. Andere Ansätze verwenden die Verarbeitung natürlicher Sprache (NLP, engl.: „Natural Language Processing") zum Identifizieren von semantisch wichtigen Textteilen, sind dadurch aber von komplexen und sprachabhängigen Grammatikmodellen abhängig und deshalb ungeeignet für eine Performance-kritische Verarbeitung von Daten.

In den folgenden Abschnitten wird eine effiziente Methode zur Ähnlichkeitsanalyse von großen Mengen XML-basierter Module vorgestellt, die zwischen beabsichtigten Varianten und potenziellen Duplikaten mit Hilfe von semantischer Gewichtung spezifischer Textteile unterscheiden kann. Nach der Definition einer Methodologie für das Textparsing, die Merkmalsextraktion und die Ähnlichkeitsanalyse, wird der Ansatz der semantischen Gewichtung mit

Hilfe einfacher Beispiele erläutert. Die Methode wurde in einem browserbasierten Redaktionswerkzeug implementiert, mit Hilfe dessen Technische Redakteure ihre CMS-Datenbanken bereinigen können. Der Aufbau der Implementierung und die Performance des Prototyps sowie Optimierungsansätze werden anschließend evaluiert. Die Neuheit des vorgestellten Ansatzes liegt darin, dass er sich explizit auf die typischen Modulvarianten der Technischen Dokumentation sowie deren Auswirkungen auf große Inhaltsdatenbanken in realen Anwendungsszenarien fokussiert.

4.2.3 Verwandte Arbeiten

In verwandten Arbeiten von RING, LÜTH & GLÄBE (2009) wurden bereits PDF-Dokumente auf Wortgruppen untersucht, um Beziehungen zwischen Konzepten zu finden (innerhalb der Domäne „Werkstofftechnik in Prozessketten“). Der hier vorgestellte Ansatz bezieht sich bei der Analyse allerdings auf den gesamten Text einer Einheit (bspw. einem Modul), nicht nur auf ausgewählte Konzepte daraus. Dabei werden Ergebnisse aus früheren Arbeiten zur automatisierte Klassifizierung von Inhalten auf Basis der Kosinusähnlichkeit (OEVERMANN & ZIEGLER, 2018) weiterentwickelt, um damit die textuelle Ähnlichkeit zwischen Module zu untersuchen. Weitere Vorarbeiten haben sich mit semantischem Änderungsmanagement befasst, wobei untersucht wurde, wie sich Änderungen im Dokument auf die zugrundeliegende Semantik auswirken (AUTEXIER & MÜLLER, 2010). Der dort vorgestellte Ansatz basiert auf XML-Strukturen und einer domänenspezifischen Regelsprache. Die Methode ist generisch einsetzbar, so lange sich die semantisch-syntaktische Struktur in XML-Elementen ausdrücken lässt.

Die domänenspezifische Ähnlichkeitsanalyse von Inhaltmodulen in der Technischen Dokumentation wurden mit dem Ziel untersucht, die Wiederverwendungsrate innerhalb von CMS durch Vorschläge ähnlicher Textfragmente zu erhöhen (SOTO u. a., 2015). Da gegen eine deduplizierte Datenbank verglichen wird, handelt es sich hierbei allerdings um einen $1:n$-Vergleich (nicht um einen $n:m$-Kreuzvergleich).

Die Anwendung von Ähnlichkeitsanalysen auf kurze Textfragmente wurde bereits von anderen Autoren eingehend betrachtet (LI, MCLEAN, BANDAR, O'SHEA & CROCKETT, 2006; METZLER, DUMAIS & MEEK, 2007). Obwohl die hier vorgestellte Methode ähnlichen Herausforderungen gegenübersteht (wenig Daten zur Merkmalsextraktion, fehlender Kontext etc.), unterscheidet sie sich in der Größe der untersuchten Texteinheiten (Sätze bzw. Fragen vs. Module im Content Management) und den vorhandenen Strukturinformationen (reiner Text vs. XML). Verschiedene Methoden zur Messung von semantischer Ähnlichkeit zwischen Texten wurden bereits in anderen Arbeiten untersucht und ausgewertet (MIHALCEA, CORLEY & STRAPPARAVA, 2006).

4.2.4 Methodologie und Versuchsaufbau

4.2.4.1 Überblick

Die Gesamtmethodologie besteht aus drei wesentlichen Schritten (siehe Abbildung 11): Parsing, Merkmalsextraktion und Ähnlichkeitsanalyse. Bei Parsing werden die extrahierten Texte des Gesamtmoduls und der gewichteten Textteile separat vorgehalten.

In den folgenden Schritten werden alle Merkmale des extrahierten Textes gebildet und ihrer Vorkommenshäufigkeit ermittelt. Semantisch gewichtete Merkmale werden mit einem Gewichtungsfaktor multipliziert und zur Vorkommenshäufigkeit im nicht gewichteten Gesamttext addiert. Diese künstliche Erhöhung der Vorkommenshäufigkeit beeinflusst die Ähnlichkeitsanalyse in vorhersagbarer Weise. Dieses Verhalten kann dazu genutzt werden, um unkontrollierte Varianten zu identifizieren.

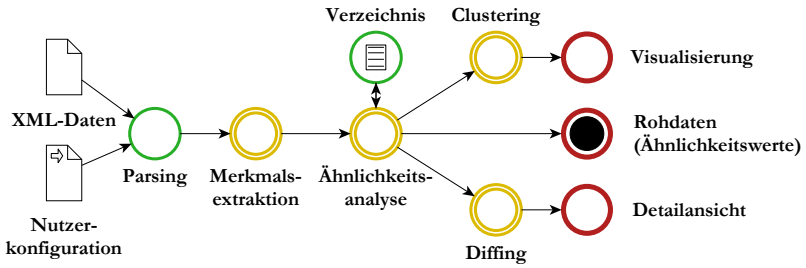

Abbildung 11: Schematischer Datenfluss bei der Ähnlichkeitsanalyse.

4.2.4.2 Parsing

Die XML-Datei wird auf Basis einer Nutzerkonfiguration eingelesen und der Text aus den XML-Elementen extrahiert, die als Module definiert wurden. XML-Attribute und ihre Werte werden nicht ausgewertet; Sonderzeichen werden entfernt. Der Text wird dann auf Basis einer einfachen Wortgrenzenerkennung in einzelne Wörter aufgeteilt (*Tokenisierung*, engl.: „Tokenizing“).

4.2.4.3 Merkmalsextraktion

Als Textmerkmale wurde eine Kombination aus Unigrammen und Bigrammen von Wörter ($n = \{1,2\}$) gewählt, basierend auf den Resultaten vorheriger Arbeiten (OEVERMANN & ZIEGLER, 2018) und der Notwendigkeit die gewichteten Unterschiede bis zur Wortebene identifizieren zu können. Die generierten n-Gramme werden zu Objekten verkettet, die jeweils ein spezifisches Modul repräsentieren. Die Häufigkeiten der generierten Merkmale werden pro Modul und für den Gesamtkorpus gezählt. Die Anzahl der Merkmale, die innerhalb der semantisch gewichteten XML-Elemente enthalten sind, wird mit einem Gewichtungsfaktor q multipliziert (hier $q = 10$) und zur bereits

ermittelten (nicht-gewichteten) Vorkommenhäufikgeit addiert. Der geeignetste Wert für q kann von den Eigenschaften des verwendeten Informationsmodells oder der Größe der Module abhängen. Somit wird ein Modul durch die Verteilung der gewichten Merkmalsvorkommen charakterisiert.

Für jedes Objekt wird ein Vektor gebildet, der alle eindeutigen Merkmale des Gesamtkorpus als Vektorkomponenten und die objektspezifischen gewichteten Vorkommenshäufigkeiten als Werte für diese Komponenten enthält (der Wert ist 0, wenn ein Merkmal nicht in einem Objekt vorhanden ist).

Durch die große Zahl eindeutiger Merkmale wird dieser Prozess in kleinere Subroutinen aufgeteilt (z. B. eine Blockgröße von 10.000 Vektorkomponenten, abhängig vom verfügbaren Arbeitsspeicher). Die Ausrichtung der Vektoren kann nun in einem Vektorraummodell durch Einsatz der Kosinusähnlichkeit miteinander verglichen werden.

4.2.4.4 Ähnlichkeitsanalyse

Die Ähnlichkeit zwischen zwei Modulen ist symmetrisch, muss aber für jede mögliche Kombination von Modulen berechnet werden ($n{:}\,m$-Vergleich). Deshalb kann die Gesamtzahl der zu berechnenden Kombinationen C für n Objekte wie folgt berechnet werden:

$$|\mathrm{C}| = \frac{\mathrm{n} * (\mathrm{n} - 1)}{2}$$

Für jede Kombination wird der Kosinus des Winkels φ der beiden Vektoren $\vec{a}$ und $\vec{b}$ als Ähnlichkeitswert s verwendet (wobei $\vec{a}$ und $\vec{b}$ zwei zufällige Vektoren sind, die nach der zuvor beschriebenen Methode gebildet wurden):

$$\mathrm{s} = \cos(\varphi) = \frac{\vec{\mathrm{a}} \circ \vec{\mathrm{b}}}{|\vec{\mathrm{a}}| \cdot |\vec{\mathrm{b}}|}$$

Die resultierenden Werte, die unter einem zuvor definierten Ähnlichkeitsgrenzwert r liegen, werden aus Gründen der Speichereffizienz verworfen ($s < r$, im Standard $r = 0{,}9$). Der gewählte Wert von r beeinflusst direkt die Anzahl der möglichen unkontrollierten Varianten, die ein Technischer Redakteur prüfen muss. In den meisten realen Anwendungsszenarien wird ein Wert zwischen $0{,}85$ und $0{,}95$ für r gewählt.

Da die semantische Gewichtung durch eine künstliche Erhöhung der Vorkommenshäufigkeit realisiert wurde, erhöht die Methode nicht die Komplexität (und somit die Ausführungszeit) des Algorithmus.

4.2.5 Semantische Gewichtung

Die Methode der semantischen Gewichtung, die im Folgenden präsentiert wird, basiert auf der zugrundeliegenden Struktur des jeweiligen XML-Informationsmodells. Das bedeutet, dass Textteile, die eine spezielle semantische Bedeutung haben (z. B. Technische Daten oder Eigennamen) mit entsprechenden XML-Elementen ausgezeichnet wurden. Zwei beispielhafte[34] Textausschnitte im PI-Mod-Informationsmodell (ZIEGLER & STEURER, 2010), die eine hohe textuelle Ähnlichkeit, aber signifikante semantische Unterschiede zueinander haben, werden im Folgenden gezeigt:

```
<paragraph nodeid="a">This device is designed to
work with a voltage of <inlinedata><si-value>
<number>110</number> <unit>V</unit></si-value>
</inlinedata> only.</paragraph>
```

Codebeispiel 1: (a) – Erste Formulierungsvariante mit 110 V (Ausschnitt).

```
<paragraph nodeid="b">This device is designed to
work with a voltage of <inlinedata><si-value>
<number>220</number> <unit>V</unit></si-value>
</inlinedata> only.</paragraph>
```

Codebeispiel 2: (b) – Erste Formulierungsvariante mit 220 V (Ausschnitt).

Wendet man die zuvor beschriebene Methodologie an, um den Ähnlichkeitswert mit (s_w) und ohne (s) Anwendung der semantischen Gewichtung ($q = 10,\ r = 0{,}9$) auf das XML-Element **<si-value>** zu berechnen, erhält man: $s(a,b) = 0{,}9021$ vs. $s_w(a,b) = 0{,}4471$. Ohne semantische Gewichtung erzeugt die Merkmalsextraktion ($n = \{1,2\}$) bei den relevanten Bereichen die folgenden Merkmale bzw. Vorkommenshäufigkeiten (vgl. Codebeispiel 1 und 2, extrahierte Merkmale hier in eckigen Klammern):

```
[of, 110] : 1; [110, V] : 1; [V, only] : 1; [110] : 1; [V] : 1
[of, 220] : 1; [220, V] : 1; [V, only] : 1; [220] : 1; [V] : 1
```

Wendet man den semantischen Gewichtungsfaktor ($q = 10$) auf den Text innerhalb von **<si-value>** an (gewichtete Vorkommen werden zu ungewichteten addiert), erhält man:

```
[of, 110] : 1; [110, V] : 11; [V, only] : 1; [110] : 11; [V] : 11
[of, 220] : 1; [220, V] : 11; [V, only] : 1; [220] : 11; [V] : 11
```

[34] Die XML-Dateien mit den hier verwendeten Beispielen können im unten angegebenen Repository abgerufen werden.Aufgrund der Vertraulichkeit der Inhalte können die Echtdaten, die zur Bewertung der Effizienz des Algorithmus verwendet wurden, nicht zur Verfügung gestellt werden (vgl. Tabelle 5).

Während ein Vergleich ohne semantische Gewichtung zu einer falsch-positiven Identifikation von Duplikaten ($s(a,b) > r$) führen kann, ergibt die gewichtete Berechnung ein geringeres Ähnlichkeitsmaß, welches in diesem Fall auf eine Nicht-Gleichheit der Module hindeutet.

Dieses Verhalten wird durch die höhere Vorkommenshäufigkeit für manche Merkmale verursacht, welche die Ausrichtung der Vektoren, die die Module repräsentieren, leicht verändert. Dies führt letztendlich zu einem geringeren Ähnlichkeitsmaß, da der Winkel zwischen den Vektoren größer und der Kosinus damit kleiner wird.

Das Verfahren hat auch Vorteile bei der Ähnlichkeitsanalyse zwischen alternativen Formulierungen. Zwei zusätzliche Beispiele mit leicht abweichendem Wortlaut:

```
<paragraph nodeid="c">This device works with a voltage of
<inlinedata><si-value><number>110</number> <unit>V</unit>
</si-value></inlinedata> only.</paragraph>
```

Codebeispiel 3: (c) – Zweite Formulierungsvariante mit 110 V (Ausschnitt).

```
<paragraph nodeid="d">This device works with a voltage of
<inlinedata><si-value><number>110</number> <unit>V</unit>
</si-value></inlinedata> only.</paragraph>
```

Codebeispiel 4: (d) – Zweite Formulierungsvariante mit 220 V (Ausschnitt).

Berechnet man die Ähnlichkeit zwischen den Formulierungsvarianten mit denselben gewichteten Daten, erhält man $s(b,d) = 0{,}7455$ vs. $s_w(b,d) = 0{,}9844$, während die gleiche Formulierung mit unterschiedlichen gewichteten Daten folgendes ergibt: $s(c,d) = 0{,}85$ vs. $s_w(c,d) = 0{,}3605$. Dieses Beispiel zeigt, dass die Methode auch die Ähnlichkeit zwischen Modulen mit den gleichen gewichteten Textteilen erhöhen kann.

4.2.6 Implementierung

Der Prototyp wurde als clientseitige browserbasierte Web-Anwendung in JavaScript implementiert und ist online verfügbar.[35] Der Quellcode und Beispieldateien sind in einem öffentlichen Repository verfügbar.[36] Die Implementierung basiert auf einem Klassifizierungsframework, das im Rahmen einer früheren Arbeit entstanden ist (OEVERMANN & ZIEGLER, 2018).

[35] http://semsim.fastclass.de/ [01.02.19]

[36] https://github.com/j-oe/semsim [01.02.19]

Die Dateiverarbeitung und die grundlegende Benutzerschnittstelle konnten mit kleinen Anpassungen übernommen werden, ebenso die Softwaremodule zur Vektorrechnung. Die Verarbeitung der Ähnlichkeitsanalysen und deren Auswertung sowie Visualisierung wurden im Rahmen dieser Arbeit entwickelt und werden komplett clientseitig im Browser des Anwenders ausgeführt.

4.2.6.1 Datenfluss

Das Programm kann XML-Dateien verarbeiten, die die beiden folgenden Voraussetzungen erfüllen:

1. Alle Module sind von Elementen desselben Namens umschlossen, die direkte XML-Nachfahren des Wurzelelements sind (im Beispiel: **`<paragraph>`**)
2. Diese Elemente haben ein XML-Attribut mit einem eindeutigen Identifikator als Wert (im Beispiel: **`nodeid`**).

Für die semantische Gewichtung können alle Elemente verwendet werden, die innerhalb der in (**1**) definierten Elemente enthalten sind (im Beispiel: **`<si-value>`**). Diese Parameter können ad-hoc in einem dynamischen Eingabeformat festgelegt werden. Der Datenfluss innerhalb der Anwendung folgt der beschriebenen Methodologie (siehe Abbildung 11). Um die Symmetrie der Ähnlichkeitsbeziehungen zu berücksichtigen, wird im Hintergrund ein Verzeichnis der bekannten Kombinationen aufgebaut, das vor einer neuen Berechnung geprüft wird. Nach der Berechnung aller Resultate[37] gruppiert die Anwendung die Module, die ähnlich zueinander sind (und über dem definierten Grenzwert r liegen) und initialisiert die Visualisierung.

4.2.6.2 Effizienz der Berechnung

Der Prozess, der die Ähnlichkeitsanalyse durchführt, ist in einem eigenen Web-Worker-Thread[38] isoliert, so dass der Hauptthread durch die rechenintensiven Vorgänge nicht beeinträchtigt wird. In den durchgeführten Experimenten konnten mehrere reale Datensätze auf einem handelsüblichen Laptop[39] mit einer durchschnittlichen Berechnungszeit von 0,198 ms pro Kombination analysiert werden. Die Größe der verarbeiteten Datensätze lag dabei zwischen 166 und 4.101 Modulen bei einer durchschnittlichen Wortmenge von ca. 317 Wörter pro Modul (siehe Tabelle 5).

---

[37] Ein Resultat ist definiert als die Kombination aus zwei Modul-IDs und deren zugeordneter Ähnlichkeitswert s bzw. s_w.

[38] vgl. https://developer.mozilla.org/en-US/docs/Web/API/Web_Workers_API [05.02.19]

[39] Intel i7 2.6 GHz, 16 GB RAM, Betriebssystem: Windows 10 64bit, Browser: Google Chrome

Tabelle 5: Tests zur Effizienz der Ähnlichkeitsanalyse.
Berechnung durchgeführt mit Echtdaten (Datensätze A-D).

Datensatz	Module (n)	Kombinationen ($\lvert C \rvert$)	$\frac{\text{Wörter}}{\text{Modul}}$	Zeit t [s]	$\frac{t}{\lvert C \rvert}$ $[ms]$
A	166	13.695	455,8	0,7	0,052
B	1.600	1.279.200	178,0	243,7	0,191
C	2.501	3.126.250	353,4	650,7	0,208
D	4.101	8.407.050	278,9	2.878,0	0,342

4.2.6.3 Visualisierung

Die Darstellung der Ergebnisse wurde mit Technischen Redakteuren, die mit der Software arbeiten, diskutiert und optimiert. Die bevorzugte Arbeitsweise besteht darin, zunächst eine Gruppe von ähnlichen Modulen auszuwählen (farbig hervorgehoben in Abbildung 12, Mitte) und dann im Detail jede Modulkombination dieser Gruppe durchzuarbeiten, um zu bestimmen, welches das führende Modul einer Gruppe ist (z. B. das Modul, das am ehesten den redaktionellen Vorgaben entspricht).

Um diese Arbeitsweise zu erleichtern, wurde ein etablierter Text-Diffing-Algorithmus[40] integriert, der Unterschiede zwischen Texten auf Wortebene anzeigt, sobald eine entsprechende „Vergleiche"-Aktion vom Redakteur aufgerufen wird.[41] Diese Funktionalität unterstütz die Anwender bei der schnellen Beurteilung der Ähnlichkeiten.

Die Benutzeroberfläche (siehe Abbildung 12) liefert den Benutzern somit allgemeine Informationen über die Daten (oben), die Modulgruppen (mittig) und einzelne Kombinationen (unten).

[40] https://github.com/kpdecker/jsdiff [01.02.19]

[41] Das Diffing erfolgt erst im Moment der Vergleichsanforderung, um die die Perfomanz der Gesamtanalyse nicht zu beeinflussen.

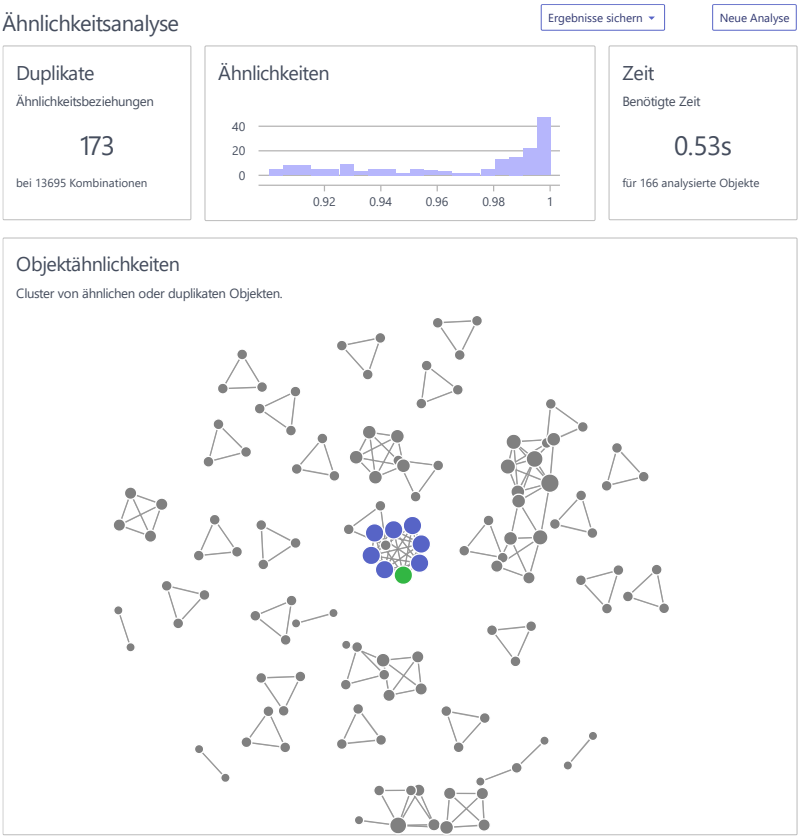

Abbildung 12: Screenshot der Benutzeroberfläche der Ähnlichkeitsanalyse. Mit einer selektierten Gruppe von ähnlichen Modulen (Bildausschnitt, Datensatz A, $q = 10$ mit Gewichtung von **`<si-value>`**, $r = 0{,}9$*). Screenshot wurde angepasst für Bildausschnitt.*

4.2.7 Ausblick

In einem nächsten Schritt soll eine ausführliche Evaluierung der vorgestellten Methode durchgeführt werden, die auch alternative, auf dem Vektorraummodell basierende Ähnlichkeitsmaße miteinbezieht. Inwieweit verschiede Informationsmodelle die Ergebnisse beeinflussen oder eine besondere Eignung haben, ist ebenfalls Gegenstand zukünftiger Forschung.

Für Inhalte, die nicht in semantischen Informationsmodellen erfasst wurden, soll eine Vorverarbeitung die entsprechenden Auszeichnungen z. B. über eine Entitätenerkennung (NER, engl.: „Named Entity Recognition“) oder *Reguläre Ausdrücke* (RegEx, engl.: „Regular Expressions“) erzeugen und in XML abbilden. Dies kann bei regelmäßigen und gleichförmigen Vorkommen, wie z. B. die Angabe technischer Daten, Eigen- oder Markennamen und Negierungen, zu guten Ergebnissen führen.

4.2.8 Fazit

Die beschriebene Methode stellt eine effiziente Ähnlichkeitsanalyse für Module vor, die beabsichtigte und unkontrollierte Varianten der Technischen Dokumentation berücksichtigt. Die vorgestellte Lösung bietet klare Vorteile gegenüber nicht-semantischen Methoden und kann durch die eine prototypische Implementierung validiert werden.

Nach der Definition einer Methodologie und des Testaufbaus wurde das Verfahren zur semantischen Gewichtung mit Hilfe von XML-Beispielen erklärt. Anschließend wurde die Implementierung vorgestellt und die Effizienz des Algorithmus evaluiert.

Wie mit Hilfe von simplen Beispielen gezeigt werden konnte, ist eine semantische Gewichtung von spezifischen Textteilen unabdingbar für eine Ähnlichkeitsanalyse Technischer Dokumentation. Durch eine entsprechende Gewichtung können falsch-positive Duplikate erkannt (vgl. Codebeispiel 1 und 2) und die Ähnlichkeit zwischen Formulierungsvarianten erhöht werden (vgl. Codebeispiel 3 und 4).

Trotz der geringen Komplexität der Methode im Vergleich zu anderen Ähnlichkeitsanalysen, die auch Wortähnlichkeiten oder externes Wissen mit einbeziehen, kann sie ein performanter und einfacher Weg sein, um die Ergebnisse einer Duplikatserkennung in großen XML-Datenbeständen von Technischer Dokumentation zu verbessern, sowie als Basis für künftige Forschung dienen.

4.3 Automatisierte Klassifizierung

Der folgende Inhalt wurde zu großen Teilen aus dem Artikel „Automated Classification of Content Components in Technical Communication“ übernommen (OEVERMANN & ZIEGLER, 2018), der wiederum auf den Vorarbeiten in OEVERMANN & ZIEGLER (2016) basiert. An geeigneten Stellen wurden im Sinne der Gesamtarbeit Querverweise, Anmerkungen oder Ergänzungen eingefügt und die Formatierung angepasst.

4.3.1 Zusammenfassung

Automatisierte Klassifizierung ist in den meisten Fällen nicht auf spezielle Domänen angepasst, da es an geeigneten Daten für Training und Validierung mangelt und genaue Charakterisierungen der domänenspezifischen Eigenschaften sowie deren Auswirkungen auf den Klassifizierungsprozess fehlen. In dieser Untersuchung wird ein Ansatz für die automatisierte Multiklassen-Klassifizierung von Modulen in der Technischen Dokumentation vorgestellt, die auf dem Vektorraummodell basiert. Es wird gezeigt, dass Unterschiede in Form und Inhalt der Module eine Anpassung von dokumentbasierten Klassifizierungsmethoden erfordern, deren Auswirkungen anhand von mehreren realen Datensätzen in zwei Sprachen validiert werden.

Im Folgenden werden allgemeine Anpassungen an der Merkmalsauswahl und Token-Gewichtung vorgeschlagen sowie ein Ansatz zur Konfidenzbestimmung des Klassifikators und zur semantischen Gewichtung von XML-basierten Trainingsdaten beschrieben. Verschiedene potenzielle Anwendungen der Methode werden erörtert und eine prototypische Implementierung zur Verfügung gestellt.

Als Ergebnis kann ein dediziertes Vorgehensmodell für die automatisierte Klassifizierung von Modulen in der Technischen Dokumentation präsentiert werden, das andere dokumentbasierte oder domänenunabhängige Ansätze in der Genauigkeit der vorhergesagten Klassifikation übertrifft.

4.3.2 Einleitung

Große und komplexe Dokumente werden innerhalb der Technischen Dokumentation in der Regel aus kleineren Bausteinen, den sogenannten Modulen[42], zusammengestellt (ANDERSEN, 2011). Dadurch werden referenzierende Wiederverwendung über Dokumente hinweg und kosteneffiziente Übersetzungen ermöglicht (SOTO u. a., 2015). Beispiele für diese Dokumentarten sind alle Arten Technischer Informationen (z. B. Handbücher, Anleitungen, Service-

[42] In der Literatur oder Industrieanwendungen auch abweichend „Topics“, „Textbausteine“, „Inhaltsknoten“ oder speziell im Englischen „Content Components“ genannt (DREWER & ZIEGLER, 2011; ROCKLEY, KOSTUR & MANNING, 2003).

berichte, Lernmaterial) aber auch Normen, Patente oder Lastenhefte. Module können – müssen aber nicht – Absätzen oder Abschnitten von Dokumenten entsprechen und sind in den meisten Fällen inhaltlich in sich abgeschlossen.

Content-Management-Systeme (CMS, synonym auch „Redaktionssysteme") sind ein weit verbreiteter Weg diese Module zu erstellen, zu verwalten und zusammenzustellen, speziell für die Erstellung von Multiautoren-Dokumenten (GRAHLMANN u. a., 2010). In den meisten Fällen wird der Inhalt in einem semantisch strukturierten XML-basierten Informationsmodell[43] geschrieben und gespeichert (DI IORIO u. a., 2012). Dabei werden die Inhalte in der Regel manuell mit Metadaten, wie Klassen aus einem zuvor definierten Klassifikationssystem, ausgezeichnet, um diese bei *Retrieval* und *Delivery* über bestimmte Kriterien identifizieren zu können (DREWER & ZIEGLER, 2011). Moderne CMS können anhand von Klassifikationen auch Informationsprodukte (wie z. B. gedruckte Handbücher) automatisiert aus Modulen und einer entsprechenden Dokumentstruktur zusammenstellen. Informationsmodelle können entweder durch das CMS vorgegeben werden, Eigenentwicklungen des anwendenden Unternehmens oder standardisiert sein (wie z. B. DocBook (OASIS, 2008) oder PI-Mod (ZIEGLER & STEURER, 2010)).

Für die Technische Dokumentation ist die Methode der taxonomischen PI-Klassifikation ein etablierter und verbreiteter Ansatz, Module in CMS zu klassifizieren (ZIEGLER & BEIER, 2014). Die Zuweisung von Klassen erfolgt üblicherweise durch Technische Redakteure direkt beim Erfassen des Inhalts und basiert auf der Erfahrung des Autors und redaktionellen Vorgaben. Die manuelle Klassifizierung großer Mengen von Inhalten (z. B. bei der Migration von Bestandsdaten) ist jedoch ein zeitaufwändiger und fehleranfälliger Prozess. Nach aktuellem Kenntnisstand gibt es derzeit keine spezialisierten Werkzeuge oder Methoden zur Automatisierung dieser Aufgabe, die sich auf die Eigenschaften von Modulen in der Technischen Dokumentation konzentrieren. Dies steht im Gegensatz zu den wachsenden Anforderungen der Industrie, die auf eine gezielte und zuverlässige Informationsbereitstellung, z. B. für Servicetechniker oder technisches Personal, setzen. Klassifizierende Metadaten werden verwendet, um Module in dynamische Szenarien zu integrieren, in denen Informationen automatisch aggregiert oder gefiltert werden oder dienen als Merkmale in einer facettierten Suche (BROUGHTON, 2006; ZHENG u. a., 2013). Dies spiegelt sich auch in der steigenden Beliebtheit von Content Delivery Portalen (CDP) wider, die Anwendern einen metadatenbasierten Zugriff auf modulare Informationen ermöglichen (ZIEGLER & BEIER, 2014).

[43] XML-basierte Informationsmodelle werden oft synonym mit ihren technischen Umsetzungen bezeichnet: DTDs oder Schemas (XSD).

Klassifizierende Metadaten können auch als Grundlage für moderne oder zukünftige Methoden des Informationszugriffs dienen, bei denen die Filterkriterien durch QR-Codes, RFID-Chips oder den Standort des Benutzers automatisch gesetzt werden. Auf Grund der großen Menge an Bestandsinhalten in Unternehmen ist die automatisierte Klassifizierung dringend notwendig, um diese intelligenten Informationsservices bereitstellen zu können (vgl. OEVERMANN, 2017a). Ein kürzlich vorgestellter Standard zum Austausch von digitaler Technischer Dokumentation implementiert diese Ideen mit Hilfe von Linked-Data-Technologien (TEKOM E.V., 2017a).[44]

Klassifizierung auf Basis des Vektorraumodells (VSM) ist eine etablierte (MANNING & SCHÜTZE, 1999; SEBASTIANI, 2002) und effiziente (LE & MIKOLOV, 2014) Methode zur Automatisierung dieser Aufgaben. Jedoch sind die meisten Anwendungen hinsichtlich Merkmalsextraktion und -gewichtung darauf optimiert, ganze Dokumente zu klassifizieren und nicht nur (kleine) Teile daraus. Darüber hinaus konzentrieren sich die meisten Implementierungen nur auf die Verarbeitung reinen Textes und erkennen keine semantischen Strukturen (DI IORIO u. a., 2014), z. B. in XML-basierten Trainingsdaten, die im Content Management weit verbreitet sind und zusätzliche Metainformationen über den Inhalt enthalten können.

Im Folgenden wird ein Ansatz vorgestellt, der diese Besonderheiten von Modulen in der Technischen Dokumentation berücksichtigt und die Vektorraum-basierte Klassifizierung dahingehend optimiert, dass eine höhere Genauigkeit bei diesen spezialisierten Aufgaben erzielt werden kann. Bestandteil der Forschung sind eine neue Gewichtungsmethode und optimierte Merkmalsextraktion für Module sowie erste Ansätze für ein verlässliches Konfidenzmaß und Ideen zur semantischen Gewichtung von XML-Elementen mit spezieller semantischer Textfunktion. Darüber hinaus werden Zusammenhänge hinsichtlich der gewählten Klassifizierungsmethode, der Anzahl und Art der verwendeten Klassen sowie Sprache und Größe der Module untersucht. Die durchgeführten Versuche dienen als Basis für weitere Forschung in dieser domänenspezifischen Anwendung des Maschinellen Lernens.

4.3.3 Verwandte Arbeiten

Die Text-Charakteristiken von Modulen im Content Management wurden u. A. von ANDERSEN (2011), BAILIE & HUSET (2015), DREWER & ZIEGLER (2011), GRAHLMANN u. a. (2010) sowie ROCKLEY, KOSTUR & MANNING (2003) diskutiert.

[44] Mitllerweile ist die Version 1.0 des hier referenzierten Standards iiRDS veröffentlicht worden (vgl. BECKER u. a., 2018).

Effektive Methoden zur Merkmalsextraktion bei der Textklassifizierung wurden von BIRICIK, DIRI & SÖNMEZ (2012) und BIRICIK, DIRI & SÖNMEZ (2009) betrachtet. Die Autoren stellen eine neue Methode namens „Abstract Feature Extraction" vor, die durch $TF - IDF$ und $TF - ICF$ motiviert ist und die erzielte Genauigkeit über verschiedenen Klassifikatoren hinweg deutlich verbessert. Die $TF - IDF - CF$-Methode, auf der die in dieser Arbeit vorgestellte Merkmalsgewichtung aufgebaut ist, wurde zuerst von LIU & YANG (2012) eingeführt und getestet. Alternative Gewichtungsmethoden werden von KO (2012) und LAN, TAN, LOW & SUNG (2005) diskutiert und verglichen.

Domänenspezifische Ansätze zur automatisierten Klassifizierung werden von GOLUB (2006) mit dem Fokus auf Webdokumente diskutiert. Der Autor sieht in dem Mangel an verfügbaren Daten- bzw. Dokumentensammlungen einen der Gründe für die fehlende Klassifikationsforschung in bestimmten Domänen. Ein Beispiel für einen detaillierten domänenzentrierten Ansatz ist die Arbeit von CALDAS, SOIBELMAN & HAN (2002). Die Autoren analysieren verschiedene automatisierte Klassifikationsverfahren und ihre Anwendungsmöglichkeiten für den Bereich der Projektunterlagen im Bauwesen. Ähnlichkeiten zwischen dieser Arbeit und der hier vorgestellten sind die Verfügbarkeit vordefinierter Klassifikationsmodelle und der Fokus auf die Prozessautomatisierung der Klassifizierungsaufgabe.

Information Retrieval für XML-basierte Dokumente wird in der Literatur ausführlich behandelt. In der Arbeit von LALMAS (2009) werden verschiedene Ranking-Methoden für Elemente in XML-basierten Dokumenten diskutiert. Der Autor kommt zu dem Schluss, dass der hierarchische Kontext (z. B. Eltern- und Kind-Elemente) vor allem beim Vergleich von Bewertungsstrategien für Ergebnisse beim Ranking von XML-Elementen wichtig ist. Ein ähnlicher Ansatz mit Klassenhierarchien wird in dieser Arbeit in Abschnitt 4.3.10.3 beschrieben. KOTSAKIS (2002) stellt einen Ranking-Ansatz vor, der $TF - IDF$ mit einem Koeffizienten für die Positionierung der Strukturelemente kombiniert. Nachteile dieser Methode sind, dass für jeden möglichen Pfad im Korpus ein Koeffizient gewählt werden muss. Dieser Ansatz war für die hier verwendeten Testdatensätze nicht geeignet.

DI IORIO u. a. (2014) beschreiben, wie textuelle XML-Dokumente auf mehreren Strukturmustern basieren, die unterschiedliche Relevanzstufen haben können (z. B. „junk structures" vs. „representational structures"). Ihre Schlussfolgerungen bestätigen die hier vorgebrachte Hypothese, dass eine Gewichtung von einzelnen Elementen in XML-basierten Inhalten sinnvoll sein kann (insbesondere bei sogenannten „surrogate elements").

Untersuchungen zur Anwendung von Methoden des Maschinellen Lernens und Ähnlichkeitsmaßen im Bereich der Technischen Dokumentation wurden von SOTO u. a. (2015) durchgeführt. Die Arbeit beschreibt Methoden zur Unterstützung von Technischen Redakteuren bei der Wiederverwendung von Modulen mit Hilfe von automatisierten Ähnlichkeitsanalysen. Die dort verwendete Methode könnte mit der hier vorgestellten kombiniert werden, um zu überprüfen, ob den zur Wiederverwendung identifizierten Modulen auch die passende Klassifikation zugeordnet wurde.

4.3.4 Methodologie

Nachdem wichtige Eigenschaften von Modulen basierend auf Best Practices der Industrie und relevanten Normen definiert wurden, werden Schlussfolgerungen zu Auswirkungen auf VSM-basierte Klassifizierungsaufgaben gezogen. Die sich daraus ergebenden Anpassungen am Klassifizierungsprozess werden anschließend mit elf Versuchen auf Basis von vier realen Datensätzen mit insgesamt ca. 7.000 manuell klassifizierten Module in zwei Sprachen validiert.

4.3.4.1 Abgrenzung

Obwohl die Versuche in dieser Arbeit ausschließlich mit Inhalten aus der Technischen Dokumentation durchgeführt wurden, sind die daraus resultierenden Ergebnisse und Schlussfolgerungen auf weitere Dokumentarten anwendbar, die den in Abschnitt 4.3.5 definierten Merkmalen entsprechen, wie beispielsweise Normen, Patente oder modularisierte Dokumente aus der Unternehmenswelt (z. B. standardisierte Lastenhefte oder Ausschreibungen).

4.3.4.2 Testdaten

Die Testdaten bestehen aus unterschiedlichen Arten von technischen Informationen und wurden von Unternehmen aus verschiedenen technischen Bereichen für Forschungszwecke zur Verfügung gestellt (siehe Tabelle 6). Aufgrund der vertraulichen Inhalte können die Datensätze nicht öffentlich zugänglich gemacht werden. Die Inhaltssprache ist entweder Deutsch oder Englisch. Ein Datensatz (D) enthält etwa 80 Module, die in beiden Sprachen verfügbar sind. Alle Inhalte wurden von Experten (Technischen Redakteuren oder Technikern) nach redaktionellen Standards verfasst, so dass die verfügbaren Daten einem gewissen redaktionellen Qualitätsstandard entsprechen (siehe auch Abschnitt 4.3.5.6). Im Vergleich zur vorherigen Untersuchungen (OEVERMANN & ZIEGLER, 2016) konnte der Gesamtkorpus von Modulen weiter vergrößert, eine weitere Sprache hinzugefügt und ein neuer Satz von Echtdaten aus Unternehmen integriert werden.

Tabelle 6: Datensätze für Trainings- und Testdaten (autom. Klassifizierung).

Datensatz	Info.-modell[45]	Sprache	Module	$\frac{\text{Wörter}}{\text{Modul}}$	Klassifikationsmodell	Anzahl Klassen
A	System	en-US	1.087	515	**Informationsklasse** – 2-stufig	10 / 26
B	Standard	de-DE	4.186	87	**Informationsklasse** – 2-stufig	6 / 22
					Produktklasse – 1-stufig	28
C	Unternehmen	de-DE	663	180	**Produktklasse** – 1-stufig	11
					Produktklasse – 1-stufig	22
D	Standard	de-DE	584	51	**Informationsklasse** – 2-stufig	8 / 14
		en-GB	1.070	57	**Informationsklasse** – 2-stufig	8 / 17

Alle Datensätze sind XML-basiert und wurden auf Grundlage der PI-Klassifikationsmethode mit taxonomischen Informationsklassen (intrinsisch) ausgezeichnet. Zwei der Datensätze haben zusätzlich vergebene Produktklassen. Die Anzahl der Klassen und die durchschnittliche Größe der Module unterscheiden sich von Unternehmen zu Unternehmen (siehe auch 4.3.5.2).

Als Beispiel für typische Klassen werden exemplarisch die in Datensatz *C* vergebenen Klassifikationen aufgelistet (jeweils 10 Beispiele):

Intrinsische Informationsklassen (Informationstyp)
Wartung, Ausbau und Einbau, Betrieb, Diagnose, Notbetrieb, Produktbeschreibung, Sicherheit, Störungsbeseitigung, Transport, Vorwort/Einleitung

Intrinsische Produktklassen (Produkttyp bzw. Funktionsbaugruppe)
Antriebsgruppe, Arbeitsausrüstung, Arbeitshydraulik, Ausstattungen/Optionen, Bremsanlage, Elektrische Anlage, Hydraulikkomponenten, Kühlanlage, Lenkanlage, Schmieranlage

[45] Das XML-basierte Informationsmodell, in dem der Inhalt erfasst wurde. *System*: natives Informationsmodell des Redaktionssystems; *Standard*: standardisierte Open-Source-Informationsmodell; *Unternehmen*: Inhouse-Entwicklung des einsetzenden Unternehmens.

Die Datensätze B und D basieren auf dem Open-Source-Informationsmodell PI-Mod (ZIEGLER & STEURER, 2010), Datensatz A wurde in einer CMS-spezifischen Variante von semantischem HTML erfasst und Datensatz C in einem eigenentwickelten Informationsmodell, das innerhalb des Unternehmen verwendet wird.

4.3.4.3 Vorverarbeitung

In einer Vorverarbeitung wird der gesamte Text aus den konfigurierten Modulen extrahiert und unnötiger Leerraum, Ziffern, Sonderzeichen und Satzzeichen entfernt. Merkmale werden als Kombination aus Einzelwörtern und Wortgruppen gebildet (wie in Abschnitt 4.3.6.1 beschrieben) und dann mit der in Abschnitt 4.3.6.3 beschriebenen $\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$-Methode gewichtet. Aus den in Abschnitt 4.3.6.2 dargelegten Gründen wurde bei den extrahierten Wörtern keine Lemmatisierung angewendet.

4.3.4.4 Versuchsaufbau

Beim überwachten Lernen wird eine $n \times c$ Token-Klassen-Matrix $M = w_{ij}$ für eine Menge von abgegrenzten Klassen C gebildet. Für jedes Token i wird die klassenspezifische Gewichtung w_{ij} für die jeweilige Klasse j (KO, 2012) berechnet.[46] Jede Klasse wird dementsprechend durch einen prototypischen Klassenvektor $\overrightarrow{p_j}$ repräsentiert, der die charakteristische Tokenverteilung der Klasse über alle Module in den Trainingsdaten enthält (vgl. Abschnitt 4.3.6.3). Ein zu klassifizierendes Modul wird als Vektor $\vec{m} = (w_1, w_2, \dots w_n)$ repräsentiert, wobei n die Anzahl der als Merkmale gewählten Tokens eines Moduls ist und w die Gewichtung der Tokens abbildet. Der Kosinus der Vektoren im gemeinsamen Vektorraum wird für jede Modul-Klassen-Kombination berechnet (siehe Gleichung zu $\cos(\varphi)$ in Abschnitt 4.2.4.4) und die Konfidenz für die Vorhersage (nächstgelegene Klasse) abgeleitet (siehe Abschnitt 4.3.7).

Alle Klassifizierungsaufgaben, die im Folgenden vorgestellt werden, sind Multiklassen-Probleme (*multi class classification*). Für eine optimale Performanz basiert der Versuchsaufbau auf dem Vektorraummodell. Als Klassifikator wird aufgrund der hohen Anzahl an Merkmalen und der heterogenen Größe und Verteilung der Klassen (COLAS, PACLÍK, KOK & BRAZDIL, 2007) die Kosinusähnlichkeit (MANNING & SCHÜTZE, 1999) mit einem Nächste-Nachbarn-Klassifikator (auf engl.: k-nearest neighbor)[47] statt eines naiven Bayes-Klassifikators oder Support Vector Machines (SVM) eingesetzt.

---

[46] Gewichtungen werden innerhalb einer Klasse j normalisiert als $\frac{w_{ij}}{\sum w_{ij}}$.

[47] In diesem Fall ein „k-nearest neighbor"-Klassifikator mit $k = 1$.

Die gleichen Parameter und Konfigurationen werden für alle Klassifikationsaufgaben unabhängig von der Sprache des Datensatzes angewandt. Im Vergleich zu früheren Versuchen (OEVERMANN & ZIEGLER, 2016) wurde keine semantische Gewichtung spezifischer XML-Elemente verwendet, um eine bessere Vergleichbarkeit der Ergebnisse zu gewährleisten (siehe auch Abschnitt 4.3.6.4).[48]

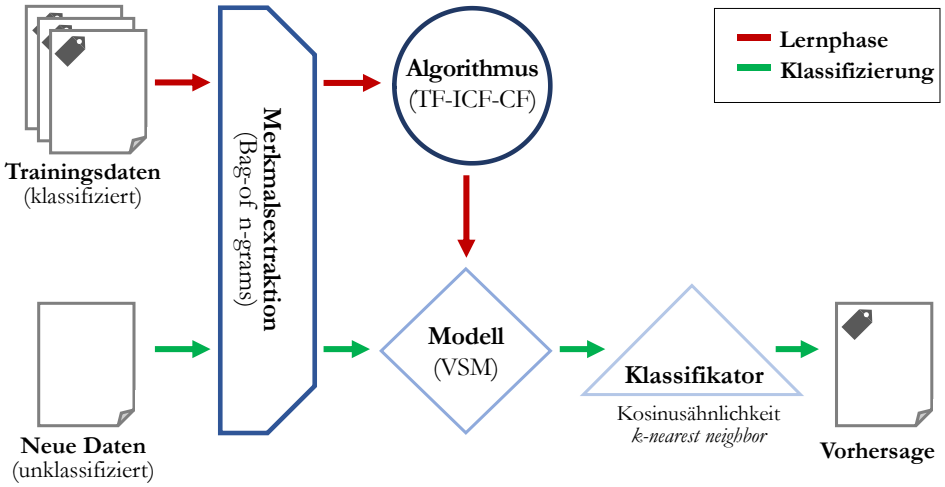

Abbildung 13: Versuchsaufbau der automatisierten Klassifizierung von Daten. Unterschieden in Lernphase und Klassifizierung.
Abbildung basierend auf einer Grafik aus Oevermann (2016a).

4.3.4.5 Messung der Ergebnisse

Für eine 10-fache Kreuzvalidierung wurden Module durch zufällige Auswahl in einen Trainingssatz und einen Testsatz im Verhältnis $9:1$ aufgeteilt (KOHAVI, 1995). Die Ergebnisse werden als mittlere Genauigkeit – im Englischen „mean accuracy" – (SOKOLOVA & LAPALME, 2009) über alle Klassen im jeweiligen Datensatz gemessen. Die Genauigkeit (*accuracy*) gibt demnach den Prozentsatz der durch den Klassifikator richtig vorhergesagten Klassifikationen an. Abweichungen zwischen den Ergebnissen der einzelnen Kreuzvalidierungstests werden als Standardabweichung s (MSE, engl. „Mean squared error") angegeben.

$$s_{n-1}^2 = \frac{1}{n-1}\sum 1_{i=1}^{n}(x_i - \overline{x})^2$$

[48] In frühen Versuchen wurde auch die Manhattan-Metrik (L_1-Distanz) und ein gewichtetes Skalarprodukt ($\frac{\vec{a}\circ\vec{b}}{|\vec{a}|}\cdot|\vec{b}|$) als Ähnlichkeitsmaß verwendet. Diese lieferten jedoch schlechtere Ergebnisse.

4.3.5 Charakteristiken

In den folgenden Abschnitten werden Charakteristiken von Modulen in der Technischen Dokumentation untersucht, die sich auf Vektorraum-basierte Klassifizierungsaufgaben auswirken können.

4.3.5.1 Inhaltsart

Die fachliche Domäne der Technischen Dokumentation umfasst die Erstellung und Strukturierung von Benutzerhandbüchern. Die darin enthaltenen Inhalte und deren Aufbau werden in vielerlei Hinsicht durch Normen und Vorschriften vorgegeben oder eingeschränkt. Eine der wichtigsten Regelungen definiert Inhaltsarten in der Reihenfolge der traditionellen Kapitelstrukturen für Handbücher und *Interaktive Elektronische Technische Dokumentation* (IEC 82079-1, 2012). Im Content Management werden diese Abschnitte durch Module abgebildet. Die zugehörigen Inhaltstypen entsprechen in der Regel den Lebenszyklusphasen von Produkten (2006/42/EG, 2006).

Dazu gehören beispielsweise Informationen zum Transport, der Installation und Einstellung von Maschinen oder Handlungsanweisungen zur Verwendung, Wartung und Entsorgung eines Produkts. Zusätzliche technische Daten, Sicherheitshinweise sowie konzeptionelle oder andere beschreibende Informationen (z. B. Konfiguration, Layout und Funktionalität des Produkts) sind vorgeschriebene Bestandteile. Die relevanten Vorschriften für ein Produkt geben daher eine bestimmte Zusammenstellung von Inhaltsarten für die zugehörige Technische Dokumentation bereits vor.

Diese Zusammenstellungen sind in vielen Fällen auch die Grundlage für spezialisierte Informationsmodelle. Verbreitete Beispiele sind Benutzerhandbücher für Militär und Luftfahrt (ATA iSPEC 2200, 2014; S1000D, 2017) oder medizinische Geräte (GHTF/SG1/N70, 2011).

4.3.5.2 Klassifikationsmodelle

Für Content-Management-Anwendungen können diese vorgegebenen Inhaltsarten in der Regel in abgegrenzte intrinsische Informationsklassen übertragen werden, während Dokumente (die wiederum verschiedene Inhaltsarten enthalten können) mit extrinsischen Informationsklassen wie „Servicehandbuch“ oder „Betriebsanleitung“ klassifiziert werden. Die gleiche Vorgehensweise kann auch auf produktbezogene Klassifikationen angewandt werden. In diesem Fall stellen intrinsische Produktklassen die Beziehung zwischen Inhalt und der Baugruppe oder Produktbestandteil her, der beschrieben wird (z. B. „Hydraulikpumpe“ oder „Kühleinheit“). Extrinsische Produktklassen beziehen sich auf Produkte (Modelle, Serien, Baureihen), in deren Kontext der Inhalt gültig ist.

Dieser Metadaten-basierte Ansatz für Content Management wird über die PI-Klassifikationsmethode von Ziegler definiert (DREWER & ZIEGLER, 2011) und wurde für die Klassifikation von Modulen konzipiert. Üblicherweise werden PI-Klassifikationen als Taxonomien definiert, die einen mindestens zweidimensionalen Informationsraum beschreiben. Eine Erweiterung des Metadatenmodells kann auch komplexere Szenarien berücksichtigen, wie z. B. ontologiebasierte Ansätze, bei denen Beziehungen zwischen Modulen definiert werden. Da die meisten Content-Management-Systeme in der Technischen Dokumentation auf taxonomische und listenbasierte Metadaten beschränkt sind, konzentrieren sich die folgenden Untersuchungen auf diese.

Nach der PI-Klassifikationsmethode muss jedes Modul eindeutige Koordinationen im Informationsraum der intrinsischen Produkt- und Informationsklassen haben. Technische Redakteure weisen diese Klassen üblicherweise bei der Erstellung des Inhalts zu und halten sich dabei an redaktionelle Vorgaben, die z. B. in einem Redaktionsleitfaden festgehalten sind. Der Hauptnutzen der Methode besteht in effizienten Retrieval-Mechanismen für Content-Management-Systeme und Content-Delivery-Portale (Suche, Filterung), sowie in der automatischen Aggregation von Dokumenten und der Ermöglichung von klassifikationsbasierten Querverweisen.

In den folgenden Abschnitten werden hauptsächlich intrinsische Informationsklassen betrachtet, da diese eine direkte Verbindung zu den linguistischen Eigenschaften der Module und den zugrundeliegenden Informationsmodellen haben. Zusätzlich werden die gleichen Klassifikationsmethoden auf intrinsische Produktklassen angewandt, um zu evaluieren, ob diese andere Anpassungen an den Klassifizierungsprozess erfordern. Extrinsische Klassen werden nicht behandelt, da es sich bei diesen in den meisten Fällen um Multilabel-Probleme (Mehrfachzuweisungen von Klassen) handelt und diese durch andere Verfahren gelöst werden können. Eine Möglichkeit zur Zuordnung von extrinsischen Produktklassen stellt z. B. die Entitätenerkennung (NER; engl. „Named entity recognition“) von Produktnamen dar.

4.3.5.3 Formulierungsmuster

Innerhalb von Modulen werden in der Technischen Dokumentation standardisierte grammatische Formulierungsmuster verwendet, um die Wiederverwendung und inhaltliche Konsistenz über verschiedene Dokumente hinweg zu gewährleisten. Gerade Betriebsanweisungen müssen auf Grund der rechtlichen Ausgangslage besonders prägnant und eindeutig formuliert sein. Wiederkehrende strukturelle und grammatische Muster helfen dabei, die Textverständlichkeit zu verbessern. Ein bekanntes Beispiel dafür sind normierte Sicherheitshinweise (ANSI Z535.6, 2006).

Vorgaben zum standardisierten Aufbau einer Technischen Dokumentation werden idealerweise in einem Redaktionsleitfaden oder internen Richtlinien festgehalten (IEC 82079-1, 2012), welche Technische Redakteure dazu anhalten auf Synonyme, Mehrdeutigkeit, direkte Anrede, Füllwörter, emotionale Formulierungen oder leere Phrasen zu verzichten. Manche Unternehmen verwenden darüber hinaus kontrollierte Sprachen – wie z. B. *Simplified Technical English* – um die Grammatik und das Vokabular weiter einzuschränken (KINCAID, KINCAID, KNIFFIN, THOMAS & LANG, 1991). Standardisierte Formulierungsmuster können auch die Übersetzungskosten senken, wenn sie in Kombination mit Übersetzungsmanagementsystemen (TMS, engl.: „Translation Management System") eingesetzt werden (ALLEN, 1999) ebenso können sie die Lesbarkeit von Texten verbessern. Ein Beispiel dafür ist die deutliche grammatische Unterscheidung von anleitenden (*Instruktion*) und beschreibenden (*Deskription*) Inhalten. Dies unterstützt Leser bei der Einordnung der entsprechenden Textteile. Module einer Informationsklasse erfordern oft automatisch eine spezifische Kombination aus Formulierungsmustern. XML-basierte Informationsmodelle, wie z. B. DITA (OASIS, 2010), DocBook (OASIS, 2008) oder PI-Mod (ZIEGLER & STEURER, 2010), setzen dies mit Hilfe von speziellen XML-Elementen für semantisch unterschiedliche Modultypen um (vgl. die standardisierten Typen von DITA-Topics: Concept, Task, Reference etc.). Sprachkontrollwerkzeuge (CLC; engl.: „Controlled language checker") oder Redaktionsleitfäden können helfen, diese grammatischen und syntaktischen Regeln abhängig von Topictyp oder der Informationsklasse umzusetzen.

4.3.5.4 Terminologie

Die in Technischen Dokumentationen (und anderen regulierten Dokumenten) eingesetzte Terminologie ist oft sehr spezifisch für den beschriebenen Inhalt festgelegt worden und wird idealerweise während des Schreibprozesses kontrolliert. Grundlage dafür sind die Prinzipien der Terminologiearbeit, die durch Terminologieprüfwerkzeuge forciert werden (ISO 704, 2009, S. 704; ISO 26162, 2012). Charakteristische Terme in Technischer Dokumentation sind meist technische Fachausdrücke, die nur innerhalb einer bestimmten Branche oder Industrie Verwendung finden. Da in manchen Fällen die Technische Dokumentation auch als Marketinginstrument betrachtet wird, setzen einige Unternehmen in Texten die expliziten Produktbezeichnungen (oft eine Kombination aus Hersteller-, Produkt- und Modellname) ein, um die Markenwiedererkennung zu fördern. Dies führt zu hoch-charakteristischen Wortverteilungen in Modulen, was zwar Vorteile für die Klassifikationsgenauigkeit hat, jedoch den Einsatz von trainierten Modellen stark auf den ursprünglichen Bereich des Inhalts einschränkt (weshalb z. B. trainierte Modelle nicht über verschiedene Firmen hinweg verwendet werden können).

4.3.5.5 Modularisierung und Inhaltsgröße

Die konkrete Größe von Modulen hängt von verschiedenen Faktoren ab, wie z. B. strategischen Entscheidungen, der Produktkomplexität oder der Funktionalität des eingesetzten Redaktionssystems (DREWER & ZIEGLER, 2011). In verwandten Arbeiten wurden relevante Moduleigenschaften systematisch für verschiedene Datensätze von Unternehmen analysiert, wobei die eigentliche Größe von kleinen Fragmenten mit wenigen Wörtern bis zu Inhalten mit mehreren Tausenden Wörtern gereicht hat (OBERLE & ZIEGLER, 2012). Ein untersuchter Datensatz hatte z. B. eine durchschnittliche Größe von 150 Wörtern pro Modul im Gegensatz zu einer Dokumentgröße von ca. 12.000 Wörtern. Die sog. *Fragmente* (DREWER & ZIEGLER, 2011) werden üblicherweise in andere Modulen eingebettet, können aber auch eigene Klassifikationen tragen (z. B. bei Sicherheitshinweisen). Diese sehr kleinen Inhaltsbausteine sind in komplexen Wiederverwendungsszenarien üblich, bei denen das Variantenmanagement systemgestützt umgesetzt wird (ROCKLEY u. a., 2003).

Die Datensätze, die in dieser Arbeit untersucht werden, haben eine durchschnittliche Größe von 51 bis 515 Wörtern pro Modul (siehe Tabelle 6). Die Größe von Modulen ist demnach signifikant kleiner als die eines typischen Dokuments (ca. 1 : 75, dieses Verhältnis entspricht auch der durchschnittlichen Anzahl von Modulen pro Dokument). Daraus ergeben sich im Vergleich zur etablierten Dokumentenklassifikation erheblich weniger Merkmale, die von einem Klassifikator ausgewertet werden können und es besteht eine hohe Varianz in der Größe der Elemente (*Heterogenität*) über unterschiedliche Firmen und Datensätze hinweg (siehe Tabelle 6).

4.3.5.6 Trainings- und Testdaten

Unternehmen, die Content-Management-Methoden in Kombination mit einem ausdefinierten Klassifikationsmodell einsetzen, haben bereits Trainingsdaten in hoher Qualität vorliegen, die für ein überwachtes Lernen geeignet sind. In diesem Fall wurden die Module von Hand von Experten klassifiziert und nach Redaktionsrichtlinien verfasst. Standardisierte Informationsmodelle oder semantisch angereichertes HTML können weitere Informationen über semantische Eigenschaften und Funktionen von Textteilen liefern (siehe Abschnitt 4.3.6.4). Für einige Teile von Technischer Dokumentation kann sich der sehr technische Charakter des Inhalts jedoch auch negativ auf die Klassifikationsperformance auswirken (z. B. für Tabellen, Legenden oder Listen).

Testdaten für die automatisierte Klassifizierung können entweder strukturierte, aber nicht-klassifizierte Module aus Redaktionssystemen und anderen Quellen sein (z. B. bevor in einem Unternehmen Klassifizierungsmodelle eingeführt wurden) oder unstrukturierte und nicht-klassifizierte PDF-Dokumente oder andere Dateiformate, die zur Archivierung verwendet werden.

Insbesondere diese Bestandsdaten spielen in der Technischen Dokumentation eine wichtige Rolle, da Hersteller gesetzlich dazu verpflichtet sind, Dokumentationen für mehrere Jahre nach dem Inverkehrbringen eines Produktes aufzubewahren (in der EU beträgt die Pflicht zur Aufbewahrung der Dokumentation für Maschinen z. B. 10 Jahre (2006/42/EG, 2006)). Diese Diskrepanz führt zu potenziellen Unterschieden zwischen Trainings- und Testdaten bezüglich Format, Struktur und Qualität, weshalb domänenspezifische Klassifikationsansätze diese Faktoren berücksichtigen müssen.

4.3.5.7 Qualitätssicherung

Aufgrund der hohen rechtlichen und sicherheitsrelevanten Anforderungen, die für Technische Dokumentation gelten, ist eine gründliche Qualitätssicherung vor der Veröffentlichung der Inhalte verpflichtend (IEC 82079-1, 2012; ISO 9001, 2008). Insbesondere in der EU werden alle notwendigen Technischen Unterlagen (zu denen auch die Betriebsanleitung zählt) als wesentlicher Bestandteil des Produkts betrachtet (2006/42/EG, 2006). Korrektheit und Vollständigkeit der publizierten Dokumente sind demnach auch entscheidend für die Integrität des gesamten Produkts. Da manche Redaktionssysteme die automatisierte Zusammenstellung von Dokumenten aus Modulen auf der Basis von Klassifikationen vornehmen, ist ein Klassifikationsalgorithmus eine mögliche Schwachstelle für die Qualität einer Dokumentation. Aus diesen Anforderungen ergibt sich, dass eine messbare Vertrauensgröße – die Konfidenz des Klassifikators – unerlässlich für den Einsatz im Bereich der Technischen Dokumentation ist, da diese als möglicher Grenzwert dienen kann, wenn Ergebnisse nicht eindeutig oder unzuverlässig erscheinen und eine manuelle Kontrolle notwendig wird.

4.3.6 Auswirkungen und Anpassungen

In den folgenden Abschnitten werden die Auswirkungen auf die automatisierte Klassifizierung diskutiert, die sich aus den Charakteristiken Technischer Dokumentation ergeben. Die daraus folgenden Anpassungen am Prozess werden anschließend mit Hilfe von Testdaten verifiziert.

4.3.6.1 Merkmalsauswahl

Standardisierte Terminologie und Formulierungsmuster reduzieren die Gesamtzahl der unterschiedlichen Wörter und Wortkombinationen in Technischer Dokumentation im Vergleich zu anderen Textsorten. Dies ist bei der Textklassifizierung generell von Vorteil, da es die übliche hohe Dimensionalität des Vektorraums reduziert (CALDAS u. a., 2002). Da Module in der Regel wesentlich kleiner sind als Dokumente (vgl. Abschnitt 4.3.5.5), wird die Anzahl der Merkmale zur Darstellung eines Objekts weiter reduziert.

Jedoch werden die meisten Module in der Technischen Dokumentation sowohl von unterschiedlichen Termen als auch von erkennbaren Formulie-

rungsmustern charakterisiert, die deshalb wichtige Merkmale ihrer informations- oder produktbezogenen Klassifizierung sind (siehe Erläuterungen in den Abschnitten 4.3.5.3 und 4.3.5.4). Obwohl die optimale Merkmalsauswahl von den spezifischen Eigenschaften eines Datensatzes abhängt, funktioniert eine Kombination aus Einzelwörtern und Wortmustern am besten über verschiedene Datensätze hinweg (siehe Tabelle 7 für $w_{ij} = \mathrm{TF} - \mathrm{ICF} - \mathrm{CF}$).

Die erzielten Ergebnisse bestätigen die Annahme, dass eine Kombination von n-Grammen (wobei n die Anzahl der Wörter pro Gruppe ist) in den meisten Fällen die bevorzugte Methode zur Repräsentation von Modulen ist (siehe Tabelle 7 für $w_{ij} = \mathrm{TF} - \mathrm{ICF} - \mathrm{CF}$). Unter Berücksichtigung der Standardabweichung der Kreuzvalidierungstests (zwischen $1 - 3\,\%$ bei allen Tests) zeigt sich, dass die beste Durchschnittsgenauigkeit durch kombinierte Wortmuster als Merkmale erreicht wurde ($n = \{1,2\}$ und $n = \{1,2,3\}$).

Da eine hohe Anzahl von Merkmalen die Rechenzeit negativ beeinflussen kann, ist $n = 2$ die beste Wahl für leistungskritische Anwendungen.

Die Klassifizierungszeit kann dadurch bei ähnlich guter Genauigkeit erheblich reduziert werden. Es konnte kein signifikanter Zusammenhang zwischen der Sprache des Inhalts und dem optimalen Wert für n festgestellt werden.

Tabelle 7: Genauigkeit für verschiedene n-Gramme als Merkmale
Klassifikationsaufgabe: Informationsklasse Stufe 1, wobei n die Anzahl der Wörter pro Merkmalsgruppe für $w_{ij} = TF - ICF - CF$ ist.
*Die besten Ergebnisse werden pro Datensatz jeweils in **fett** hervorgehoben.*

n	Satz A (en) [%]	Satz B (de) [%]	Satz C (de) [%]	Satz D (en) [%]	Durchschnitt [%]
1	86,7	78,5	75,9	75,5	79,2
2	91,7	85,4	81,7	82,1	85,2
3	**92,5**	85,9	73,6	75,7	81,9
4	92,1	83,7	67,9	73,6	79,3
{1,2}	90,1	83,5	79,7	80,7	83,5
{2,3}	91,9	**87,0**	81,1	82,4	85,6
{1,2,3}	91,6	85,3	**82,6**	**83,6**	**85,8**

4.3.6.2 Lemmatisierung

Die Anwendung einer listenbasierten Lemmatisierung[49] für die deutschsprachigen Datensätze hat die Klassifikationsgenauigkeit nicht verbessern können und sie in manchen Fällen sogar verschlechtert.[50] Dieses Verhalten kann auf die Verwendung von Wortmustern als Merkmale zurückgeführt werden, welche wichtige grammatische Informationen enthalten können (z. B. Verbkonjugationen bei der Klassifikation von Informationsarten), die bei einer Lemmatisierung oder einem Stemming verloren gehen können. Aus diesem Grund wurde keine Lemmatisierung angewandt.

4.3.6.3 Merkmalsgewichtung

Unter den vielen Methoden einem Merkmal eine kontextabhängige Gewichtung zuzuweisen, ist $\mathrm{TF}-\mathrm{IDF}$ die mit Abstand bekannteste (CALDAS u. a., 2002; KO, 2012; LIU & YANG, 2012). $\mathrm{TF}-\mathrm{IDF}$, das meist im Dokumenten-Retrieval eingesetzt wird, kombiniert die Gesamtvorkommenshäufigkeit eines Merkmals (Termfrequenz: TF) mit der inversen Dokumentenfrequenz (IDF, vgl. z. B. bei BAEZA-YATES & RIBEIRO-NETO (1999)), die als Indikator für die Seltenheit bzw. Einzigartigkeit eines Terms i in einem Dokument n gilt:

$$\mathrm{w_{ij}} = \mathrm{tf_{ij}} \cdot \log\left(\frac{\mathrm{N}}{\mathrm{n_i}}\right)$$

Darüber hinaus existieren viele Varianten von $\mathrm{TF}-\mathrm{IDF}$, die ein Glätten (engl.: „Smoothing“) integrieren, um auch Fälle zu berücksichtigen, in denen die Termfrequenz null ist (LIU & YANG, 2012). Diese Variante wird im Folgenden als $\mathrm{TF}-\mathrm{IDF_{smooth}}$ bezeichnet:

$$\mathrm{w_{ij}} = \log\left(1+\mathrm{tf_{ij}}\right) \cdot \log\left(\frac{1+\mathrm{N}}{\mathrm{n_i}}\right)$$

Um die Genauigkeit von Dokumentkategorisierungen zur verbessern, wurde $\mathrm{TF}-\mathrm{IDF}$ zu $\mathrm{TF}-\mathrm{IDF}-\mathrm{CF}$ erweitert, welches auch die Merkmalsvorkommen innerhalb einer bestimmten Klasse berücksichtigt (LIU & YANG, 2012):

$$\mathrm{w_{ij}} = \log\left(1+\mathrm{tf_{ij}}\right) \cdot \log\left(\frac{1+\mathrm{N}}{\mathrm{n_i}}\right) \cdot \frac{\mathrm{tf_{ij}}}{\mathrm{C_j}}$$

---

[49] Im Englischen oft synonym als *Stemming* bezeichnet. Zu den Unterschieden vgl. MANNING, RAGHAVAN & SCHÜTZE (2008): „Stemmers use language-specific rules, but they require less knowledge than a lemmatizer“. Hier wurde *Morphy* als Basis der Liste verwendet (LEZIUS, 2000).

[50] Dieses Verhalten lässt sich im Information Retrieval für die meisten Sprachen beobachten (darunter Deutsch und Englisch), siehe z. B. bei MANNING, RAGHAVAN & SCHÜTZE (2008); SINGHAL (2001). Spanisch und Finnisch gelten als Außnahmen, bei denen ein Stemming die Ergebnisse verbessern kann.

Im Content Management ist die Referenzgröße einer Einheit jedoch ein Modul und kein Dokument. Daher ist die dokumentenbasierte Gewichtung nicht unbedingt für die Klassifizierung von Modulen geeignet. Aufgrund der Art der Trainingsdaten, aus denen die Gesamtfrequenz tf_i, die Klassenfrequenz cf_{ij} und die inverse Klassenfrequenz icf_{ij} abgeleitet werden kann, wurde $\mathrm{TF}-\mathrm{IDF}-\mathrm{CF}$ an die Verwendung der inversen Klassenfrequenz (ICF, *Inverse Class Frequency*) angepasst, um Klassen statt IDF zu unterscheiden. Für einen Satz von verschiedenen Klassen C mit den Klassen j und Merkmalen i wird die Gewichtung w_{ij} mit $\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$ wie folgt berechnet:

$$\mathrm{w_{ij}} = \log(1+\mathrm{tf_i}) \cdot \log\left(1+\frac{|\mathrm{C}|}{\mathrm{tf_i}}\right) \cdot \frac{\mathrm{tf_{ij}}}{\mathrm{C_j}}$$

Die Versuchsergebnisse bestätigen, dass $\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$ bei den untersuchten Datensätzen im Vergleich zu dokumentzentriertem Vorgehen als Gewichtungsmethode am besten abschneidet (vgl. Tabelle 8 für $n=\{1,2,3\}$)

Tabelle 8: Genauigkeit für verschiedene Gewichtungsmethoden.
Klassifikationsaufgabe: Informationsklasse Stufe 1 für $n=\{1,2,3\}$.
*Die besten Ergebnisse werden pro Datensatz jeweils in **fett** hervorgehoben.*

w_{ij}	Satz A (en) [%]	Satz B (de) [%]	Satz C (de) [%]	Satz D (en) [%]	Durchschnitt [%]
$\mathrm{TF}-\mathrm{IDF}$	25,3	50,5	25,2	47,6	37,2
$\mathrm{TF}-\mathrm{IDF_{smooth}}$	63,5	39,8	25,2	65,4	48,5
$\mathrm{TF}-\mathrm{IDF}-\mathrm{CF}$	85,8	69,2	72,6	69,4	74,3
$\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$	**91,6**	**85,3**	**82,6**	**83,6**	**85,8**

Ein zusätzlicher Vorteil der vorgeschlagenen $\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$-Metrik ist die Unabhängigkeit von Dokumenteneinheiten (in diesem Fall: Modulen) bei der Berechnung der Gewichtungen, was zu einer sehr effizienten Trainingsphase führt, die nur linear von der Anzahl der Klassen und der Anzahl der extrahierten Merkmale abhängt. Dies ist möglich, da innerhalb der PI-Klassifikationsmethode ein Modul immer eine Instanz einer intrinsischen Klasse oder Klassenkombination ist.

Die $\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$-Gewichtung eignet sich daher für Anwendungen, die sich auf gemeinsame Merkmale einer Klasse konzentrieren und nicht auf Informationen über einzelne Einheiten innerhalb einer Klasse angewiesen sind (wie

z. B. beim Ranking im Information Retrieval). Beispiele für Anwendungen von $\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$ sind überwachtes Lernen (wie in dieser Arbeit gezeigt) und Qualitätsmaßnahmen für Klassifizierungssysteme (z. B. Klassenverteilung, fehlende Klassen, durchschnittliche Klassenentfernung im Vektorraum).

4.3.6.4 Semantische Gewichtung

Semantische Informationen über die Textstruktur von Modulen sind in Trainingsdaten in der Regel als XML-Elemente, deren Attribute oder Metastruktur verfügbar (DI IORIO u. a., 2012). Diese Zusatzinformationen fehlen jedoch oft in den Testdaten (z. B. in Bestandsdokumentation wie PDF-Dokumenten), was einen direkten Vergleich erschwert. Um dies zu umgehen, ist es möglich, die Termfrequenz tf_i mit einem Faktor q künstlich zu erhöhen. Sinnvoll ist dies für Text innerhalb von Elementen, die in einer Klasse eine besondere semantische Bedeutung haben (z. B. Überschriften, Technische Daten, Resultate von Handlungen). Dadurch wird tf_i im überwachten Lernen durch einen konstanten Faktor q erweitert zu:

$$tf_{iq} = tf_i \cdot q \qquad \text{für } q > 0$$

Ergebnisse früherer Arbeiten (OEVERMANN & ZIEGLER, 2016) zeigen, dass sich die Klassifizierungsgenauigkeit bei $2 < q < 5$ um bis zu 10% verbessern kann ($q = 2{,}5$). Dieses Verhalten ist auf ein bekanntes Problem der hochdimensionalen Merkmalsauswahl für Textklassifizierung zurückzuführen: das Fehlen gut vorhersagbarer Merkmale, die eine Klasse von einer anderen unterscheidet (FORMAN, 2004). Dieser Effekt wird auch als „Siren pitfall" bezeichnet. Allerdings beeinflussen die Qualität und die Wahl der semantischen Strukturen für die Gewichtung stark die Vorteile der Quantisierung. Elemente für die semantische Gewichtung können derzeit nur von Hand ausgewählt werden, was zu einem schwierigen und verzerrten direkten Vergleich zwischen Datensätzen mit unterschiedlichen Informationsmodellen führt. In einer Versuchsreihe wurde versucht, diese Auswahl auf der Grundlage von zwei Hypothesen zu automatisieren:

- Die Auswahl von XML-Elementen, die nur in einer Klasse vorkommen und daher für diese Klasse charakterisierend sind.
- Die Anwendung der Gewichtungsmethode $\mathrm{TF}-\mathrm{ICF}-\mathrm{CF}$ auf alle XML-Elemente und die anschließende Auswahl der Elemente mit dem höchsten Gewicht pro Klasse.

Beide Versuche führten nicht zu einer signifikanten Verbesserung der Klassifikationsergebnisse sondern vielmehr zu einer Verschlechterung der Genauigkeit bei der Kreuzvalidierung aufgrund einer Überanpassungen des Modells (dem sog. *Overfitting*).

4.3.6.5 Konfidenzmessung

Auf Basis der zuvor dargelegten Gründe (vgl. Abschnitt 4.3.5.7) ist es notwendig, die Konfidenz des Klassifikators messen zu können, um die automatisierte Klassifizierung in einen redaktionellen Workflow zu integrieren. Während in Retrieval-Szenarien, wie dem Filtern in einem Content-Delivery-Portal, eine falsche oder fehlende Klassifizierung nur wenig Auswirkungen hat (Recall ist wichtig), kann sie für automatisierte Publikationsprozesse entscheidend sein (Precision ist wichtig).

Es gibt mehrere Methoden zum Vergleich der klassenspezifischen Klassifizierungswerte s_c, wie z. B. die *Softmax*-Funktion oder die Berechnung der Standardabweichung, aber keine davon entsprach den Anforderungen an eine zuverlässige Qualitätssicherungsmaßnahme. Im Folgenden wurde daher ein einfaches Verhältnismaß verwendet:

$$p = \frac{s_1 - s_2}{s_1 - s_n}$$

Der vorgestellte Konfidenzwert basiert auf dem Vorhandensein von deutlichen Ausreißern oder einem knappen Unterschied der Klassifikationswerte. Die klassenspezifischen Klassifizierungswerte s_c für n Klassen c werden von hoch (1) nach niedrig (n) sortiert. Die Konfidenz p entspricht dann dem Verhältnis von erstem zu zweitem und erstem zu letzem Klassifikationswert.

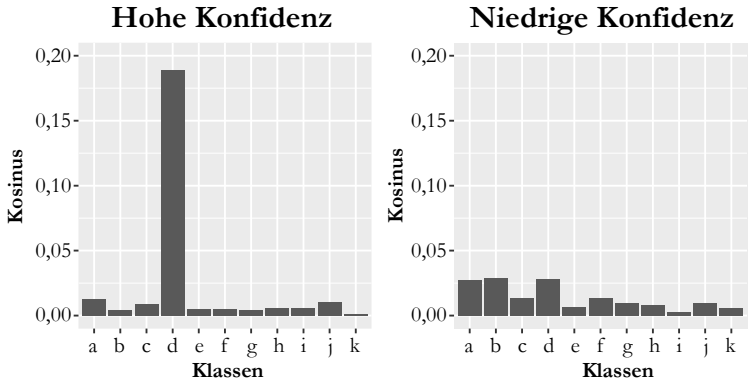

Abbildung 14: Visualisierung der Konfidenz-Hypothese.
Grundlage für die Berechnung des Konfidenzwerts p. Hohe Konfidenz (links) bei eindeutiger Nähe eines Klassenvektoren (in diesem Fall d), niedrige Konfidenz (rechts) bei ähnlich nahen Klassenvektoren (a, b, d) im Vektorraum.
Abbildung basierend auf einer Grafik aus Oevermann (2016b) .

4.3.7 Ergebnisse und Diskussion

Die Anpassungen wurden mit elf Klassifizierungsaufgaben auf der Grundlage von vier Datensätzen ($A - D$) getestet, die in Tabelle 6 aufgelistet sind.

Allgemeine Ergebnisse sind in Tabelle 9 aufgeführt, jeweilige Detailangaben sind in Tabelle 7, Tabelle 8 und Tabelle 10 gelistet. Die unterschiedlichen Aufgaben ergeben sich aus den unternehmensspezifischen PI-Klassifikationsmodellen der Daten und unterscheiden sich in ihren Eigenschaften erheblich.

4.3.7.1 Allgemeine Ergebnisse

Die besten Resultate wurden mit intrinsischen Stufe-1-Klassifikationen von Informationsarten ($91{,}6\ \% \pm 1{,}7$ für 10 Klassen bei Datensatz A) und Produktklassen ($82{,}5\ \% \pm 2{,}1$ für 28 Klassen bei Datensatz B) erzielt. Mit einer datenspezifischen Merkmalsauswahl kann die Genauigkeit für Datensatz A bis zu $92{,}5\ \% \pm 2{,}3$ für 10 Klassen weiter verbessert werden. Stufe-2-Ergebnisse von Informationsklassen variieren zwischen $74{,}8\ \%$ (D) und $87{,}8\ \%$ (A). Für diese Szenarien könnte die Genauigkeit durch den Einsatz von Fallback-Mechanismen, wie in Abschnitt 4.3.10.3 beschrieben, erhöht werden.

Die Ergebnisse der Experimente zeigen, dass ein Vektorraum-basierter Klassifikationsprozess in den vorgestellten Anwendungsfällen sinnvoll und effizient sein kann (vgl. Abschnitt 4.3.9). Alle Ergebnisse basieren (soweit nicht anders angegeben) auf den gleichen Parametereinstellungen (sog. „Zero configuration“) und unveränderten Datensätzen aus der Industrie, die direkt aus Redaktionssystemen exportiert wurden, was für eine reale Anwendung der vorgestellten Methode von großer Bedeutung ist. In bestimmten Fällen konnte gezeigt werden, dass Probleme der Produktklassifizierung mit den gleichen Methoden wie bei der Klassifizierung von Informationstypen gelöst werden können. Zur Bestätigung dieser Beobachtung müssen allerdings zusätzliche Experimente an weiteren Datensätzen durchgeführt werden.

Aufgrund der begrenzten Datenmenge können keine endgültigen Rückschlüsse auf den Einfluss der äußeren Faktoren (Informationsmodell, Datensatzgröße, Modulgröße, Sprache, Klassifikationstyp und Anzahl der Klassen) auf die Klassifikationsergebnisse gezogen werden. Basierend auf subjektiver Einschätzung ist der wichtigste Faktor für eine hohe Genauigkeit bei der automatisierten Klassifizierung weiterhin qualitativ hochwertiger Inhalt – sowohl in den Trainings- als auch in den Testdaten. Dazu gehören ein gut definiertes Klassifikationsmodell, eine korrekt durchgeführte manuelle Klassifikation und standardisierter Text, der nach den redaktionellen Richtlinien verfasst wurde. Diese angesprochenen Qualitätsaspekte von Content objektiv zu messen, ist Gegenstand weiterer Forschung.

Tabelle 9: Gesamtergebnisse der automatisierten Klassifizierung.
Verschiedene Klassifikationsaufgaben mit $n = \{1,2,3\}$ und $w_{ij} = TF - ICF - CF$.
*Die besten Ergebnisse werden pro Klassifikationsaufgabe jeweils in **fett** hervorgehoben.*

Daten-satz	Sprache	Klassifikations-aufgabe	Anzahl Klassen	Genauigkeit[51] [%]	Standard-abweichung
A	en-US	**Informationsklasse** – Stufe 1	10	**91,6**	1,7
		Informationsklasse – Stufe 2	26	**87,8**	3,6
B	de-DE	**Informationsklasse** – Stufe 1	6	85,3	2,5
		Informationsklasse – Stufe 2	22	78,0	2,3
		Produktklasse – Stufe 1	28	**82,5**	2,1
C	de-DE	**Informationsklasse** – Stufe 1	11	82,6	3,1
		Produktklasse – Stufe 1	22	74,5	7,1
D	de-DE	**Informationsklasse** – Stufe 1	8	78,4	6,8
		Informationsklasse – Stufe 2	14	80,7	4,8
	en-GB	**Informationsklasse** – Stufe 1	8	83,6	3,7
		Informationsklasse – Stufe 2	17	74,8	4,1

[51] Als „Mean accuracy“, siehe Abschnitt 4.3.4.5

4.3.7.2 Korrelationen

Die Ergebnisse weisen die folgenden messbaren Korrelationen[52] auf:

- Die Größe der Trainingsdaten korreliert negativ mit der Standardabweichung der Kreuzvalidierung ($\rho = -0{,}6$).
- Die durchschnittliche Größe von Modulen (in Anzahl der Wörter pro Modul) korreliert mit der Klassifikationsgenauigkeit ($\rho = 0{,}7$).

Auch wenn aus den vorliegenden Daten keine signifikante Korrelation hervorgeht, ist davon auszugehen, dass die Anzahl der Klassen sich negativ auf die Klassifikationsgenauigkeit auswirkt, da mit mehr Klassen auch die Wahrscheinlichkeit einer (zufällig) korrekten Vorhersage abnimmt. Innerhalb einzelner Datensätze ist dieses Verhalten bereits beobachtbar, wenn die Ergebnisse zwischen den Stufen innerhalb einer Klassifikationsaufgabe abweichen. Eine Anomalie dieser Vermutung in den Ergebnissen, bei der die Stufe 2 der Informationsklasse im deutschen Teil von Datensatz D bessere Ergebnisse erzielt als Stufe 1, liegt innerhalb der Standardabweichung.

4.3.7.3 Auswahl und Gewichtung der Merkmale

Die Resultate in Tabelle 7 und Tabelle 8 zeigen, dass Auswahl und Gewichtung von Merkmalen auf die Charakteristiken von Modulen der Technischen Dokumentation angepasst werden können. Durch den hohen Standardisierungsgrad innerhalb der Textsorte wird deutlich, dass eine Kombination aus Wortgruppen ($n = 2$ bzw. $n = 3$) und Einzelwörtern ($n = 1$) der beste Weg ist, um Module in Klassifikationsszenarien zu repräsentieren (mit Bigrammen als eine effiziente Alternative).

Zur Gewichtung der Merkmale konnten durch Einsatz der $\mathrm{TF} - \mathrm{ICF} - \mathrm{CF}$-Methode die Ergebnisse signifikant gegenüber dokumentzentrierten Ansätzen verbessert werden (vgl. Tabelle 8).

4.3.7.4 Konfidenz des Klassifikators

Um die Verlässlichkeit einer Qualitätssicherung zu testen, die auf einem zuvor festgelegten Konfidenzgrenzwert p basiert, wurden die Konfidenzwerte bei der Kreuzvalidierung berechnet und die Module identifiziert, die falsch klassifiziert wurden, aber eine hohe Konfidenz ($p > 0{,}7$) zugewiesen bekommen haben.[53]

---

[52] Berechnet als Korrelationskoeefizient nach Pearson: $\rho_{X,Y} = \frac{\mathrm{cov}(X,Y)}{\sigma_X \sigma_Y}$

[53] Wird die Unterschreitung des Grenzwerts als Indikator für eine zusätzliche menschliche Kontrolle verwendet, würden diese Module trotz inkorrekter Klassifikation dann fälschlicherweise unkontrolliert bleiben.

Tabelle 10: Validierung der Methode zur Konfidenzmessung. Anteil der falsch klassifizierten Module (F) bei denen der Konfidenzwert $p > 0{,}7$ (Klassifikationsaufgabe: Informationsklasse der Stufe 1) mit $n = \{1,2,3\}$ und der Gewichtung $w_{ij} = TF - ICF - CF$.

	Satz ***A*** (en) [%]	Satz ***B*** (de) [%]	Satz ***C*** (de) [%]	Satz ***D*** (en) [%]	Durchschnitt [%]
$F_{p>0,7}/_{F}$	4,49	1,59	9,49	4,76	5,08

Die Ergebnisse zeigen, dass ein passend gewählter Grenzwert (hier: $p > 0{,}7$) ausreichend verlässlich den Anforderungen einer Qualitätssicherung genügt und automatisierte Workflows ermöglicht, bei denen Module, die mit einer hohen Konfidenz klassifiziert wurden, ohne weitere manuelle Kontrolle bei niedriger Fehlerquote verarbeitet werden können.

4.3.7.5 Einschränkungen

Aufgrund noch fehlender Forschung zur automatisierten Klassifikation im Bereich der Technischen Dokumentation gibt es wenig Grundlagen für einen Quervergleich der Ergebnisse. Zusätzliche Experimente mit alternativen maschinellen Lernmethoden zur Textklassifizierung an vergleichbaren Daten müssen durchgeführt werden, um eine allgemeinere Bewertung und eine vergleichende Einordnung zu erhalten.

4.3.8 Implementierung

Im Rahmen der Arbeit wurde ein plattformunabhängiges Framework für die vorgeschlagenen Methoden implementiert, das zur Validierung der praktischen Anwendbarkeit dienen soll. Der Prototyp ist so konzipiert, dass er eine einfache Erweiterung, Konfiguration und Nutzung ermöglicht und mit einer Vielzahl an Datenformaten arbeiten kann. Dadurch ist es auch möglich, zusätzliche Funktionen wie Ähnlichkeitsanalysen für Module (siehe Abschnitt 4.2), Analysen für unstrukturierte Inhalte (siehe Abschnitt 4.4) oder den Export von Ergebnissen in verschiedene Datenformate hinzuzufügen.

Die Software wurde in JavaScript implementiert und ist in allen modernen Browsern lauffähig. Training und Klassifizierung werden clientseitig verarbeitet, so dass keine Daten an den Server gesendet werden. Eine browserbasierte Demo[54] und der dazugehörige Quellcode[55] sind öffentlich zugänglich.

[54] http://coin.fastclass.de/ [01.02.19]

[55] https://github.com/j-oe/coin-demo [01.02.19]

4.3.9 Anwendungen

Die folgenden Abschnitte geben einen kurzen Überblick über mögliche Anwendungen für die automatisierte Klassifizierung von Modulen in der Technischen Dokumentation.

4.3.9.1 Autorenunterstützung

Autoren, die Inhalte in einem Redaktionssystem erstellen, legen die Klassifikation eines Moduls in der Regel fest, wenn sie mit der Erstellung des Textes beginnen. In einigen Fällen ändert sich jedoch der Inhalt im Laufe der Zeit oder der Autor wählt zu Beginn die falsche Klassifikation aus. Dies kann zu Problemen bei der Identifizierung und Einordung des Moduls in einer späteren Phase des Informationslebenszyklus führen.

Vor der Speicherung bzw. dem Einchecken des neu geschriebenen Moduls in die Datenbank kann die automatisierte Klassifizierung als zusätzliche Qualitätssicherung im Hintergrund dienen, indem sie die manuell zugeordnete Klasse mit den Ergebnissen einer automatisierten Klassifizierung vergleicht (OEVERMANN, 2016a). Im Falle einer Abweichung der beiden kann der Autor darauf hingewiesen werden, die zugewiesene Klassifikation nochmals zu prüfen (z. B. in Form einer Warnmeldung oder eines Indikators).[56]

4.3.9.2 Datenmigration

Mit der Einführung eines Redaktionssystems beginnen Unternehmen auch oft mit der Verwendung von Klassifikationen (z. B. in Form eines PI-Klassifikationsmodells), um die Vorteile klassifizierender Metadaten wie eine automatisierte Dokumentenaggregation oder die erweiterte Bereitstellung über Content-Delivery-Portale zu nutzen. Darüber hinaus ist zu beobachten, dass die Einführung eines Redaktionssystems auch den Schreibstil von Technischen Redakteuren beeinflussen kann (BAILIE & HUSET, 2015).

Um (strukturierte) Bestandsdaten in das neue CMS zu migrieren, ist es meist notwendig, eine Klassifizierung der zu migrierenden Daten durchzuführen, was eine zeitaufwändige und arbeitsintensive Aufgabe darstellt. Eine mögliche Lösung zur Reduzierung des manuellen Arbeitsaufwands besteht darin, einen repräsentativen Teil des Korpus (z. B. $500 - 1000$ Module) als Trainingsdaten auszuwählen und die restlichen Inhalte automatisiert zu klassifizieren. Anhand eines Konfidenzgrenzwerts können Technische Redakteure dann Module überprüfen, bei denen die zugewiesene Klassifikation falsch sein könnte.

---

[56] In diesem Fall würde sich auch die Zurodnung im CoSMOS-Modell ändern von „Automatisierung“ zu „Assistenz“.

4.3.9.3 Unstrukturierte Dokumente

Eine weitere Anwendung, die auf der modulbasierten Klassifizierung basiert, ist die Segmentierung von unstrukturierten Bestandsdaten, wie z. B. PDF-Dokumenten, die für die Verwendung in Content-Delivery-Portalen aufbereitet werden sollen (OEVERMANN, 2016b).

Dieser Ansatz wird an anderer Stelle innerhalb dieser Arbeit ausführlich beleuchtet (siehe Kapitel 4.4: „Automatisierte Segmentierung").

4.3.9.4 Strukturierte Suche

Klassifizierte Inhalte können für eine facettierte Suche zur Verfügung gestellt werden, die es Anwendern ermöglicht, eine Volltextsuche nach der Informationsart weiter einzugrenzen.[57] Die Filterung nach Klassifikationen ist auch ein häufiger Anwendungsfall in Content-Delivery-Portalen, bei denen manche Filter automatisch gesetzt werden können (z. B. abgeleitet aus Serviceaufträgen), um den Anwendern Informationen zielgerichtet bereitstellen zu können. Weitere Anwendungen sind klassifikationsbasierte Ergebnisvorschläge oder kontextbezogene Suchen.

4.3.10 Ausblick

In zukünftigen Arbeiten soll die Forschung auf weitere Datensätze ausgedehnt werden und die Klassifizierung von Bestandsdokumenten genauer untersucht werden. Eine Erweiterung der Modelle hinsichtlich der Einbeziehung von grammatikalischen Mustern als Merkmale und der Vergleich mit alternativen Klassifizierungstechniken ist geplant. Des Weiteren sollen in zukünftiger Forschung auch alternative Merkmalsextraktions- und Gewichtungsmethoden wie AFE (siehe Abschnitt 4.3.3) einbezogen werden (BIRICIK u. a., 2009).

4.3.10.1 Sprachen

Globalisierte Unternehmen schreiben Inhalte meist in einer Quellsprache und übersetzen diese dann in mehrere Zielsprachen. Dieser Umstand ermöglicht es, die Klassifizierungsgenauigkeit für den gleichen Inhalt in verschiedenen Sprachen zu messen. Die Ergebnisse könnten Fragen darüber beantworten, ob einige Sprachen für eine automatisierte Klassifizierung auf Grundlage von statistischer Sprachverarbeitung besser geeignet sind als andere oder ob die Genauigkeit der Klassifizierung mit der Übersetzung abnimmt. Erste Experimente zur Merkmalsauswahl (siehe Abschnitt 4.3.6.1) konnten keine Korrelationen finden, weshalb die Versuche mit weiteren mehrsprachigen Datensätzen durchgeführt werden müssen.

---

[57] Als Beispiel: Ein Servicetechniker schränkt seine Volltextsuche auf die Klassifikation „Wartungsinformationen" und die Zielgruppe „Service" ein.

Wenn die Sprache des Inhalts die Genauigkeit der automatisierten Klassifizierung nicht beeinträchtigt, könnten die Ergebnisse zur Bewertung von Übersetzungen oder Übersetzungsanbietern verwendet werden.

Die zugrunde liegende Hypothese für ein solches Ranking lautet, dass eine gute Übersetzung die Genauigkeit der Klassifizierung nicht verändert. Daher könnte die Übersetzungsqualität automatisiert getestet werden, wobei erhebliche Einbußen bei der Genauigkeit als Indikator für schlechte Übersetzungen gedeutet werden können.

4.3.10.2 Linguistische Merkmale und Wortstellung

Derzeit werden die Merkmale für die Klassifizierung nur durch statistische Methoden gebildet und beinhalten keine linguistischen Eigenschaften wie z. B. die Verbform oder die Flexion eines Substantivs. Diese Informationen könnten zur Verbesserung der Klassifikationsergebnisse verwendet werden, insbesondere bei der Klassifizierung von Informationstypen der Technischen Kommunikation, bei denen grammatikalische Muster oft die Art des Inhalts vermitteln (z. B. anleitend oder beschreibend).

Ein weiterer wichtiger Aspekt grammatikalischer Muster ist die Reihenfolge der Wörter (LE & MIKOLOV, 2014). In der aktuellen Implementierung werden diese Informationen nur innerhalb von n-Grammen, nicht aber im Gesamtkontext von Modulen gespeichert. Die Verwendung der Position eines Wortes oder Wortmusters innerhalb eines Moduls als zusätzliches Merkmal könnte die Genauigkeit der Vorhersagen weiter verbessern.

4.3.10.3 Taxonomie-Fallback

In der aktuellen Implementierung werden bestehende hierarchische Beziehungen zwischen Klassen nicht ausgewertet oder verarbeitet. Bei Training und Klassifizierung werden stattdessen die Ebenen einer Taxonomie als jeweils separater Satz von Klassen betrachtet. Bei diesem Verfahren nimmt die Genauigkeit in untergeordneten Klassen tendenziell ab (vgl. Tabelle 9). Einer der Gründe für dieses Verhalten kann zum Beispiel in der zunehmenden semantischen und syntaktischen Ähnlichkeit benachbarter Klassen liegen, die es dem Klassifikator erschwert, zwischen den Klassen zu unterscheiden.

Insbesondere für Anwendungen im Information Retrieval (z. B. bei der facettierten Suche) kann es daher nützlich sein, über Fallback-Mechanismen zu verfügen. Diese könnten auf eine taxonomische übergeordnete Klasse zurückgreifen, wenn die Konfidenz des Klassifikators unter einen bestimmten Grenzwert fällt, und damit sicherstellen, dass der Recall für die klassenbasierte Filterung von Modulen hoch bleibt.

Ein solches Verhalten, das die Genauigkeit zugunsten der Usability reduziert, wird auch als „Graceful degradation“ (etwa: „elegante Verschlechterung“) bezeichnet (MENYCHTAS & KONSTANTELI, 2012) und ist eine Möglichkeit, den Anwendern auch unter schlechten Umständen praktikable Ergebnisse anbieten zu können.

Eine weitere Möglichkeit, das taxonomische Wissen zur Verbesserung der Genauigkeit zu nutzen, besteht darin, die Klassifikationsergebnisse übergeordneter Klassen miteinzubeziehen, wenn zwei klassenspezifischen Klassifizierungswerte s_c ähnliche Ergebnisse liefern (vgl. Abbildung 14, rechte Seite), aber unterschiedliche Elternklassen haben. Sollte die Konfidenz der übergeordneten Ebene höher sein, kann dieser Umstand auch zur Unterscheidung von Unterklassen verwendet werden.

4.3.10.4 Qualitätskennzahlen für Klassifikationsmodelle

Die Qualität des zugrunde liegenden Klassifikationsmodells beeinflusst die Ergebnisse der automatisierten Klassifikation stark. Wenn Klassen uneindeutig sind, kann das darauf zurückzuführen sein, dass die Trainingsdaten durch falsche manuelle Klassifizierung verzerrt wurden oder ähnlicher Inhalt zu verschiedenen Klassen gehört. In Klassifizierungsszenarien, die auf dem Vektorraummodell basieren, kann dies oft als abnormale Verteilung von Klassenvektoren beobachtet werden.

Mehrdeutige Klassen neigen dazu, ähnliche (Aus-)Richtungen im Vektorraum zu haben, während andere Klassen als fehlend identifiziert werden könnten, falls ein Klassifikator Inhalte regelmäßig zwischen zwei Klassenvektoren platziert. In zukünftiger Forschung soll ein Modell entwickelt werden, das solche Anomalien vorhersagen kann und dabei hilft, die Qualität eines Klassifikationsmodells auf der Grundlage eines klassifizierten Datensatzes zu messen.

Darüber hinaus führen klar definierte und eindeutige Klassifikationsmodelle sowie eine akkurate Klassifizierung durch Technische Redakteure zu einer Genauigkeit von nahezu 100 % bei einer sogenannten Selbstvalidierung (Training und Validierung mit dem gleichen vollständigen Datensatz). Obwohl ein dadurch verursachtes *Overfitting*[58] des trainierten Modells im Allgemeinen nicht erwünscht ist, kann dieses Verhalten jedoch auf andere Weise genutzt werden. Die Selbstvalidierung kann die allgemeine Qualität der Klassifizierung oder des gesamten Klassifizierungsmodells messen.

[58] Überanpassung des Modells auf den Datensatz, mit dem trainiert wurde.

Bei Versuchen im Rahmen der Arbeit konnte festgestellt werden, dass Klassifizierungsfehler in der Selbstvalidierung ein starker Indikator für eine falsche manuelle Klassifizierung oder ein uneindeutiges Klassifizierungsmodell sein können. Generierte Berichte, die Module enthalten, bei denen die automatisierte Klassifizierung im Rahmen einer Selbstvalidierung nicht mit der manuellen Klassifikation übereinstimmt, können helfen, falsch klassifizierte Module zu erkennen. In weiteren Arbeiten sollen diese Reports um Informationen über die vermutete Ursache der Abweichung erweitert werden (z. B. Probleme mit dem Klassifizierungssystem oder dem Inhalt des Moduls).

4.3.11 Fazit

Module in der Technischen Dokumentation haben spezielle Charakteristiken, die eine domänenspezifische Klassifizierungsmethode erforderlich machen. Die präsentierten Ergebnisse zeigen, dass ein angepasstes Vorgehensmodell für diesen Inhaltstyp die Genauigkeit bei Klassifizierungsaufgaben im Vergleich zu allgemeinen oder dokumentenzentrierten Ansätzen verbessern kann.

Wie in dieser Arbeit erläutert, gibt es mehrere praktikable Anwendungen, in denen eine automatisierte Klassifizierung nützlich oder sogar notwendig ist, insbesondere bei der Bereitstellung von sogenannten „Intelligenten Informationen“ (HENNIG & TJARKS-SOBHANI, 2017) im Bereich des Content Delivery (z. B. für Servicetechniker). Die durchgeführten Versuche haben gezeigt, dass PI-Klassifikationsmodelle dafür einen geeigneten Rahmen bieten und sich gut in die Prozesse des maschinellen Lernens integrieren lassen.

Erste Ergebnisse zeigen, dass verschiedene Klassifizierungsaufgaben mit der vorgestellten Methode gelöst werden können und dabei eine hohe Genauigkeit erzielt wird. Im Vergleich zu früheren Arbeiten wurde der Umfang auch auf produktbezogene Klassifikationen erweitert, die erfolgreich abgedeckt wurden.

Im Rahmen der Arbeit wurden mehrere Ansatzpunkte für domänenspezifische Optimierungen identifiziert und Anpassungen entwickelt, die eine Steigerung der Klassifikatorgenauigkeit zur Folge hatten. Diese Verbesserungen beinhalten die Kombination aus Wortgruppen ($n = \{1,2\}$ und $n = \{1,2,3\}$) als Merkmale für die Klassifizierung und eine angepasste Gewichtungsmethode, die klassenspezifische Charakteristiken berücksichtigt ($\mathrm{TF} - \mathrm{ICF} - \mathrm{CF}$). Durch zusätzliche Versuche konnte gezeigt werden, dass für leistungskritische Klassifizierungsaufgaben eine andere Merkmalsbildung optimal ist ($n = 2$).

Des Weiteren konnten Empfehlungen zum Einsatz von Lemmatisierungen, hierarchischen Klassifikationsmodellen, semantischen Gewichtungen und einem geeigneten Konfidenzmaß für kosinusbasierte Klassifikatoren gegeben

werden. Das Klassifikatorverhalten für verschiedene Klassenarten und -stufen innerhalb der PI-Klassifizierungsmethode wurde diskutiert und getestet. Abschließend wurden verschiedene potenzielle Anwendungen der Methode vorgestellt und Themen für die zukünftige Forschung skizziert.

Die in dieser Arbeit vorgestellten Anpassungen haben deutliche Verbesserungen gegenüber dokumentenorientierten oder allgemeineren Methoden ergeben und sind ein erster Schritt zu einer automatisierten Klassifizierung von Modulen in der Technischen Dokumentation.

4.4 Automatisierte Segmentierung

Der folgende Inhalt wurde zu großen Teilen aus dem Artikel „Semantic PDF Segmentation for Legacy Documents in Technical Documentation“ übernommen (OEVERMANN, 2018b), der wiederum auf den Vorarbeiten in OEVERMANN (2016b) basiert. An geeigneten Stellen wurden im Sinne der Gesamtarbeit Querverweise, Anmerkungen, Abbildungen oder Ergänzungen eingefügt und die Formatierung angepasst.

4.4.1 Zusammenfassung

Das meistgenutzte Dateiformat für die Speicherung und Bereitstellung von Technischer Dokumentation ist PDF. Aufgrund der Unstrukturiertheit des Formats sind diese Dokumente jedoch oft von einem granularen semantischen Zugriff ausgeschlossen. Während immer mehr Unternehmen XML-basierte Content-Management-Systeme einführen, die mit Metadaten versehene strukturierte Inhalte ausliefern können, bleiben ältere Bestandsdokumentationen in ihrer monolithischen Form erhalten.

In dieser Arbeit wird ein neuer Ansatz vorgestellt, der PDF-Dokumente mit Hilfe von vorhandenem Klassifikationswissen aus strukturierten Trainingsdaten in semantisch zusammenhängende Abschnitte unterteilen kann. Dieser, auf maschinellem Lernen basierende, Ansatz ist unabhängig von Formatierungen oder visuellen Informationen innerhalb des PDFs.

Die Ergebnisse mehrerer vorangegangener Arbeiten werden im Folgenden zu einem ganzheitlichen Vorgehensmodell zusammengefasst. Des Weiteren wird ein parametrisierbarer Suchalgorithmus für Segmente vorgestellt und zum Austausch der generierten Metadaten ein RDF-basiertes Format erzeugt, das innerhalb von Information-Retrieval-Szenarien ausgewertet werden kann, um Anwendern eine effizientere Informationsrecherche zu ermöglichen.

4.4.2 Einleitung

In den meisten regulierten Märkten, wie beispielsweise der EU, sind Hersteller gesetzlich dazu verpflichtet, die Technische Dokumentation ihrer Produkte über einen Zeitraum von bis zu 30 Jahren nach dem Inverkehrbringen aufzubewahren und bei Bedarf zur Verfügung zu stellen (2006/42/EG, 2006).[59]

Um diese gesetzlichen Anforderungen erfüllen zu können, speichern die meisten Unternehmen ihre Inhalte in Form von PDF-Dokumenten (STRAUB, 2016). Die Eignung des PDF-Formats für diese Art von Aufgaben basiert auf verschiedenen Eigenschaften, wie z. B. dem klaren Fokus auf einer exakten

[59] Die EU-Maschinenrichtlinie schreibt eine Aufbewahrungspflicht der Technischen Unterlagen von mindestens 10 Jahren „nach dem Tag der Fertigstellung der letzten Einheit“ eines Produkts gesetzlich vor (vgl. 2006/42/EG, Anhang VII, A2).

visuellen Reproduktion[60] der Inhalte, dem Umstand, dass PDF-Dokumente keine einfachen Änderungen am Inhalt zulassen und der weitreichenden Softwareunterstützung, die auf maximale Kompatibilität zielt (vgl. HASSAN, 2018). Gleichzeitig sind die Hersteller mit den Anforderungen ihrer Kunden konfrontiert, die eine zeitgemäße Informationsbereitstellung erwarten, z. B. über Content-Delivery-Portale (CDP) mit Such- und Filterfunktionen. Diese Unternehmensportale[61] gewinnen als primäre Methode der Informationsrecherche für Nutzer immer mehr an Popularität (ZIEGLER & BEIER, 2014).

Während moderne PDF-Versionen einige Funktionalitäten für einen strukturierteren Inhaltszugriff bereitstellen (z. B. *getaggte* PDF-Dateien, vgl. ADOBE SYSTEMS (2001)), sind diese Formatvarianten in der Praxis eher ungewöhnlich (CHAO & FAN, 2004) oder verwertende Portale nutzen diese Informationen aufgrund technischer Einschränkungen nicht. Einige Methoden können große PDF-Dokumente in mehrere kleinere Dateien aufteilen[62], jedoch sind diese Ansätze für den Bereich der Technischen Dokumentation eher ungeeignet, da einmal freigegeben Dateien aus rechtlichen Gründen nicht mehr verändert werden sollen. Dieser Umstand führt dazu, dass ältere PDF-Dokumente vom granularen semantischen Zugriff ausgeschlossen und im Informationszugriff auf eine Volltextsuche oder manuelles Recherchieren beschränkt sind. Der in dieser Arbeit vorgestellte Ansatz nutzt die Tatsache aus, dass viele Unternehmen mittlerweile XML-basierte Redaktionssysteme zur Erstellung, Verwaltung und Veröffentlichung ihrer Technischen Dokumentation verwenden (STRAUB, 2016). In Content-Management-Systemen werden Texte modular erfasst, was eine kosteneffiziente Wiederverwendung und eine einfache Zusammenstellung von Dokumenten ermöglicht. Module oder Topics werden dabei oft mit klassifizierenden Metadaten angereichert, um bestimmte semantische Eigenschaften des Textes zu beschreiben (z. B. den Informationstyp oder die beschriebene Produktkomponente). Diese (aus Sicht des Maschinellen Lernens) hochwertigen Inhalte können als Trainingsdaten für ein überwachtes Lernen verwendet und das daraus gewonnene Klassifizierungsmodell auf die Inhalte älterer PDFs angewendet werden, um semantisch zusammenhängende Segmente zu rekonstruieren.

---

[60] insbesondere bei der PDF/A-Variante (A = Accessible), bei der die Langzeitarchivierung und der garantierte Zugriff auf die Informationen im Vordergrund steht.

[61] Für eine Übersicht der verschiedenen Typen von Unternehmensportalen, zu denen auch die Content-Delivery-Portale gehören, siehe ZIEGLER & BEIER (2014).

[62] Diese Ansätze beruhen in der Regel auf Seitenangaben aus dem Inhaltsverzeichnis oder Formatierungsinformationen wie Überschriften, Absätze oder Kapitelumbrüche.

Um das Wissen aus klassifizierten Modulen auf unstrukturierte Dokumente zu übertragen, wird der aus dem PDF extrahierte Text in kleine Einheiten – die sog. „Chunks“[63] – aufgeteilt, auf die das Modell dann angewendet werden kann. Dieser Prozess wird im Folgenden als *Chunking* bezeichnet. Die zugrunde liegende Hypothese besagt, dass Chunks, die Überschneidungen verschiedener semantischer Segmente enthalten, im automatisierten Klassifizierungsprozess schlechter abschneiden (in Bezug auf die Konfidenz des Klassifikators) als welche, die nur eine semantisch relevante Inhaltsart enthalten (vgl. Abbildung 15 für eine Visualisierung der Hypothese).

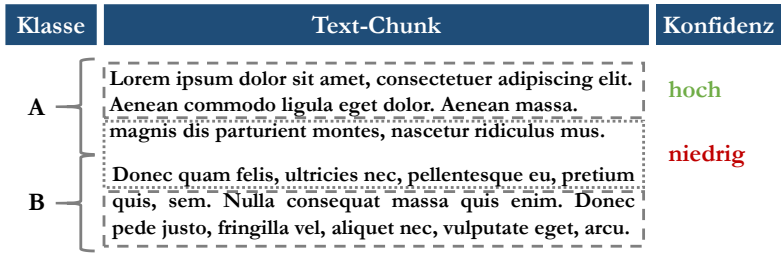

Abbildung 15: Visualisierung der Chunking-Hypothese.
Die zufällige Verteilung der Chunks über den Text ergibt Übereinstimmungen mit Klassengrenzen (hohe Konfidenz) aber auch uneindeutige Klassifikationen (niedrige Konfidenz). Diese Annahme ist die Grundlage für den Segmentierungsansatz. Abbildung basierend auf einer Grafik aus Oevermann (2016b).

Basierend auf den einzelnen Klassifizierungsergebnissen pro Texteinheit und den zugehörigen Konfidenzwerten wird ein Algorithmus zur Abschnittserkennung angewendet, um vorhandene Segmente im Dokument zu rekonstruieren. Um diese generierten Metadaten in einer Vielzahl von Content-Delivery-Portalen verwenden zu können, werden sie in den unabhängigen Standard iiRDS („intelligent information – Request and Delivery Standard“) überführt, ein Austauschformat für digitale Technische Dokumentation. iiRDS wird aktuell von einem privaten Konsortium[64] verschiedener Firmen unter der Schirmherrschaft der tekom – dem Berufsverband der Technischen Redakteure – entwickelt. iiRDS kombiniert ein Paketformat und ein standardisiertes RDF-basiertes Vokabular mit relevanten Metadaten für die Technische Dokumentation (die teilweise von der PI-Klassifizierungsmethode abgeleitet wurden (PARSON u. a., 2017)). iiRDS ist seit April 2018 in der Version 1.0 verfügbar und kann frei genutzt werden (BECKER u. a., 2018).

[63] „Chunk“ bedeutet in diesem Sinne so viel wie Textstück oder Textschnipsel.

[64] Offizielle Website des iiRDS-Konsortiums: https://iirds.org/ [01.02.19].

Da bei dem vorgestellten Prozess nur zusätzliche Metadaten produziert werden, die neben oder mit der Original-PDF-Datei geliefert und verarbeitet werden, sind davon keine dateibasierten Freigabeprozesse betroffen, die im Bereich der Technischen Dokumentation relativ verbreitet sind.

Diese Arbeit präsentiert ein spezielles Vorgehensmodell zur semantischen PDF-Segmentierung von Bestandsdokumenten in der Technischen Dokumentation. Basierend auf früheren Arbeiten wird die Methode um eine verfeinerte Abschnittserkennung, eine standardisierte Metadatengenerierung und eine Validierung mit realen Daten erweitert. Darüber hinaus werden die bisherigen Ansätze zu einer vollständigen Beschreibung des Verfahrens und einer Demo-Implementierung kombiniert.

4.4.3 Verwandte Arbeiten

Die Segmentierungs-Methode, auf der diese Arbeit basiert, wurde erstmals in OEVERMANN (2016b) vorgestellt. Dort wurden in einem kurzen Konferenzbeitrag die zugrunde liegende Hypothese, ein Chunking-Algorithmus und erste Versuche mit realen Daten beschrieben.[65]

In BADER & OEVERMANN (2017) wurde dieser Prozess teilweise um eine simple Abschnittssuche und Metadatengenerierung auf Basis des „W3C Web Annotation"-Standards erweitert. Die Methodik zur automatisierten Klassifizierung von Modulen wurde in OEVERMANN & ZIEGLER (2018) ausführlich beschrieben und bewertet (siehe Abschnitt 4.3).

In BRANTS, CHEN & TSOCHANTARIDIS (2002) wird eine Technik der Dokumentensegmentierung präsentiert, die auf der *Probabilistischen latent-semantischen Analyse* (PLSA) basiert.[66] Die Arbeit ähnelt dem hier vorgestellten Ansatz bei der Zuordnung von Themen zu verschiedenen Teilen eines Dokuments basierend auf Einbrüchen („Dips") in den Ähnlichkeitswerten benachbarter Blöcke. Die Methode unterscheidet sich jedoch in Bezug auf die verwendeten Features (PLSA vs. n-Gram), die Wahl der „elementaren Blöcke" (Sätze vs. Module) und die relevante Messgröße für die Abschnittserkennung (Ähnlichkeit vs. Konfidenz).

In MCDONALD, CRAMMER & PEREIRA (2005) stellen die Autoren ein Modell für die Segmentierung von Texten auf Basis einer Multilabel-Klassifikation vor (hier: Multiklassen-Klassifikation). Der dort präsentierte Ansatz klassifiziert auf der Token-Ebene im Gegensatz zur Modul- bzw. Segmentebene.

---

[65] Die Ergebnisse dieser ersten Versuche sind in Abschnitt 4.4.10 zu finden.

[66] Die latent-semantische Analyse wird auch in Abschnitt 3.2.3 beschrieben.

4.4.4 Methodologie

4.4.4.1 Übersicht

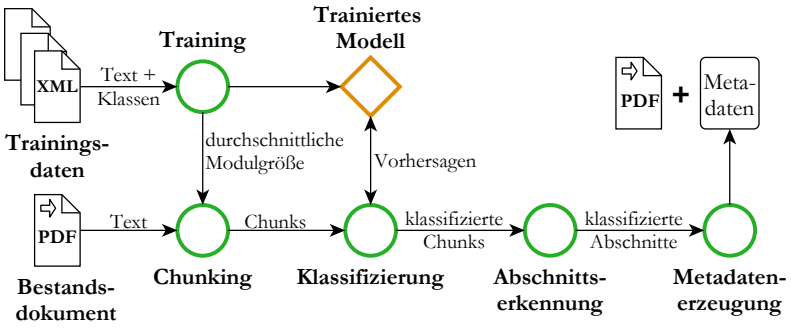

Abbildung 16: Schematischer Datenfluss im Segmentierungsprozess.

Der Text eines PDF-Dokuments wird auf der Grundlage von XML-Trainingsdaten zerteilt (*Chunking*) und klassifiziert. Ausgehend von den vorhergesagten Klassifikationen kann ein Algorithmus zur Abschnittserkennung die semantischen Segmente in einem standardisierten Metadatenformat bereitstellen (vgl. Abbildung 16).

4.4.4.2 Voraussetzungen

Als Trainingsdaten werden Inhalte aus einer Technische Produktdokumentation aus dem Maschinenbaubereich verwendet, die als strukturierte XML-Module (zwischen 50 und 500 Wörter pro Modul) bereitgestellt wurden (siehe auch OEVERMANN & ZIEGLER, 2018). Die Trainingsdaten wurden nachträglich mit iiRDS-Konzepten versehen, die auf Basis der bestehenden Klassifikation systematisch zugeordnet wurden. Dafür wurden ausschließlich intrinsische Konzepte aus den iiRDS-Metadaten verwendet, da diese direkt aus den Texteigenschaften abgeleitet werden können und sich deshalb für eine statistische Merkmalsextraktion eignen (vgl. OEVERMANN & ZIEGLER, 2016). Beispiele für diese Konzepte sind Instanzen aus den folgenden Klassen:

- **`iirds:TopicType`**, z. B. Aufgabe (*Task*), Konzept (*Concept*) oder Referenz (*Reference*).
- **`iirds:InformationSubject`** (Informationsthema), z. B. Technische Daten, Sicherheitsinformationen oder Prozessbeschreibungen.
- **`iirds:ProductLifeCyclePhase`** (Produktlebenszyklusphase), z. B. Betrieb, Wartung oder Reparatur.
- **`iirds:Component`** (Baugruppe/Komponente), dient als Anknüpfpunkt für unternehmensspezifische Produktontologien.

In den meisten Fällen können andere etablierte Klassifikationsmethoden wie PI-Class® (ZIEGLER, 2015) oder eCl@ss (HEPP u. a., 2007) über ein Mapping direkt auf diese iiRDS-Konzepte abgebildet werden. Die aufgeführten iiRDS-Konzepte enthalten semantisch relevante Informationen über den Inhalt selbst, weshalb sie sich für eine semantische Segmentierung eignen. Ziel der Methode ist es, Metadaten zu semantisch zusammenhängenden Segmenten im PDF-Dokument zu generieren, die als Seitenbereiche definiert sind (z. B. „Wartungsinformationen finden Sie auf den Seiten 75-120“).

Als Testdaten werden unstrukturierte PDF-Dokumente mit Technischer Dokumentation verwendet, die den gleichen Produkttyp aus dem gleichen Unternehmen behandelt wie die Trainingsdaten. Obwohl die vorgestellte Implementierung auf PDFs mit extrahierbarem Text basiert, kann die Methode auch auf gescannte PDFs angewendet werden, bei denen der Text über eine OCR[67]-Vorverarbeitung generiert wird (z. B. für Bestandsdokumente, die nur als Scans von Papierkopien existieren).

Um die Genauigkeit der Segmentierung zu beurteilen, wird für jedes Dokument eine *Baseline* definiert, die auf dem Inhaltsverzeichnis[68] und einer manuellen Bewertung basiert. Diese Referenzstruktur des Dokuments wird in Versuchen dann mit verschiedenen Segmentierungsansätzen und Parameter-Konfigurationen verglichen.

4.4.4.3 Training

Der Text wird aus den XML-basierten Modulen der Trainingsdaten extrahiert und normalisiert. Zur Merkmalsextraktion wird ein „Bag of n-Grams“-Modell mit $n = 2$ verwendet, bei dem die generierten n-Gramme nach der in OEVERMANN & ZIEGLER (2016) eingeführten $\mathrm{TC} - \mathrm{ICF} - \mathrm{CF}$-Methode gewichtet werden, um den Prozess an die Textcharakteristiken von Technischer Dokumentation anzupassen. Für jede Klasse in den Trainingsdaten wird ein prototypischer Vektor gebildet, der die normalisierten gewichteten Merkmale als Vektorkomponenten enthält.

Aus diesen Vektoren wird das trainierte Modell M mit allen Klassenvektoren (OEVERMANN & ZIEGLER, 2018) erstellt. Während der Trainingsphase werden auch Informationen über die durchschnittliche Größe der Module gewonnen, im Folgenden bezeichnet als $a\ [\frac{\text{Wörter}}{\text{Modul}}]$.

---

[67] OCR, engl. „Optical Character Recognition“, Optische Zeichenerkennung (z. B. bei Texten, die in Bildern vorkommen). Wird oft in Scannern oder zugehöriger Software eingesetzt.

[68] In der TD orientieren sich Inhaltsverzeichnisse in der Regel an einer Kombination aus **`iirds:InformationSubject`** und **`iirds:ProductLifeCyclePhase`**.

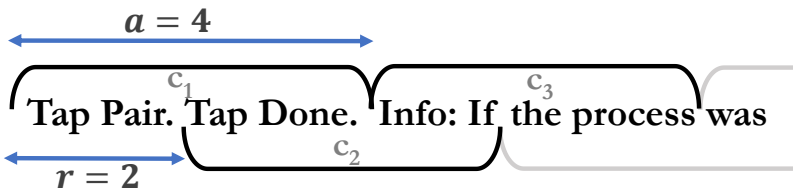

Abbildung 17: Chunking-Methodologie.
c_i ist ein generierter Chunk, Parameter a ist die Größe des Chunks als Wortanzahl (typischerweise zwischen 50–200 Wörtern), Parameter r ist der Versatz, mit dem Chunks generiert werden als Wortanzahl (typischerweise ein Bruchteil von a, z. B. 40 Wörter).

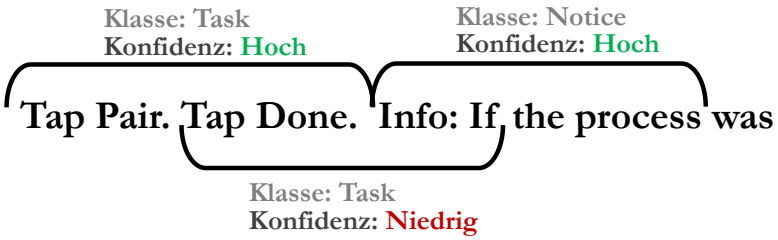

Abbildung 18: Beispiel für Konfidenz-Hypothese.
Verschiedenen Informationstypen in Kombination mit der gemessenen Konfidenz für die jeweiligen Chunks. Der Text wurde aus einem Benutzerhandbuch entnommen, bei dem auf mehrere Schritte einer Aufgabe eine Infobox folgt.

4.4.4.4 Chunking

Der Inhalt eines PDF-Dokuments wird als reiner Text extrahiert. Alle Formatierungsinformationen werden verworfen, da sie für ältere Dokumente oft nicht verfügbar oder unzuverlässig sind (z. B. nach einer OCR-Texterkennung). Darüber hinaus ist die Rekonstruktion von Absätzen, Überschriften oder zusammenhängenden Sätzen ein weitverbreitetes Problem bei der PDF-Verarbeitung (FANG u. a., 2011). Da diese Bestandsdaten im Mittelpunkt der Arbeit stehen, wurde bewusst auf einen Ansatz gesetzt, der nur auf reinem Text basiert. Die extrahierte Zeichenkette wird anschließend auf Basis von Wortgrenzen (Leerzeichen, Interpunktion etc.) in einzelne Wörter aufgeteilt, welche die Menge der extrahierten Wörter W ergeben. Aus der Menge der extrahierten Wörter W werden arbiträre Chunks $C = c_1, c_2, \ldots c_n$ gebildet, wobei $c_i \subset W$ gilt. Die Größe der Chunks basiert auf der zuvor gesammelten durchschnittlichen Wortanzahl in Modulen a. Der Wert für a wurde als geeignete Chunk-Größe gewählt, da er der typischen Größe (und damit auch

Merkmalsverteilung) einer Informationseinheit[69] für diesen spezifischen Produktinhalt entspricht, was wiederum für eine hohe Klassifizierungsgenauigkeit entscheidend ist. Um Text-Chunks über den Dokumentinhalt verteilt zu generieren, wird eine natürliche Zahl r als Versatz (*Offset*) mit $r \leq a$ definiert (sog. *Sliding window*). Diese Verschiebung definiert, wie sich mehrere Chunks überschneiden (vgl. Abbildung 17). Daher kann ein Chunk c_i an der Position i (für $i > 1$) definiert werden als:

$$c_i = \{W_{(i-1)\cdot r}, W_{(i-1)\cdot r+1}, \ldots, W_{(i-1)\cdot r+a}\}$$

Die Gesamtzahl der Chunks $|C|$, die für eine gegebene Menge von Wörtern W in Abhängigkeit von Größe (a) und Versatz (r) der Chunks erzeugt werden, kann wie folgt berechnet werden (OEVERMANN, 2016b):

$$|C| = \left\lfloor \frac{|W| - a}{r} \right\rfloor$$

Ein kleiner Wert für r erhöht die Gesamtzahl der Chunks und damit die Auflösung der Segmentsuche, hat aber auch negativen Einfluss auf die Leistung des Klassifikators. Der Versatz kann z. B. als Bruchteil der durchschnittlichen Modulgröße gewählt werden. Für die Versuche wurde ein Standardwert von $r = \left\lfloor \frac{1}{4} a \right\rfloor$ gewählt. Offsets, die kleiner als dieser Wert sind, bieten keine wesentlichen Vorteile bei der Interpretation der Ergebnisse, erhöhen aber die benötigte Rechenzeit. Für jeden generierten Chunk wird der Textinhalt, die Größe und die Position (bezogen auf die PDF-Seite, Seitenzahl und Gesamtanzahl der Zeichen) gespeichert.

4.4.4.5 Klassifizierung und Konfidenz

Alle Chunks in C werden klassifiziert, indem die in Abschnitt 4.4.4.3 beschriebene Merkmalsextraktion angewandt und die Kosinusähnlichkeit für alle Klassenvektoren in M berechnet wird. Zusätzlich wird für die Vorhersage (die Klasse mit dem höchsten Ähnlichkeitswert s in n Klassen) ein Konfidenzwert p wie folgt berechnet:[70]

$$p = \frac{s_1 - s_2}{s_1 - s_n}$$

[69] Eine Informationseinheit ist die verarbeitete Textgröße. In der Trainingsphase ein Modul und in der Segmentierungsphase ein Chunk.

[70] Für die klassenspezifische Ähnlichkeitswerte die von hoch (s_1) nach niedrig (s_n) sortiert wurden. Siehe auch den Abschnitt 4.3.6.5 zur Konfidenzmessung bei der automatisierten Klassifizierung.

Eine typische Verteilung der Konfidenzwerte (y-Achse) und Vorhersagen (Farbe) für mehrere Chunks über die Seiten eines Dokuments (x-Achse) ist in Abbildung 19 dargestellt. Weitere Details zum Konfidenzwert werden an anderer Stelle in dieser Arbeit behandelt (siehe Ausführungen im Abschnitt 4.3.6.5 und die Evaluierung in Abschnitt 4.3.7.4).

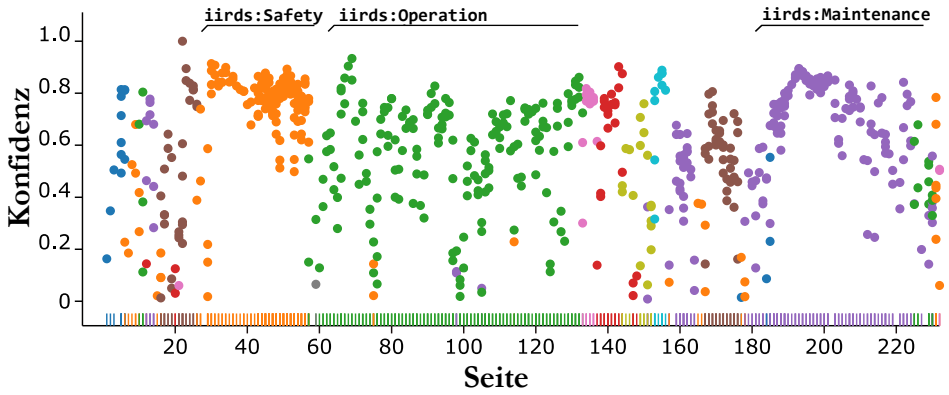

Abbildung 19: Beispiel- Segmentierung mit iiRDS-Annotationen.
Die Klassifizierungsergebnisse werden mit Konfidenz p (y-Achse), Seitenposition (x-Achse) und vorhergesagter Klassifikation (Farbe) dargestellt. Drei Beispielsegmente, die aus diesen Daten abgeleitet werden konnten, wurden manuell annotiert. Ausreißer sind deutlich erkennbar.
Abbildung basierend auf einer Grafik aus Bader & Oevermann (2017).

Die vorhergesagte Klassifikation und der zugehörige Konfidenzwert p werden mit den generierten Chunks gespeichert. Durch die Analyse des Konfidenzverlaufs im Dokument kann bestimmt werden, wo die Grenzen zwischen den Segmenten verschiedener Klassifikationen liegen (zusammenhängende Module gleicher Klassifikation bilden ein Segment).

Die zugrundeliegende Annahme dafür ist, dass bei der arbiträren Generierung der Chunks, einige Chunks nur Text aus einer (semantischen) Klasse enthalten werden, während andere Chunks Text aus mehreren Klassen enthalten (vgl. Abbildung 18 für eine Visualisierung dieses Szenarios). Der erste Fall führt zu einer hohen Konfidenz, da die Ähnlichkeit zu genau einer Klasse bei der Klassifikation eines Moduls hoch sein sollte. Im anderen Fall werden nach der Klassifizierung mehrere Klassen höhere Ähnlichkeitswerte und damit eine insgesamt geringere Konfidenz (vgl. auch Abbildung 14) verursachen. Diese Einbrüche in der Konfidenzkurve („Dips“ oder lokale Minima) über das Dokument verteilt (vgl. oberer Teil von Abbildung 20) sind gute Indikatoren für Grenzen zwischen verschiedenen Segmenten.

4.4.4.6 Abschnittserkennung

Vorherige Versuche zur Abschnittserkennung basierten auf der Gruppierung von Chunks mit derselben Klassifikation zu Segmenten, ohne die Information zur Konfidenz mit einzubeziehen (BADER & OEVERMANN, 2017). Dieser Ansatz ist jedoch anfällig für häufig auftretende Ausreißer, die größere Segmente in mehrere kleine Abschnitte aufteilen. Mit dem hier erläuterten Ansatz sollen Makrostrukturen wie Kapitel oder Dokumentteile rekonstruiert werden, die Informationen über einen bestimmten Teil des Produkts enthalten oder einer bestimmten Informationsart angehören.

Die vorgestellte Abschnittserkennung basiert auf den Eigenschaften der Konfidenzkurve und produziert bessere Ergebnisse für Echtdaten als die vorherigen Ansätze (vgl. die Ausführungen in Abschnitt 4.5.5.2.2). In diversen Versuchen konnte eine erhebliche Steigerung der Genauigkeit von Segmentdefinitionen erzielt werden (siehe Abschnitt 4.4.5).

Der Algorithmus zur Abschnittserkennung besteht aus den folgenden Schritten (visualisiert in Abbildung 20):

1. Finde alle lokalen Minima ($p_{i-1} \geq p_i \leq p_{i+1}$) in den Konfidenz-werten der in $\mathbf{C}$ enthaltenen Elemente, die unterhalb des Grenzwerts $\mathrm{p_{minima}}$ liegen und füge sie der Menge $\mathbf{N}$ hinzu (wobei gilt $\mathbf{N} \subset \mathbf{C}$).
2. Gruppiere die in $\mathbf{N}$ enthaltenen Elemente, die näher zueinander liegen als in $\mathrm{i_{threshold}}$ definiert und füge sie der Menge $\mathbf{N_{clustered}}$ hinzu (wobei gilt $\mathbf{N_{clustered}} \subset \mathbf{N}$).
3. Definiere die Menge von Abschnitten $\mathbf{R}$, die alle Elemente von $\mathbf{C}$ enthält, die zwischen den Elementen von $\mathbf{N_{clustered}}$ liegen und eine Mindestlänge von $\mathrm{r_{threshold}}$ haben.
4. Ermittle die dominante Klassifikation für jedes Element in $\mathbf{R}$ über den höchsten Median der Konfidenz (basierend auf den Konfidenzwerten der Vorhersagen der Chunks, die sich innerhalb des betrachtetet Abschnitts $\mathrm{R_i}$ befinden) und füge alle Elemente mit einer höheren mittlere Konfidenz als $\mathrm{p_{range}}$ der Menge $\mathbf{R_{labeled}}$ hinzu (wobei gilt $\mathbf{R_{labeled}} \subset \mathbf{R}$).
5. Gruppiere die in $\mathbf{R_{labeled}}$ enthaltenen und aufeinanderfolgenden Elemente mit der gleichen dominanten Klassifikation und füge sie der Menge $\mathbf{R_{clustered}}$ hinzu.

Die Elemente $\mathrm{R_i}$ (in diesem Fall Segmentdefinitionen), die sich nach der Durchführung des Algorithmus in der Menge $\mathbf{R_{clustered}}$ befinden, sind die Ergebnisse der Abschnittserkennung.

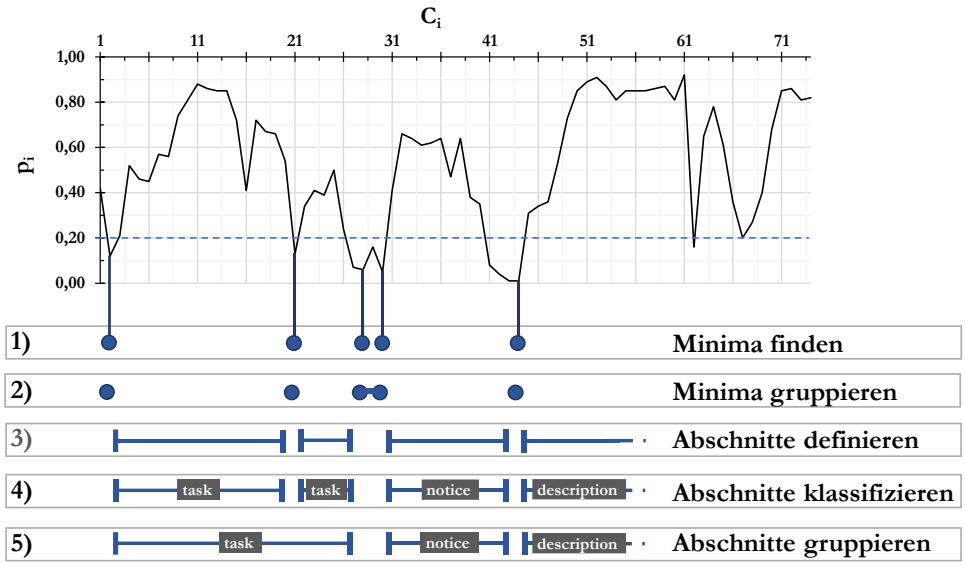

Abbildung 20: Beispiel für Konfidenzkurve mit Algorithmus.
Basierend auf dem Textauszug einer PDF-Dokumentation mit manuell abgeleiteten Schritten der Abschnittserkennung. Für dieses Beispiel wurden die folgenden Parameter verwendet: $p_{minima} = 0{,}20$ *und* $i_{threshold} = 10$.
Das Chunking wurde mit $a = 172$ *und* $r = 42$ *durchgeführt (aus den Traininsgdaten abgeleitet). Die Kurve zeigt die Chunks* $1 - 75$*, die die Seiten* $1 - 24$ *des Dokuments abdecken.*
Abbildung basierend auf einer Grafik aus Oevermann (2018b).

4.4.4.7 Metadatenerzeugung

Um die generierten Metadaten für das jeweilige PDF-Dokument (Segmente als Seitenbereiche mit entsprechend zugewiesenen Klassifikationen) auszudrücken, kann das Vokabular des iiRDS RDF Schemas verwendet werden. Der Standard basiert auf Informationseinheiten (**`iirds:InformationUnit`**), die als abstrakte Klasse für die Kombination von Metadaten und Inhalten dienen. Jede dieser Informationseinheiten kann mehrere physische Ausführungen (**`iirds:Rendition`**) haben, um den gleichen Inhalt in verschiedenen Zielformaten bereitzustellen. Eine Ausführung kann direkt auf eine Datei zeigen oder durch eine Selektor-Logik (**`iirds:Selector`**) verfeinert werden, um nur einen bestimmten Bereich oder eine bestimmte Position in der referenzierten Datei zu definieren (vgl. auch BECKER u. a., 2018, Abschn. 6.3.1).

Zum Ausdrücken von Abschnitten innerhalb eines PDF-Dokuments müssen daher Instanzen von **`iirds:Fragment`** erzeugt werden (welches die am besten geeignete Art von Informationseinheit ist). Pro Segment in $\mathbf{R}_{\text{clustered}}$ wird ein Fragment erzeugt, das die dominante Klassifikation als zugewiesenes

Konzept sowie eine Referenz auf die PDF-Datei enthält, die von einem Seitenbereichs-Selektor weiter verfeinert wird (siehe Codebeispiel 5). Die für das Dokument erzeugten Instanzen von **`iirds:Fragment`** können dann innerhalb eines **`iirds:Package`** zusammengefasst werden und zusammen mit der zugehörigen PDF-Datei verpackt als *iiRDS ZIP Container* publiziert werden.[71] Alle Softwaresysteme, die den Standard in der Version 1.0 voll unterstützen, können nun ein solches iiRDS-Paket verarbeiten und die definierten Segmente interpretieren (z. B. bei einer facettierten Suche).

```
<iirds:Fragment rdf:about="urn:uuid:0b86fd8a-[…]">
  <iirds:has-subject
     rdf:resource="http://iirds.tekom.de/iirds#Safety"/>
  <iirds:has-rendition>
    <iirds:Rendition>
      <iirds:format>application/pdf</iirds:format>
      <iirds:source>files/manual.pdf</iirds:source>
      <iirds:has-selector>
          <iirds:RangeSelector>
              <iirds:has-start-selector>
                  <iirds:FragmentSelector>
                      <dcterms:conformsTo
                          rdf:resource="[…]/rfc3778"/>
                      <rdf:value>page=15</rdf:value>
                  </iirds:FragmentSelector>
              </iirds:has-start-selector>
              <iirds:has-end-selector>
                  <iirds:FragmentSelector>
                      <dcterms:conformsTo
                         rdf:resource="[…]/rfc3778"/>
                      <rdf:value>page=63</rdf:value>
                  </iirds:FragmentSelector>
              </iirds:has-end-selector>
          </iirds:RangeSelector>
      </iirds:has-selector>
    </iirds:Rendition>
  </iirds:has-rendition>
</iirds:Fragment>
```

Codebeispiel 5: Semantische Annotation eines Abschnitts in iiRDS.
Eine erzeugte Instanz von **`iirds:Fragment`** *für einen beispielhaften Abschnitt: Inhalt mit dem zugewiesenen Konzept „Sicherheit“ (*`iirds:Safety`*) kann im Seitenbereich* 15 – 63 *innerhalb des referenzierten PDFs gefunden werden.*

[71] Ein solche Container enthält im wesentlichen die PDF-Dateien, die referenziert werden sowie eine zentrale RDF-Metadatendatei nach dem iiRDS RDFS, die alle erzeugten Instanzen enthält.

4.4.5 Evaluierung

Für eine erste Bewertung des vorgestellten Segmentierungsansatzes wurden Echtdaten aus der Technischen Dokumentation eines Unternehmens im Bereich Schwermaschinenbau verwendet. Die XML-basierten Trainingsdaten enthalten 663 Module mit einer durchschnittlichen Größe (a) von 162 Wörter pro Modul in deutscher Sprache.

Jedes Modul im Trainingsdatensatz wurde von Technischen Redakteuren eines von zehn iiRDS-Konzepten zugeordnet[72] (in alphabetischer Reihenfolge: **`Assembly`**, **`EmergencyOperation`**, **`Formality`**, **`Maintenance`**, **`Operation`**, **`Safety`**, **`TechnicalData`**, **`TechnicalOverview`**, **`Transport`**, **`Troubleshooting`**).

Bei einer 10-fachen Kreuzvalidierung hat dieser Datensatz eine Klassifizierungsgenauigkeit (*accuracy*) von 81,7 %. Das PDF-Dokument, mit dem die Methode getestet wurde, ist ein unverändertes Benutzerhandbuch mit 234 Seiten Inhalt, der die gleiche Produktgruppe wie die Trainingsdaten behandelt. Die grundlegende Struktur des Dokuments (die auch als Baseline verwendet wurde) wird in Tabelle 11 dargestellt (vgl. auch Abschnitt 4.4.4.2).

Die in BADER & OEVERMANN (2017) vorgestellte *einfache Abschnittserkennung* erzeugt 59 Segmente, die sich nur teilweise mit der Baseline überschneiden. Wie in Abschnitt 4.4.4.6 beschrieben, wird die hohe Anzahl der Segmente durch Ausreißer in der Klassifizierung einzelner Chunks verursacht (falsch positive Ergebnisse). Diese hohe Anzahl von Segmenten ist für den Einsatz im Information Retrieval ungünstig, da sie nicht der erwarteten Struktur der Technischen Dokumentation entspricht.

Zur Evaluierung der präsentierten Methode wurden die folgenden Parameterwerte gewählt (Verweise auf zugehörige Erläuterungen in Klammern):

- Versatz beim Chunking: $r = \left\lfloor \frac{1}{4} a \right\rfloor = 40$ Wörter (vgl. 4.4.4.4)
- Grenzwert für Konfidenz: $p_{minima} = 0{,}2 = 20\ \%$ (vgl. 4.4.4.6/**1**)
- Mindestabstand der lokalen Minima [in Anzahl Chunks]: $i_{threshold} = 5$ (vgl. 4.4.4.6/**2**)
- Mindestlänge Abschnitt [in Anzahl Chunks]: $r_{threshold} = 5$ (vgl. 4.4.4.6/**3**)
- Grenzwert mittlere Konfidenz: $p_{range} = 0{,}5 = 50\ \%$ (vgl. 4.4.4.6/**4**)

[72] Auf Deutsch: Montage, Notbetrieb, Formalität, Wartung, Betrieb, Sicherheit, Technische Daten, Technische Übersicht, Transport, Fehlersuche. Jeweils im iiRDS-Namensraum (**`iirds:*`**).

Tabelle 11: Baseline für Segmentierung.
Manuell ausgewertete Struktur des analysierten Dokuments für Abschnitte R_i.

R_i [i]	Anfang [Seite]	Ende [Seite]	iiRDS-Konzept
1	1	8	**`iirds:Formality`**
2	9	14	Inhaltsverzeichnis (nicht in Trainingsdaten)
3	15	28	**`iirds:TechnicalOverview`**
4	29	58	**`iirds:Safety`**
5	59	119	**`iirds:Operation`**
6	120	134	Ausrüstung (nicht in Trainingsdaten)
7	135	138	**`iirds:Transport`**
8	139	146	**`iirds:EmergencyOperation`**
9	147	166	**`iirds:Troubleshooting`**
10	167	179	**`iirds:TechnicalData`**
11	180	228	**`iirds:Maintenance`**
12	229	234	Index (nicht in Trainingsdaten)

Dies sind die Standardwerte für den Algorithmus zur Abschnittserkennung, basierend auf den Gesamtergebnissen von Versuchen mit verschiedenen Dokumenten. Durch die Anpassung von p_{minima} in Abhängigkeit von der allgemeinen Konfidenzverteilung können bessere Ergebnisse für einzelne Dokumente erzielt werden.

Der hier vorgestellte erweiterte Algorithmus zur Abschnittserkennung liefert Ergebnisse, die nahe an der Baseline-Segmentierung liegen (vgl. Tabelle 12 und Abbildung 21). Einige der Abweichungen von der Baseline lassen sich durch Rundungsfehler bei der Berechnung von Seitenpositionen für bestimmte Chunks erklären, die durch Probleme bei der Zeichenverarbeitung in JavaScript verursacht werden.

Tabelle 12: Ergebnisse der automatisierten Segmentierung.
Ermittelte Abschnitte R_i für das analysierte Dokument mit Annotationen.
Die Abschnitte 13 und 14 wurden fälschlicherweise erkannt, die Abschnitte 2, 6 und 12 wurden nicht erkannt, da sie nicht in den Trainingsdaten enthalten waren.

R_i [i]	Anfang [Seite]	Ende [Seite]	iiRDS-Konzept
1	1	9	**iirds:Formality**
3	16	23	**iirds:TechnicalOverview**
4	24	55	**iirds:Safety**
5	62	126	**iirds:Operation**
7	126	145	**iirds:Transport**
8	145	151	**iirds:EmergencyOperation**
9	155	159	**iirds:Troubleshooting**
10	160	174	**iirds:TechnicalData**
11	174	225	**iirds:Maintenance**
13	231	232	**iirds:Troubleshooting**
14	232	234	**iirds:Maintenance**

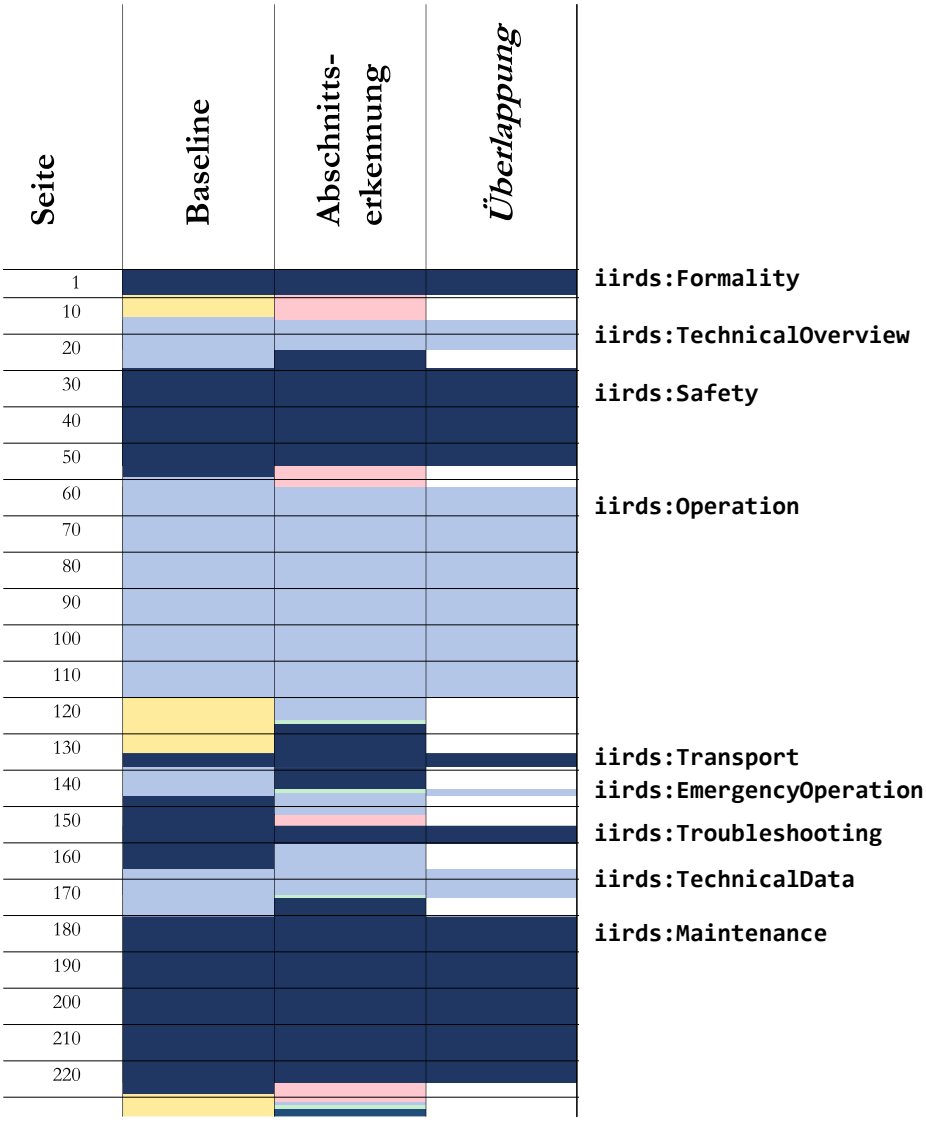

Abbildung 21: Visualisierung der Segmentierungsergebnisse.
Dargestellt ist ein Vergleich zwischen der manuellen Baseline vs. der vorgestellten automatisierten Segmentierung vs. der Überlappung von beiden.

Gelbe Abschnitte waren in den Trainingsdaten nicht enthalten, gestrichelte rote Bereiche in den Segmentierungsergebnissen sind Bereiche mit hoher Unsicherheit (niedrige mittlere Konfidenz), die keine Klassifizierung erhalten haben. Grüne Seiten enthalten Beginn bzw. Ende von zwei Segmenten. Die Überlappung (rechte Spalte) zeigt alle Seitenbereiche, die sich in einem korrekt identifizierten Segment im Vergleich zur Baseline befinden.

Abbildung basierend auf einer Grafik aus Oevermann (2018b).

4.4.6 Implementierung

Im Rahmen dieser Arbeit wurde die vorgestellte Methode als clientseitige browserbasierte Anwendung in JavaScript implementiert. Der Prototyp ist online verfügbar.[73] Der Quellcode kann über ein öffentliches Repository eingesehen und heruntergeladen werden.[74] Aufgrund von vertraulichen Inhalten können die verwendeten Beispieldateien nicht bereitgestellt werden.

Die Implementierung basiert auf früheren Arbeiten (BADER & OEVERMANN, 2017). Die Software wurde um ein modifiziertes Chunking, einen verbesserten Algorithmus zur Abschnittserkennung und eine iiRDS-konforme Metadatenausgabe erweitert. Der Prototyp kann XML- und JSON-Daten für das Training des Modells und PDF-Dateien zur Segmentierung verarbeiten; die Ergebnisse werden als RDF/XML nach dem iiRDS RDF Schema exportiert. Wahlweise können Daten auch dem Web-Annotation-Standard entsprechend oder in einem proprietären JSON-Format ausgegeben werden.

Neben dem browserbasierten Prototypen existiert ebenfalls eine Kommandozeilen-Anwendung, die auf *Node.js*[75] basiert (OEVERMANN, 2016b). In dieser Implementierung werden auf Basis von Heuristiken ebenfalls Kopf- und Fußzeilen aus dem extrahierten Text entfernt, was sich jedoch in manchen Fällen als unzuverlässig erweisen kann.

4.4.7 Anwendung

Der domänenspezifische Anwendungsfall für die vorgestellte Methode ist die Generierung von Metadaten und Segmenten für Bestandsdokumentation (in Form von PDF-Dateien), um die Information-Retrieval-Eigenschaften der Inhalte zu verbessern. Mit der zunehmenden Popularität von Content-Delivery-Portalen wird ein strukturierter und metadatenbasierter Informationszugriff immer wichtiger (ZIEGLER & BEIER, 2015).

Während XML-basierte Module in solcher Weise einfach bereitgestellt werden können, ist Bestandsdokumentation oft von diesen Zugriffsmethoden ausgeschlossen (OEVERMANN, 2016b). Durch den Einsatz von iiRDS können die erzeugten Metadaten von jeder Software ausgewertet werden, die den Standard umsetzt.[76]

---

[73] http://segments.fastclass.de/ [01.02.19]

[74] https://github.com/j-oe/segments [01.02.19]

[75] https://nodejs.org/de/ [07.02.19]

[76] Die iiRDS-Selektorlogik basiert auf dem W3C WebAnnotation Standard (W3C, 2017a). Selektoren bei iiRDS wurden mit der Version 0.9 (RfC) eingeführt.

Die erkannten Segmente können z. B. bei der facettierten Suche verwendet werden, um das F_1-Maß[77] im Information Retrieval zu verbessern (BADER & OEVERMANN, 2017). So werden z. B. Filtermethoden ermöglicht, die eine Volltextsuche auf Seitenabschnitte mit verschiedenen Klassifikationen (z. B. Informationsart oder eine bestimmte Baugruppe) beschränken.

4.4.8 Ausblick

Die vorgestellte Methode kann auch auf andere Dokument- oder Textsorten (außerhalb der Technischen Dokumentation) und alternative unstrukturierte oder schwach strukturierte Dateiformate (z. B. Microsoft-Word-Dateien) angewendet werden. Weitere Versuche mit alterativen Segmentierungsmethoden müssen durchgeführt werden, um den vorgestellten Ansatz abschließend zu evaluieren. So wäre z. B. ein Vergleich oder eine Kombination mit dem in Brants u. a. (2002) beschriebenen Ansatz eine vielversprechende Ergänzung.

4.4.9 Fazit

In dieser Arbeit wurde ein neuartiger Ansatz zur PDF-Segmentierung vorgestellt, der auf semantischen Texteigenschaften basiert und somit unabhängig von Formatierungen oder anderen visuellen Merkmalen von Dokumenten ist. Aufgrund der Charakteristiken der Technischen Dokumentation kann Wissen über die verfügbaren strukturierten Daten genutzt werden, um es auf unstrukturierte Dokumente anzuwenden und daraus Metadaten abzuleiten, die in Information-Retrieval-Anwendungen ausgewertet werden können. Wie eine erste Auswertung zeigt, können semantisch relevante Segmente zuverlässig und mit hoher Genauigkeit in Echtdaten rekonstruiert werden. Gleichzeitig wurden die Auswirkungen von Ausreißern im Klassifikationsprozess ausreichend minimiert.

Durch die Ausgabe der Ergebnisse in einem standardisierten RDF-basierten Format kann der vorgestellte Prozess die erzeugten Metadaten für jede iiRDS-konforme Software bereitstellen, um den Informationszugriff auf Bestandsdokumentationen zu verbessern. Zur einfachen Validierung der Methode und als Grundlage für zukünftige Forschung wird der Quellcode der Implementierung neben einem gehosteten Prototypen zur Verfügung gestellt.

[77] *F-measure*, Kombination aus Precision und Recall, definiert als $F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$.

4.4.10 Weitere Ergebnisse

Aus früheren Arbeiten existieren weitere Analyseergebnisse aus PDF-Dokumenten, die auf Basis der PI-Klassifikation annotiert wurden. Diese beinhalten noch keine Abschnittserkennung, jedoch erste Beurteilungen bezüglich der Stärken und Schwächen der Methode.

Erste Versuche zeigen deutlich den Zusammenhang zwischen Konfidenzverlauf und Klassenwechsel (vgl. Abbildung 22). Ebenfalls interessant zu beobachten ist das Verhalten des Klassifikators in den gekennzeichneten Bereichen (siehe Abbildung 23 und Abbildung 24). Hierbei handelt es sich um Verzeichnisse oder tabellarische Inhalte, die Wörter aus verschiedenen Klassen enthalten und somit nicht einem charakteristischen Muster entsprechen. Der Klassifikator „springt" deshalb in seinen Vorhersagen zwischen mehreren Klassifikationen.

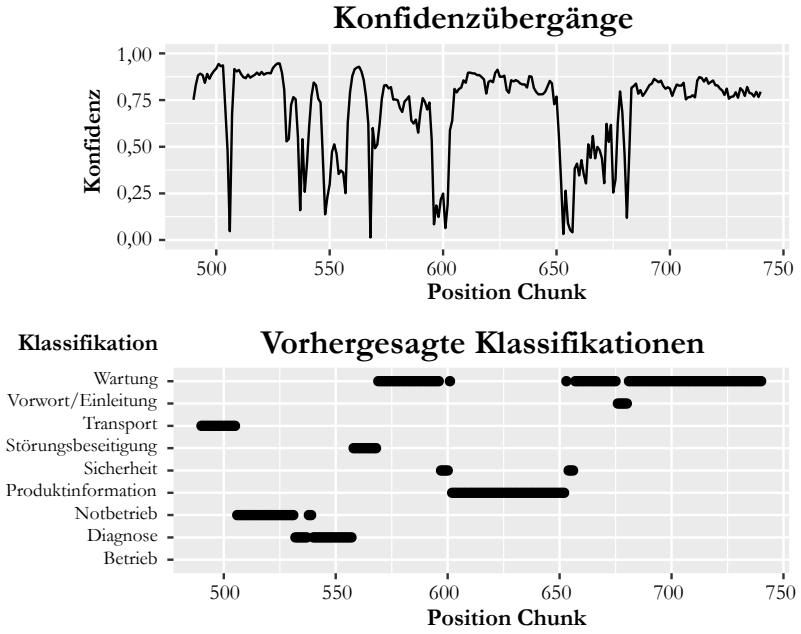

Abbildung 22: Beispiel für Segment- vs. Konfidenzverhalten.
Vergleich von Konfidenzwert zum vorhergesagten PI-Klassifikationswert.
Dargestellt ist ein Ausschnitt aus einem Dokument (Chunks $475 - 750$*).*

Abbildung basierend auf einer Grafik aus Oevermann (2016b).

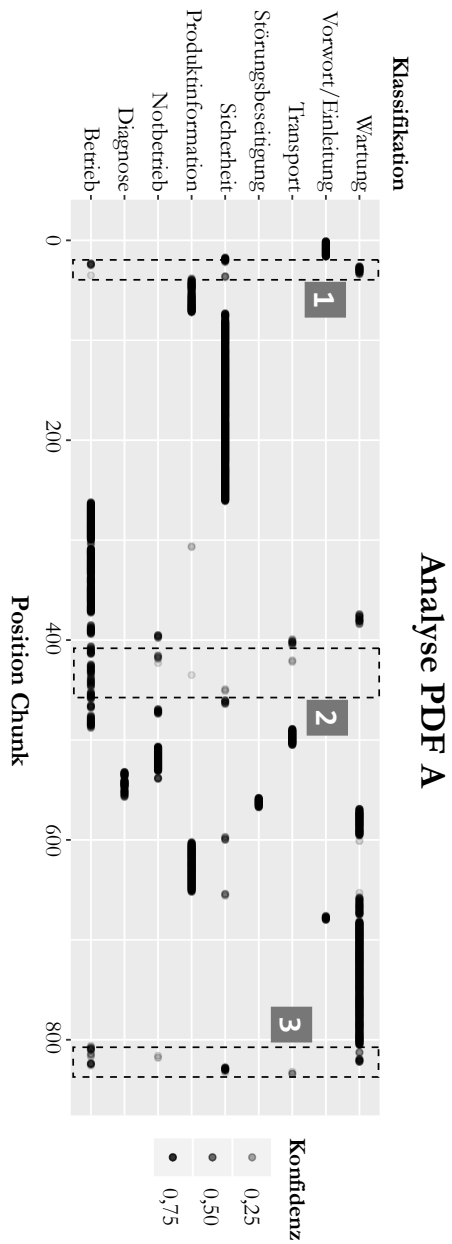

Abbildung 23: Segment-Ergebnisse eine Test-PDFs (Set A). Gekennzeichnet sind die Seitenbereiche von (1) Inhaltsverzeichnis, (2) Wartungstabellen und (3) Index.

Abbildung basierend auf einer Grafik aus Oevermann (2016b).

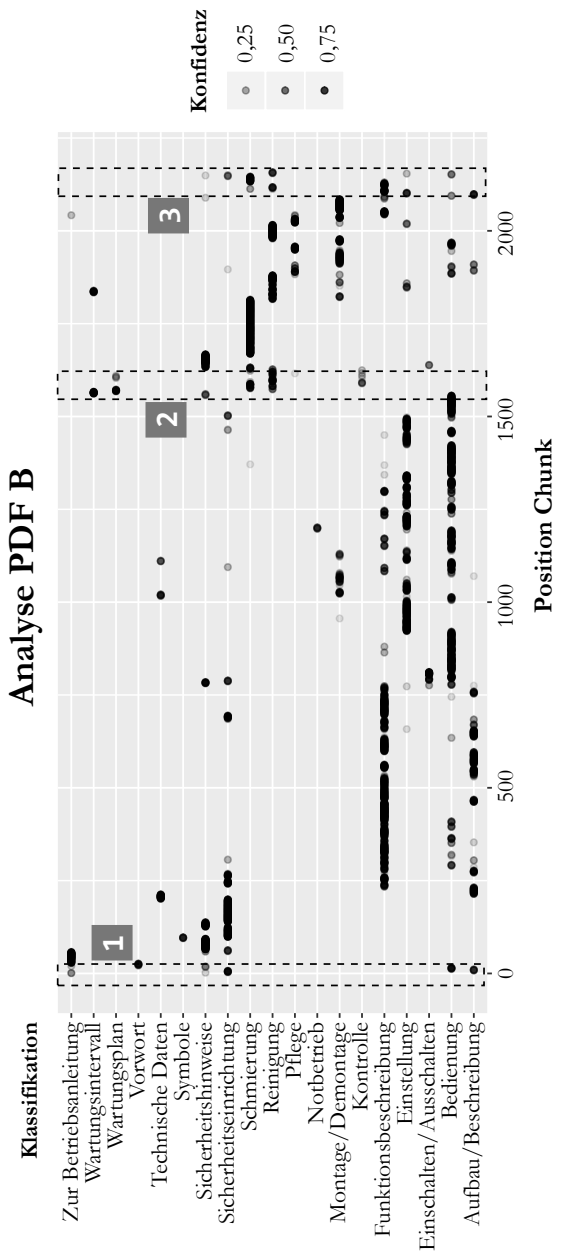

Abbildung 24: Segment-Ergebnisse eine Test-PDFs (Set B).
Gekennzeichnet sind die Seitenbereiche von (1) Inhaltsverzeichnis, (2) Wartungstabellen und (3) Index.

Abbildung basierend auf einer Grafik aus Oevermann (2016b).

4.5 Informationsintegration

Der folgende Inhalt wurde zu großen Teilen aus dem Konferenzbeitrag „Semantic Annotation of Heterogeneous Data Sources : Towards an Integrated Information Framework for Service Technicians“ übernommen (BADER & OEVERMANN, 2017). An geeigneten Stellen wurden im Sinne der Gesamtarbeit Querverweise, Anmerkungen, Abbildungen oder Ergänzungen eingefügt und die Formatierung angepasst.

4.5.1 Zusammenfassung

Servicetechniker im industriellen Bereich benötigen zur Ausübung ihrer Aufgaben umfangreiche technische Kenntnisse und Erfahrungen. Ein Teil dieses benötigten Wissens wird in dokumentenbasierten Handbüchern oder in Einsatzberichten aus früheren Aufträgen bereitgestellt. Aufgrund der großen Datenmengen verbringen Servicetechniker jedoch einen erheblichen Teil ihrer Arbeitszeit mit der Suche nach den richtigen Informationen. Eine weitere Herausforderung stellt die Tatsache dar, dass wertvolle Erkenntnisse aus Serviceberichten aufgrund unzureichender Textqualität und inhaltlicher Mehrdeutigkeit noch nicht in vollem Umfang bei der Informationsrecherche berücksichtigt werden können.

In dieser Arbeit wird ein Framework vorgestellt, das diese heterogenen Datenquellen mit semantischen Annotationen anreichert, um sie dann als Informationseinheiten durch Linked-Data-Technologien verfügbar zu machen. Maschinelles Lernen wird dazu verwendet, um Informationen aus Servicehandbüchern zu modularisieren und zu klassifizieren; mit einer ontologiebasierten Autovervollständigung werden Serviceberichte mit vordefinierten semantischen Konzepten annotiert. Durch die Kombination beider Ansätze kann eine einheitliche und strukturierte Schnittstelle für manuelle und automatisierte Informationsabfragen bereitgestellt werden.

Der vorgestellte Ansatz wird verifiziert, indem Precision und Recall für typische Informationsanfragen von Servicetechnikern gemessen werden. Dabei kann gezeigt werden, dass das Framework den semantischen Informationszugriff in Service- und Wartungsprozessen signifikant verbessern kann.

4.5.2 Einleitung

In modernen industriellen Fertigungsprozessen ist der effiziente Einsatz von Maschinen und Anlagen ein entscheidender Wettbewerbsvorteil, da Unternehmen mit einer höheren Produktionsauslastung mehr Produkte bei gleichen Kosten produzieren können und sich somit im Markt günstig platzieren können. Eine schnelle und reibungslose Instandhaltung ist aus diesen Gründen ein wichtiger Aspekt für jedes produzierende Unternehmen. Allerdings ist der etablierte Wartungsprozess im Vergleich zu seinen hohen Auswirkungen

immer noch sehr stark von manueller Arbeit abhängig. Da qualifizierte Techniker schwer verfügbar und gut bezahlt sind, ist die Unterstützung der Arbeitskräfte im Feld eine der wichtigsten Prioritäten für jeden Service-Dienstleister. In diesem Zusammenhang ist auch die effiziente Bereitstellung von problemspezifischen und relevanten Informationen eine absolute Notwendigkeit. Die bestehenden Informationsquellen für Servicetechniker lassen sich in zwei Hauptkategorien unterteilen (siehe Abbildung 25). Die erste Kategorie enthält von Fachleuten erstellte Handbücher, Verarbeitungsrichtlinien und Schulungsunterlagen. Diese Kategorie wird als „kontrollierte Dokumente" bezeichnet, da ausgebildete Technische Redakteure diese erstellen und veröffentlichen. Kontrollierte Dokumente liefern rechtsverbindliche Anleitungen und technische Informationen für Anlagen, Baugruppen und Prozesse. Sie dienen als Grundlage für die Ausbildung von Technikern und als Hauptquelle für produktspezifisches Wissen.

Kontrollierte Dokumente können je nach Publikationsform weiter in „strukturierte" und „unstrukturierte" Inhalte unterteilt werden. In der Technischen Dokumentation werden strukturierte Inhalte oft in semantisch strukturierten Informationsmodellen erfasst, welche nach den Methoden des Content Management ein Konzept aus in sich geschlossenen Modulen definieren. Typische unstrukturierte, aber kontrollierte Inhalte sind dokumentenbasierte PDF-Dateien. Aufgrund der großen Menge an kontrollierten Dokumenten ist das gezielte Finden der relevanten Informationsausschnitte ein verbreitetes Problem für Servicetechniker.

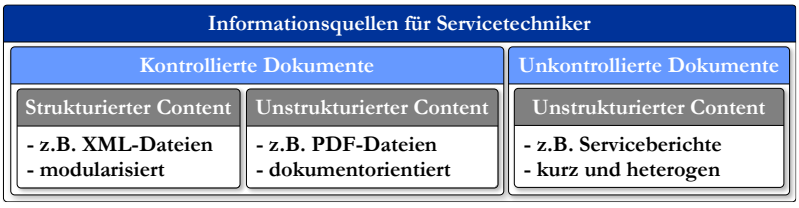

Abbildung 25: Kategorisierte Informationsquellen für Servicetechniker.

Weitere wertvolle Informationen sind in maschinell generierten Logdateien oder manuell erstellten Einsatzberichten enthalten. Diese Informationen werden als „unkontrollierte Dokumente" bezeichnet, da sie sich im Format unterscheiden können, Inhalt und Struktur nicht standardisiert sind und keine rechtlich bindenden Freigaben erfolgen. Dokumente dieser Kategorie liefern spezifische Informationen über den Zustand und die Instandhaltungshistorie einzelner Maschinen und Baugruppen, sowie über die durchgeführten Aktionen von Maschinenbedienern und Servicetechnikern.

Das größte Hindernis für die effiziente Wiederverwendung unkontrollierter Dokumente als Informationseinheiten ist die geringe grammatikalische und syntaktische Qualität der Texte sowie die Verwendung von nicht standardisierter Terminologie und Akronymen. YAMAUCHI, WHALEN & BOBROW (2003) kategorisieren diese, oft aus mehreren Quellen zusammen gesammelten, Informationen als „gleaning“ (auf dt.: „zusammentragen“ oder „sammeln“). Unkontrollierte und unstrukturierte Serviceberichte sind trotz ihrer Form jedoch oft entscheidend, da sie zusätzliche Informationen oder Hinweise enthalten, die in der Regel aktueller und wichtiger sind als die Angaben aus kontrollierten Dokumenten. Das spiegelt sich auch deutlich in der praktischen Anwendung wider: „Technicians prefer gleaning to instruction following“ (YAMAUCHI u. a., 2003). In dieser Arbeit werden beide Kategorien berücksichtigt, wobei jeweils andere Vorgehensweisen mit entsprechenden Vorteilen zum Einsatz kommen, mit dem Ziel, die Ergebnisse zu einem umfassenden Gesamtbild auf das verfügbare Wissen zu kombinieren (vgl. Abbildung 26). Für kontrollierte Dokumente werden spezialisierte Methoden zur Klassifizierung und Modularisierung technischer Inhalte verwendet. Dies ermöglicht eine Aufteilung in einzelne, zusammenhängende Textbausteine und damit eine granulare aber ausreichende Sicht auf die Informationen.

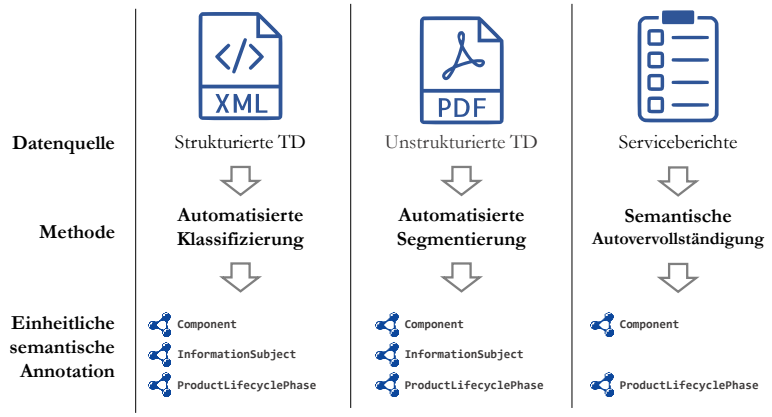

Abbildung 26: Vorgehensweise für unterschiedliche Informationsquellen.

Die größten Herausforderungen bei einer effizienten Wiederverwendung unkontrollierter Dokumente sind die geringe grammatikalische und syntaktische Qualität der Texte sowie die Verwendung von nicht standardisierten Termen und Akronymen darin. Diese Probleme werden für unkontrollierte Dokumente gelöst, indem ein ontologiebasiertes Autovervollständigungswerkzeug zum Einsatz kommt. Dieses Verfahren unterstützt den Techniker schon während des Schreibvorgangs und nutzt gleichzeitig das direkte Feedback zu den ausgewählten Einträgen, um den Bericht zu annotieren. Dadurch wird der

Berichtsprozess beschleunigt und der Anteil der korrekt erkannten semantischen Konzepte erhöht. Die höhere Qualität der Annotationen führt daher zu einer besseren und einfacheren Wiederverwendung der Berichte als Informationsquelle und zu besseren Ergebnissen im Information Retrieval.

In Rahmen dieser Arbeit wurde ein Prototyp für die semantische Annotation von Technischer Dokumentation und von kurzen unstrukturierten Serviceberichten entwickelt. Um beide Ansätze nahtlos miteinander zu kombinieren und eine leichte Erweiterbarkeit des Modells zu gewährleisten, kommen semantische Technologien zum Einsatz. Die Verwendung des vorgestellten Systems durch andere wird durch den Einsatz moderner Ontologien aus dem Bereich der Technischen Dokumentation gefördert. Das System verwendet *Open Knowledge Graphs* (Konzeptgraphen zur Repräsentation einer Wissensdomäne), um die Qualität kurzer Serviceberichte zu steigern und Methoden des Maschinellen Lernens, um Technische Dokumentation mit semantischen Konzepten anzureichern und dadurch Servicetechniker bei der Informationsrecherche zu unterstützen. Die Neuheit des vorgestellten Ansatzes liegt in der Kombination von jeweils geeigneten Methoden für die automatisierte oder unterstütze Annotation von technischen Inhalten und die Integration von universellen Linked-Data-Standards in ein einheitliches Informationsframework (siehe Abbildung 27).

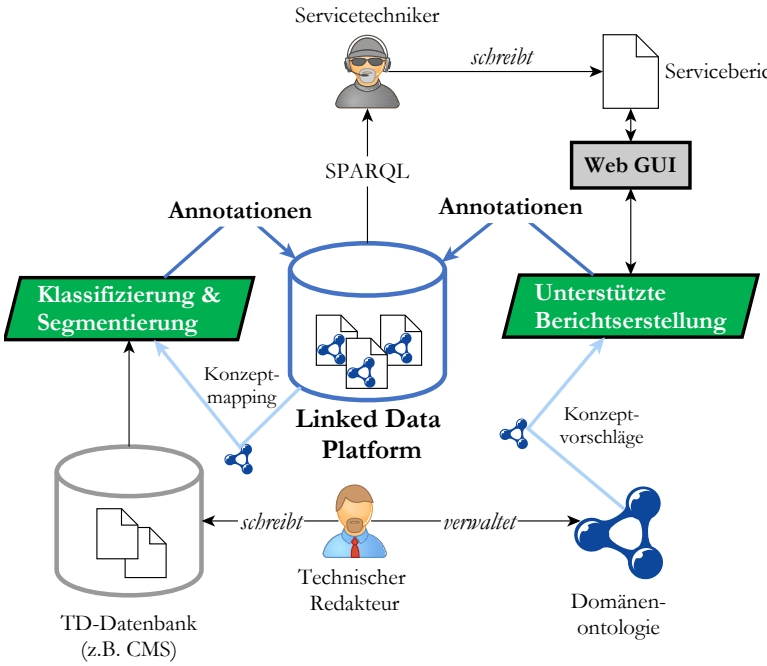

Abbildung 27: Modell des integrierten Informationsframeworks.

Zur Veranschaulichung und zur Validierung des vorgeschlagenen Modells werden Daten aus dem Bereich der industriellen Druckmaschinen verwendet, die mit freundlicher Genehmigung von Unternehmen zur Verfügung gestellt worden sind, um diese Forschung zu unterstützen. Der Ansatz wird am Beispiel eines Servicetechnikers erläutert, der relevante Informationen darüber finden möchte, wie er eine Druckmaschine mit einem verschmutzten Offset-Farbwerk warten kann. Die abschließende Bewertung erfolgt anhand einer Messung von F_1-, Precision- und Recall-Metriken für verschiedene Recherche-aufgaben im Information Retrieval.

4.5.3 Verwandte Arbeiten

Schon Mitte der 90er-Jahre wurde gezeigt, dass ein integrierter Ansatz im Datenmanagement zu einem erheblichen Wettbewerbsvorteil im Bereich des industriellen Service führen kann (BLUMBERG, 1994). In einem Vergleich von mehr als 100 Serviceanbietern wird dargelegt, wie Standardisierung und Assistenzsysteme die Effizienz von Technikern verbessern können. Einige der vorgeschlagenen Verbesserungen, wie z. B. die drahtlose Kommunikation, haben sich bereits weitgehend durchgesetzt. Das Hauptproblem, Servicetechnikern die notwendigen Informationen für die richtige Aufgabe am gewünschten Ort und zur gewünschten Zeit zur Verfügung zu stellen, ist jedoch weiterhin noch nicht gelöst.

YAMAUCHI u. a. (2003) erläutern in ihrer Arbeit, dass informelle Informationsquellen wie z. B. Kurznotizen und Einsatzberichte immer dann entscheidend sind, wenn ein nicht triviales Problem bei der Ausübung der Aufgaben auftritt. Nach den dort gemachten Beobachtungen versuchen Servicetechniker zunächst eine Erklärung zu finden, die auf eigenen Erfahrungen oder Ratschlägen von Kollegen beruht. Wenn dieses Verfahren fehlschlägt, wird die Suche in der kontrollierten Dokumentationsbasis fortgesetzt. Daher sind beide Informationsquellen notwendig, kommen jedoch in unterschiedlichen Phasen des Prozesses zum Einsatz.

DIMOU, VERBORGH, SANDE, MANNENS & VAN DE WALLE (2015) beschreiben Methoden zum RDF-Mapping von heterogenen Datenquellen für den Einsatz im Information Retrieval. SCHWEITZER & AURICH (2010) schlagen einen kontinuierlichen Verbesserungsprozess für die Serviceorganisation vor, bei dem sowohl kontrollierte als auch unkontrollierte Dokumente im Netzwerk der Wartungstechniker miteinander geteilt werden. Dieser Ansatz gewährt insbesondere Kunden, aber auch Lieferanten Zugriff auf eine gemeinsame Wissensdatenbank. Obwohl die Autoren die wirtschaftliche Notwendigkeit skizzieren, wird keine konkrete Lösung vorgeschlagen, um die verschiedenen Systeme effektiv zu vernetzen und ein gemeinsames Datenverständnis über das gesamte Netzwerk zu gewährleisten.

Die Modularisierung und Klassifizierung von Informationseinheiten mit semantischen Annotationen bietet große Vorteile für Wissensmanagementsysteme. Dazu gehören verbesserte Eigenschaften in der strukturierten Informationsabfrage durch semantische Technologien und eine bessere Interoperabilität zwischen verschiedenen Systemen. UREN u. a. (2006) geben einen Überblick über manuelle und automatisierte Werkzeuge für semantische Annotationen von Dokumenten. Es wird gezeigt, dass ein manueller Annotationsprozess zu arbeitsintensiv ist und daher automatisiert werden muss. Drei Hauptstrategien werden von den Autoren vorgeschlagen: regel- oder musterbasierte Systeme, überwachtes und unüberwachtes Maschinelles Lernen. Eine der Schlussfolgerungen besagt, dass die erforderlichen Fähigkeiten zur Konfiguration der automatisierten Annotationssysteme und der Aufwand zur Erstellung von Trainingsdaten nicht immer durch eine geeignete Annotationsqualität gerechtfertigt werden. Weitere Forschung zu automatisierter semantischer Annotation findet sich darüber hinaus bei DILL u. a. (2003).

4.5.4 Unterstützte Berichtserstellung

Semantische Autovervollständigung (engl.: „Semantic Autocompletion"), wie von HYVÖNEN & MÄKELÄ (2006) vorgeschlagen, kombiniert die Vorteile von kontrollierten Sprachen (wie z. B. definierte Begriffe aus einem vordefinierten Vokabular) mit einer Unterstützung des Autors während des Schreibprozesses. Es werden Zeichenketten-basierte Ähnlichkeiten verwendet, um verwandte Konzepte in einer Ontologie zu identifizieren. Dies ermöglicht es, Terme basierend auf ihrer Bedeutung und nicht nur über ihre *Levenshtein-Distanz*[78] vorzuschlagen. Software-Werkzeuge wie „Magpie" (DZBOR, MOTTA & DOMINGUE, 2004) und „DBpedia Spotlight" (DAIBER, JAKOB, HOKAMP & MENDES, 2013) annotieren Texte mit ontologischen Konzepten. Im vorgestellten Anwendungsfall können die identifizierten Entitäten verwendet werden, um Dokumente oder Absätze zu erstellen, die für semantische Suchmaschinen abfragbar sind. Dennoch erfordern diese Systeme eine korrekt formulierte Eingabe. Wie bereits erwähnt, ist dies nicht geeignet für Serviceberichte, bei denen Zeitdruck und inkorrekte Schreibweise zu fehlerhaften Annotationen führen können.

Kontrollierte und unkontrollierte Informationsquellen für Servicetechniker unterscheiden sich in einer Reihe von Eigenschaften. Während kontrollierte Dokumente wie z. B. Handbücher, Produktdokumentationen oder Prozessleitfäden aus langen und abschnittsweise strukturierten Texten bestehen, enthalten die meisten unkontrollierten Dokumente nur wenige Absätze oder

[78] Oft auch "Editierdistanz", im Englischen "Edit distance". Ein Maß für die Ähnlichkeit zwischen zwei Zeichenketten (CARSTENSEN, 2010, S. 558).

sogar nur einzelne Sätze, wobei das behandelte Thema nicht explizit benannt wird. Die Ersteller von unkontrollierten Dokumenten – in dieser Arbeit werden hauptsächlich Serviceberichte betrachtet – sind weder ausgebildete Autoren noch dazu bereit, viel Zeit in das Schreiben zu investieren. Um die notwendige textliche Qualität für die Wiederverwendung der Berichtstexte in zukünftigen Einsätzen zu erreichen, müssen Domänenexperten mit Erfahrung in Terminologie und Redaktionsprozessen diese interpretieren und in nutzbare Textmodule umwandeln. Obwohl dies in einigen Fällen zur Dokumentation von besonders wichtigem Wissen auch durchgeführt wird, kann ein manueller Integrationsprozess die hohe Menge und Heterogenität der eingehenden Berichte in einer modernen Serviceorganisation nicht bewältigen. Zwei große Herausforderungen hindern Anbieter von Serviceleistungen daran, Einsatzberichte umfassend und automatisiert in ihre Wissensdatenbanken zu integrieren. Die geringe textliche und orthographische Qualität verhindert eine ausreichend präzise automatisierte Klassifizierung und Segmentierung der Inhalte. Des Weiteren erfordert die fehlende Verwendung vordefinierter semantischer Konzepte eine komplexe Auflösung von Mehrdeutigkeiten. Dies ist eine schwierige Aufgabe im Maschinen- und Anlagenbau, wo ähnliche Benennungen völlig unterschiedliche Bedeutungen haben können und exakte Beschreibungen von enormer Wichtigkeit sind.

Basierend auf Gesprächen mit Domänenexperten und Führungskräften in Serviceorganisationen konnte herausgearbeitet werden, dass die einzig praktikable Strategie darin besteht, den Servicetechniker während des Schreibprozesses aktiv zu unterstützen. Deshalb wird ein unterstützender Ansatz empfohlen, bei dem das System alternative Formulierungen vorschlägt. Der Techniker wählt den Vorschlag aus, während er den Bericht weiterschreibt. Dadurch wird die Genauigkeit der dokumentierten Situation erhöht und gleichzeitig der Schreibaufwand minimiert. Dies ist besonders wichtig, da die Benutzer des Systems auch einen direkten, individuellen Nutzen wahrnehmen müssen. Somit kann beiden genannten Herausforderungen begegnet werden, bevor der Servicebericht (und die annotierten Konzepte) die Wissensdatenbank erreichen.

4.5.4.1 Semantische Autovervollständigung

Die Benutzeroberfläche ist in zwei Teile gegliedert. Ein Bereich definiert Metainformationen, die notwendig sind, um einen Bericht mit einem bestimmten Serviceauftrag zu verknüpfen. Informationen über durchgeführte Aktionen, die beobachtete Situationen und informelle „Best Practices“ werden in einen Freitextbereich eingefügt. Diese Aussagen sind für Serviceorganisationen von besonderem Interesse, da sie beschreiben, welche Probleme tatsächlich aufgetreten sind, welche Strategien angewendet wurden und wie das Problem

letztendlich behoben werden konnte. Andere Servicetechniker in ähnlichen Situationen können von diesen Vorschlägen direkt profitieren. Die Textausschnitte werden als Autovervollständigungsabfrage und als unscharfe Abfrage[79] auf einen vordefinierten Lucene[80]-Index der Domänenontologie abgesetzt, um passende Entitäten zu finden. Der Index wurde mit erweiterten Werkzeugen erstellt, die auf der Arbeit von ELL & HARTH (2014) basieren. Anschließend werden die Suchergebnisse in RDF umgewandelt und an die Benutzeroberfläche weitergeleitet. Der Techniker kann nun einen Vorschlag auswählen, der seinen Vorstellungen entspricht, oder die Vorschläge ignorieren. Eine Auswahl bedeutet entsprechend, dass die vorgeschlagene Einheit für den durchgeführten Vorgang relevant ist. Daher werden die Entität selbst, ihre Klasse und Baugruppe als übereinstimmende Annotationen gespeichert (vgl. Abbildung 28).

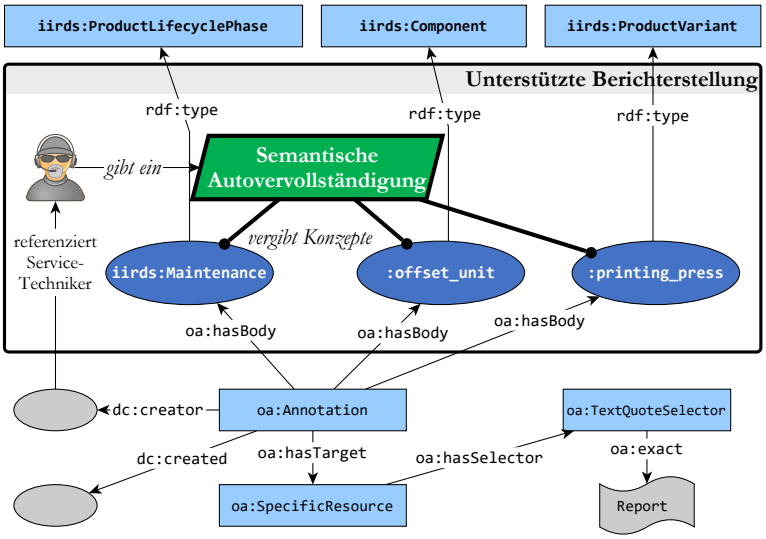

Abbildung 28: Annotationsmodell der semantischen Autovervollständigung.

Die Entitäten für den Vorschlagsprozess werden in Form einer Domänenontologie modelliert, die ähnlich zu der von KIRYAKOV, POPOV, TERZIEV, MANOV & OGNYANOFF (2004) beschriebenen ist. Relevante Konzepte für Servicetechniker wurden im Rahmen des *STEP*-Projekts gesammelt und sind

[79] Auch Fuzzy-Suche oder engl. "fuzzy query". Bei Lucence über die *Damerau-Levenshtein-Distanz*, vgl. https://lucene.apache.org/core/5_0_0/core/org/apache/lucene/search/FuzzyQuery.html [01.02.19].

[80] vgl. https://lucene.apache.org/core/5_0_0/ [01.02.19].

öffentlich zugänglich.[81] Die extrahierten Ontologie-Informationen in Form von *N-Quads*[82] sind die Basis für den Lucene-Index; jede Entität wird dabei durch ein neues Lucene-Dokument repräsentiert. Gespeichert werden alle verfügbaren Literale (insbesondere **`rdfs:label`**), die zugehörigen Klassen und die URI der Entität. Somit liefert eine Übereinstimmung mit einem beliebigen textuellen Inhalt der Entität alle verfügbaren Informationen des Konzeptgraphen zurück. Dadurch werden weitere Anfragen vermieden und die Reaktionszeit verkürzt.

Es werden alle Entitäten abgefragt, die in einer der oben genannten Eigenschaften dem eingefügten Begriffsfragment ähnlich sind. Zunächst sucht eine Autovervollständigungsabfrage nach Mustern wie „**`. *printing. *`**“. Die betrachtete Domäne wird von Benennungen dominiert, die im Englischen aus mehreren Wörtern bestehen (z. B. „printing press“ oder „offset unit“)[83], wobei eine Teilmenge von Wörtern (z. B. „press“) die jeweilige Baugruppe nicht ausreichend beschreibt. Bei diesem Abfragetyp wird darauf geachtet, spezifischere Begriffe vorzuschlagen. Die unscharfe Abfrage im zweiten Schritt gibt ähnlich geschriebene Entitäten zurück und beugt im Wesentlichen Übertragungs- und Rechtschreibfehlern vor. Die Ergebnismengen beider Abfragen werden kombiniert und über einen *Matching Score* sortiert und mit Hilfe eines Popup-Menüs in der Benutzeroberfläche der Texteingabe visualisiert.

Zukünftig sollen auch die erkannten Konzepte dazu genutzt werden, ein Muster der vorkommenden Klassen zu erstellen. Ähnlich wie bei den so genannten *Bridge Patterns*[84] (ELL & HARTH, 2014) ermöglichen es diese Graphen dem Nutzer auch komplexere Formulierungen vorzuschlagen. Darüber hinaus enthält ein ausgewähltes Bridge Pattern nicht nur Informationen über die Konzepte selbst, sondern auch über die Beziehungen dazwischen. Diese Informationen erlauben ein tieferes Verständnis der beschriebenen Situation und ermöglichen eine bessere Verknüpfung von Informationsbedürfnissen zu den verfügbaren Serviceberichten. Darüber hinaus zeigen die Arbeiten von PEREZ-BELTRACHINI, SAYED & GARDENT (2016) verschiedene Methoden zur Transformation bestehender strukturierter Daten für erweiterte Vorschläge von Textformulierungen.

---

[81] vgl. http://people.aifb.kit.edu/mu2771/step/ [01.02.19].

[82] N-Quads erweitern (N-)Tripel mit einem Kontext, z.B. einem Graphen (W3C, 2014b).

[83] Im Deutschen würden diese Mehrwortbennungen als Komposita abgebildet werden: z. B. Druckmaschine oder Druckerpresse, Offset-Einheit etc.

[84] Bridge Patterns werden im Deutschen auch simpel als "Brücken" im Sinne von Strukturmustern bezeichnet.

4.5.4.2 Methodologie

Eine Gruppe von neun Probanden wurde mit zehn Situationen konfrontiert, die auf realen Wartungsaufgaben basieren und die in Englisch oder Deutsch beschrieben sind. Jeder Proband hatte 15 Sekunden Zeit, um die zugeteilte Servicemeldung zu lesen und weitere 20 Sekunden, um die durchzuführende Aktion zu verstehen. Anschließend wurden die Probanden gebeten, alle Fakten aus der Meldung, an die sie sich erinnern können, in die Web-Oberfläche des semantischen Autovervollständigungsmoduls einzufügen. Um die Eignung der vorgestellten Lösung vergleichen zu können, wurde die Autovervollständigung nur für die Hälfte aller Aufgaben aktiviert. Um verzerrte Ergebnisse bei einfacheren oder komplexeren Beschreibungen der beschriebenen Situationen zu vermeiden, variierte auch die Bereitstellung von Terminologievorschlägen für jede Aufgabe und jeden Probanden. Um die Auswirkungen der Autovervollständigung zu bewerten, wurde die korrekten Erwähnungen von passenden Entitäten in den Servicemeldungen analysiert. Die mit diesem Versuchsaufbau erzielten Ergebnisse werden in Abschnitt 4.5.9 erläutert.

4.5.5 Annotation von Technischer Dokumentation

Die Technische Dokumentation einer Maschine ist die für Servicetechniker zuverlässigste und einzig rechtsverbindliche Informationsquelle.[85] Sie kann aus mehreren Dokumenten, Schaltplänen oder auch Webseiten bestehen, wobei gedruckte Handbücher die älteste und verbreitetste Form sind (DREWER & ZIEGLER, 2011). Im Zusammenhang mit der Integration heterogener Datenquellen wird die Gesamtheit der Technischen Dokumentation auch unter der Bezeichnung „kontrollierte Dokumente“ zusammengefasst.

Vor allem im industriellen Bereich wird die Technische Dokumentation in der Regel von ausgebildeten Technischen Redakteuren geschrieben (STRAUB, 2016), die Inhalte modularisiert und semantisch strukturiert erstellen. Diese in sich geschlossenen Informationseinheiten, die Module oder Topics genannt werden, steigern die referenzierte Wiederverwendung über verschiedene Dokumente hinweg und senken die Übersetzungskosten für globalisierte Unternehmen (SOTO u. a., 2015). Sie können als druckbare Dokumente veröffentlicht oder on-demand als einzelne Module über Online-Portale oder Smartphone-Apps bereitgestellt werden (Content Delivery). Obwohl mittlerweile in einigen Branchen eine rein digitale Bereitstellung von Technischer Dokumentation gesetzlich erlaubt ist (z. B. im Softwarebereich), sind druckfähige PDF-Dateien das am weitesten verbreitete Format (STRAUB, 2016).

[85] In den entsprechenden Richtlinien wird in der Gesamtheit (inkl. Zertifikate, Risikobeurteilung, etc.) auch oft von „Technischen Unterlagen“ gesprochen (2006/42/EG, 2006).

Klassifizierende Metadaten oder semantischen Annotationen können zur Integration von Modulen in dynamische Szenarien verwendet werden, bei denen Informationen *on-the-fly* aggregiert, automatisch gefiltert oder in der facettierten Suche verwendet werden (ZIEGLER & BEIER, 2014). Standardisierte Klassifikationsmethoden, wie z. B. die PI-Klassifikation, weisen taxonomische Klassen zu, mit denen Inhalte über semantische Eigenschaften identifiziert werden können (DREWER & ZIEGLER, 2011). Im Gegensatz dazu wird Bestandsdokumentation für ältere Maschinen oder auch neue Handbücher kleinerer Hersteller oft nur dokumentenbasiert, z. B. in Archivformaten wie PDF, gespeichert und nicht semantisch annotiert, was einen gezielten Informationszugriff verhindert. Gerade im Maschinen- und Anlagenbau kann jedoch die maximale Informationsmenge für ein Produkt allein von mehreren hundert bis tausend Seiten gedruckter Technischer Dokumentation reichen.[86] Dies stellt für Servicetechniker ein großes Problem dar, da das Auffinden relevanter Informationen mit mehr Daten immer schwieriger (und zeitaufwändiger) wird. Die verbreitete dokumentenbasierte Bereitstellung erschwert die effiziente Suche und Filterung der verfügbaren Inhalte noch weiter.

4.5.5.1 Automatisierte Klassifizierung

Zum Annotieren von strukturierten Inhalten mit einer automatisierten Multiklassen-Klassifizierung von Modulen der Technischen Kommunikation wird die Methode verwendet, die erstmals in OEVERMANN & ZIEGLER (2016) eingeführt wurden (zur Weiterentwicklung dieser Methode vgl. Abschnitt 4.3). Dort wurde festgestellt, dass Unterschiede in Form und Inhalt dieser Textsorte domänenspezifische Anpassungen an etablierte Klassifizierungstechniken erfordern. Das Verfahren basiert auf dem Vektorraummodell und baut prototypische Klassenvektoren auf, die dann über eine Messung der Kosinusähnlichkeit und einem Nächste-Nachbarn-Klassifikator klassifiziert werden.

4.5.5.1.1 Methodologie

Für den Versuchsaufbau wurden kontrollierte und strukturierte Inhalte aus 3686 manuell klassifizierten Modulen in deutscher Sprache mit einer durchschnittlichen Größe von 87 Wörtern pro Modul als Trainingsdaten vorbereitet. In einem Vorverarbeitungsschritt wurde reiner Text aus den Modulen extrahiert und unnötiger Leerraum, Ziffern, Sonderzeichen und Satzzeichen entfernt. Merkmale wurden als Wortgruppen ($n = 2$) extrahiert und dann mit der $\mathrm{TF} - \mathrm{ICF} - \mathrm{CF}$-Methode gewichtet (OEVERMANN & ZIEGLER, 2016).

[86] Die hier verwendeten Testdaten enthalten ca. 500 XML-basierte Module und etwa 700 Seiten an PDF-Dokumentation.

Ein Modul, das klassifiziert werden soll, wird als Vektor $\vec{m} = (w_1, w_2, \dots w_n)$ dargestellt, wobei n die Anzahl der als Merkmale gewählten Token[87] ist. Der Wert w_i repräsentiert die semantische Gewichtung von Token i. Durch überwachtes Lernen wurde eine $n \times c$ Token-Klasse-Matrix $M = w_{ij}$ für eine Menge verschiedener Klassen C aufgebaut, so dass jede Klasse als prototypischer Vektor dargestellt wird. Klassenvektoren bestehen aus Gewichtungen, die aus der spezifischen Verteilung eines Tokens i in der Klasse j über alle Module in den Trainingsdaten berechnet werden. Der Aufbau basiert aus Performance-Gründen auf einem Vektorraummodell und nicht auf komplexeren Methoden wie neuronalen Netzen. Als Multiklassen-Klassifikator wurde aufgrund der hohen Anzahl von Merkmalen und der heterogenen Größe und Verteilung der Klassen (COLAS u. a., 2007) die Kosinusähnlichkeit (vgl. z. B. MANNING & SCHÜTZE, 1999) gewählt (in Kombination mit einem Nächste-Nachbarn-Klassifikator) statt eines naiven Bayes-Klassifikators oder Support Vector Machines. Für alle Klassifizierungsaufgaben wurde die gleiche Konfiguration von Parametern verwendet.

4.5.5.1.2 Klassifikationssystem

Wie an anderer Stelle in dieser Arbeit beschrieben sind die verwendeten Klassifikationssysteme und -methoden innerhalb der Technischen Dokumentation ein wichtiger Faktor bei der automatisierten Klassifizierung (vgl. Abschnitt 4.3.5.2).

Grundlage für die manuelle Klassifizierung der Testdaten ist ein PI-Klassifizierungsmodell, das intrinsische Klassen für Informationsart eines Moduls beinhaltet (z. B. „Technische Daten“ oder „Wartungstätigkeit“) sowie die Zuordnung des Modules zu einem Teil des Produkts (z. B. „Druckmaschine“ oder „Offset-Einheit“). In Verbindung mit extrinsischen Metadaten (z. B. dem spezifischen Modell des Produkts, für das der Inhalt gültig ist) ist es möglich, relevante Module für spezifische Anwendungsfälle (z. B. „die Wartung der Offset-Druckeinheit für die Druckmaschine PP-3B“) zuverlässig zu filtern. Technische Redakteure weisen diese Klassen bereits beim Erfassen des Inhalts zu, oft unterstützt durch ein Content-Management-System. Im Testaufbau wurden Inhalte mit intrinsischen informationsbezogenen Klassen („welche Art von Informationen?“ → **`iirds:Maintenance`**) und mit produktbezogenen Klassen („welcher Teil des Produkts?“ → **`:offset_unit`**) klassifiziert. Die Zuordnung von konkreten PI-Klassifikationswerten zu semantischen Instanzen der Ontologie ist in Abschnitt 4.5.6 beschrieben, die zugehörige semantische Modellierung ist in Abbildung 29 dargestellt.

[87] In diesem Fall sind *Tokens* Wörter oder Einträge von Tabellenzellen

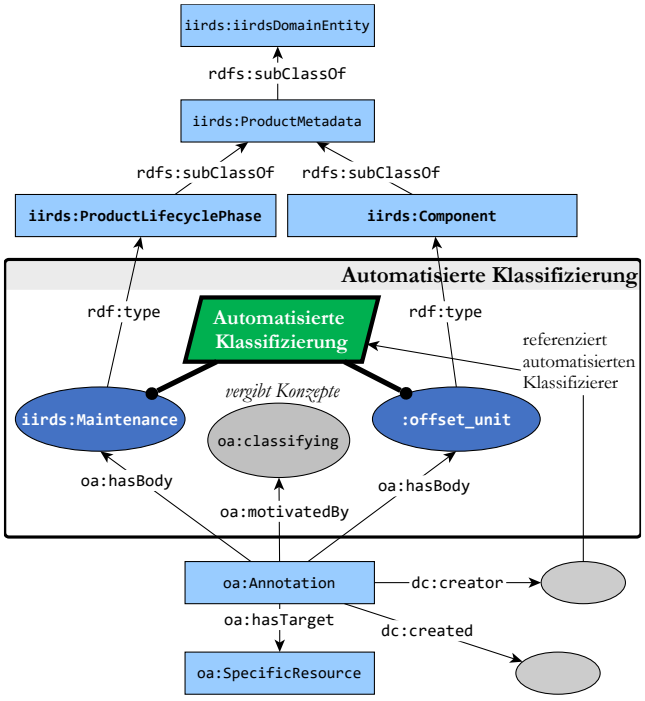

Abbildung 29: Annotationsmodell der automatisierten Klassifizierung.

4.5.5.2 Automatisierte Segmentierung

Inhalte, die nur in einem unstrukturierten und dokumentenbasierten Format vorliegen (z. B. in nicht-getaggten PDF-Dateien), sind in der Regel von einem granularen Informationszugriff durch Filterung oder Facettierung ausgeschlossen, da sie weder modularisiert noch klassifiziert sind. Zur Segmentierung wird eine zuerst in OEVERMANN (2016b) vorgestellte Methode verwendet, welche die semantische Struktur eines Dokuments rekonstruiert, indem Grenzen zwischen Inhalten verschiedener Klassen erkannt werden. Dieser Algorithmus wurde im Rahmen der Arbeit um eine Neuberechnung der relativen Seitenpositionen basierend auf Zeichenlängen, einer automatisierten Abschnittsermittlung und die Ausgabe der Ergebnisse in das standardisierte Web-Annotation-Datenmodell (siehe Abschnitt 4.5.6) erweitert. Da die Segmentierungsmethode den gleichen TD-spezifischen Klassifikator für Module verwendet, kann sie problemlos im bestehenden Testaufbau genutzt werden. Ein weiterer Vorteil besteht darin, dass die Klassifizierung ausschließlich auf extrahiertem Text basiert und somit Altdokumente unabhängig von ihrer optischen Erscheinung oder Formatierung verarbeitet werden können. Ebenso können gescannte Papier-Dokumente mit OCR-Techniken vorverarbeitet werden.

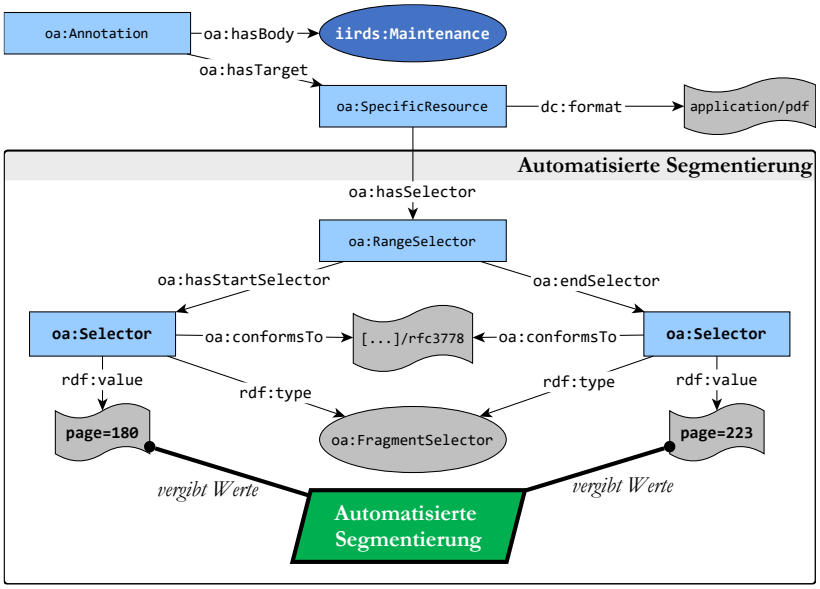

Abbildung 30: Annotationsmodell der automatisierten Segmentierung. Das Beispiel zeigt einen erkannten Abschnitt mit Wartungsinformationen in einem PDF-Dokument.

4.5.5.2.1 Methodologie

Der Text aus den PDF-Dokumenten wurde mit der Open-Source-Library PDF.js[88] extrahiert und zu einer Zeichenkette kombiniert, wobei Silbentrennung und Zeichensetzung auf Basis einfacher Heuristiken entfernt wurden. Die verbleibende Zeichenkette wird dann durch Tokenisierung nach Wortgrenzen (Leerzeichen und Zeilenumbrüche) in einzelne Wörter zerlegt.

Aus der Menge der extrahierten Wörter W werden arbiträre Chunks $C = c_1, c_2, \dots c_n$ gebildet, wobei $c_i \subset W$ gilt (vgl. ausführliche Beschreibung in Abschnitt 4.4.4.4). Die konkrete Größe der Chunks basiert auf der zuvor gesammelten durchschnittlichen Wortanzahl der Module a, im Falle der vorliegenden Testdaten $a = 87$. Um Text-Chunks über den Dokumentinhalt verteilt zu generieren (sog. *Sliding window*), wird eine natürliche Zahl r als Versatz (*Offset*) mit $r \leq a$ definiert. Diese Verschiebung definiert, wie sich mehrere Chunks überschneiden (OEVERMANN, 2016b). Chunks werden mit zusätzlichen Metainformationen über die genaue Zeichenposition im Verhältnis zum gesamten Dokument gespeichert, um die Seitenzahl im PDF und die konkrete Position auf der Seite rekonstruieren zu können.

[88] vgl. https://mozilla.github.io/pdf.js [01.02.19].

Nach der Generierung aller Chunks können diese mit der gleichen Methode wie die Module klassifiziert werden (siehe Abschnitt 4.5.5.1.1 bzw. die Erläuterungen in Abschnitt 4.3.4). Zusätzlich zur vorhergesagten Klasse wird auch die Konfidenz des Klassifikators berechnet (vgl. Abschnitt 4.3.6.5)

4.5.5.2.2 Abschnittserkennung

Nach der Klassifizierung der generierten Chunks und dem Plotten der Ergebnisse entlang der Seitenreihenfolge des Dokuments können mehrere Cluster der gleichen Klassifizierung erkannt werden (vgl. z. B. Abbildung 19). Um diese Teile des Dokuments zu annotieren, werden Seitenbereiche für zusammenhängende Chunks mit der gleichen Klassifizierung definiert. Semantisch in sich geschlossene Abschnitte im Dokument (z. B. das Wartungskapitel) lassen sich in der Regel zuverlässig vorhersagen. Ein Abschnitt wird daher durch eine Start- und Endseitenzahl im PDF-Dokument definiert und kann durch die vorhergesagte Klassifikation und das enthaltene Textäquivalent annotiert werden. Der Algorithmus kann dahingehend weiter verbessert werden, indem Ausreißer nicht berücksichtigt werden, die zwischen Chunks der gleichen Klassifikation auftreten und gleichzeitig eine geringe Konfidenz haben (vgl. Abschnitt 4.4.4.6).[89]

4.5.6 Standardisierte semantische Annotation

Um die generierten Klassifikationen von Serviceberichten, Modulen und Dokumentabschnitten in ein umfassendes Informationsframework zu integrieren, werden mehrere Standards verwendet, die auf den Prinzipien einer „Linked Data Platform“ basieren (W3C, 2015). Ohne ein einheitliches semantisches Modell wäre es nicht möglich, diese heterogenen Datenquellen zu kombinieren. Deshalb wurde darauf geachtet, dass bestehende Standards und Technologien wiederverwendet werden, um die Daten für eine Vielzahl von Anwendungen verfügbar zu machen und Andockpunkte für zukünftige Erweiterungen bereitzustellen. In dieser Arbeit wird die Flexibilität des semantischen Informationsframeworks durch die Integration von Inhalten aus drei verschiedenen Datenquellen demonstriert: (XML-)Module, (PDF-)Dokumente und (textuelle) Serviceberichte.

4.5.6.1 Web Annotation

Im Februar 2017 veröffentlichte das W3C eine *Recommendation* für „Web Annotation“, die aus drei Teilen besteht: Datenmodell (W3C, 2017b), Vokabular (W3C, 2017a) und Protokoll. Die Spezifikation ist das Ergebnis jahrelanger Bemühungen in der Semantic Web Community, eine universelle Methode zur

[89] Die hier beschriebene „einfache“ Abschnittserkennung wurde mittlerweile durch den im Abschnitt 4.4.4.6 vorgestellten Algorithmus abgelöst, der bei realen Daten wesentlich bessere Ergebnisse erzielt, indem die vorgeschlagenen Änderungen berücksichtigt wurden.

Annotation beliebiger Ressourcen oder Teilen davon zu entwickeln (HASLHOFER, SANDERSON, SIMON & VAN DE SOMPEL, 2012). Da der Standard „Klassifizieren“[90] als eine der möglichen Motivationen für das Annotieren definiert, entspricht sie dem Bedarf nach einem flexiblen semantischen Framework zur Integration mehrerer annotierter Datenquellen. Annotationen bestehen aus zwei Hauptteilen: einem *Body* (die eigentliche Annotation) und einem *Target* (die annotierte Ressource, Ziel). Targets können durch *Selektoren* weiter verfeinert werden, die auf einen bestimmten Teil oder Ausschnitt einer Ressource verweisen können.[91] Annotationen können zusätzliche Metadaten (z. B. Informationen über ihre Herkunft) enthalten und sind wiederum selbst Ressourcen.

4.5.6.2 iiRDS

Im März 2017 veröffentlichte die Gesellschaft für Technische Kommunikation (*tekom*) einen „First Public Working Draft“ des „intelligent information – Request and Delivery Standard“ (iiRDS) (TEKOM E.V., 2017a). Neben der Festlegung eines Containerformats für Dokumente führt der Standard auch ein Datenmodell (TEKOM E.V., 2017b) für die Kommentierung von Technischen Dokumentationen ein:

> „iiRDS defines a taxonomy of information types and describes relations between information units as a basic ontology. Thus, iiRDS is the first standard that provides a comprehensive vocabulary for technical documentation.“ (TEKOM E.V., 2017a).

Das Schema definiert Andockpunkte (in Form von Klassen), um die Beziehung zwischen einer Informationseinheit und einem Teil eines Produkts auszudrücken. Die entsprechende Klasse (**`iirds:Component`**) wurde zur bestehenden Produkt-Baugruppen-Ontologie hinzugefügt, um beide zu verbinden und intrinsische Produktklassen (siehe Abschnitt 4.5.5.1.2) auf in der Ontologie definierte Komponenten abzubilden. Da das Metadatenmodell von iiRDS teilweise auf den Prinzipien der PI-Klassifizierung basiert, ist ein 1:1-Mapping zwischen Klassen (siehe Abschnitt 4.5.5.1.2) und semantischen Instanzen (siehe Abschnitt 4.5.6.2) möglich. Die übergeordnete intrinsische Informationsklasse der Trainingsdaten wurde auf die in iiRDS definierten Lebenszyklusphasen (**`iirds:ProductLifecyclePhase`**) abgebildet, z. B. Wartungsinformationen → **`iirds:Maintenance`**.

---

[90] Im Englischen Original: "classifying"

[91] Mit der RfC-Version von iiRDS wurde auch eine Auswahl an Selektoren aus Web Annotation direkt integriert, um die hier beschriebenen Szenarien „mit Bordmitteln“ umsetzen zu können.

4.5.6.3 Integration

Der Body-Teil einer Annotation wird verwendet, um direkt auf eine Instanz einer Informationstyp-Klassifikation von iiRDS oder eine bestimmte Komponente der Produktontologie zu referenzieren. Auf diese Weise werden alle bestehenden Standards genutzt, indem sie für den vorliegenden Anwendungsfall zu einem integrierten Informationsframework zusammengefasst werden. Der Target-Teil der Annotation enthält Selektoren, um genau zu definieren, worauf sich eine Annotation bezieht. Für die drei im Testaufbau verwendeten Datenquellen werden verschiedene Selektoren verwendet.

Module werden direkt als **`oa:SpecificResource`** (ein Modul pro Datei) oder mit einem **`oa:XPathSelector`** oder **`oa:FragmentSelector`** als Teil einer Sammlung von Modulen innerhalb einer Datei referenziert (Abbildung 29).

Klassifizierte Abschnitte von PDF-Dateien verwenden ein Tupel aus **`oa:FragmentSelector`**[92], um Start- und Endseiten eines **`oa:RangeSelectors`** in einem Dokument zu definieren (vgl. Abbildung 30). Serviceberichte werden aufgrund der engen Integration mit der Benutzeroberfläche zur Autovervollständigung direkt als **`oa:TextualBody`** mit der Annotation gespeichert (vgl. Abbildung 28). Die Annotationen von Serviceberichten sind mit allen ausgewählten Ontologiekonzepten der Domäne verknüpft.

Für alle Datenquellen wird dem Target-Teil ein Textauszug (**`oa:TextQuoteSelector`**) als Verfeinerung (**`oa:refinedBy`**) zu den oben genannten quellenspezifischen Selektoren hinzugefügt. Dieser zusätzliche Selektor ermöglicht eine schnelle Volltextsuche über alle Ressourcen mit nativen SPARQL-Methoden und kann die Genauigkeit bei niedrig auflösenden Selektoren wie Seitenbereichen weiter verbessern. Darüber hinaus kann die Verwendung von Textauszügen die Robustheit gegenüber Änderungen am Quelldokument erhöhen (BRUSH, BARGERON, GUPTA & CADIZ, 2001).

Um die gesetzlichen Vorschriften für Technische Dokumentation einzuhalten, wird ein Verweis auf die Originalquelle **`oa:hasSource`** gespeichert, um ein vertrauenswürdiges Ranking der Filterergebnisse oder eine spezifische Kennzeichnung kanonischer Quellen (unkontrollierte vs. kontrollierte Dokumente) zu ermöglichen. Darüber hinaus ist der Ursprung der Annotation (**`dcterms:creator`**) entweder mit der Software (zur Klassifizierung und Segmentierung) oder der Person (zur Autovervollständigung) verbunden.

[92] Fragment-Selektoren wurde granulareren Alternativen vorgezogen, da sie von allen PDF-konformen Anzeigeprogrammen und Browsern unterstützt werden.

Alle generierten Annotationen aus Serviceberichten, Modulen und Dokumentbereichen werden auf einem „Linked Data Platform Server“[93] gespeichert, der mit SPARQL abgefragt werden kann. Dadurch sind die Annotationen für Lese-, Schreib- und Abfrageoperationen für jeden Client mit Linked-Data-Platform- oder SPARQL-Unterstützung zugänglich. Darüber hinaus werden Links zu externen Ressourcen bereitgestellt und Verweise auf Informationsressourcen Dritter können problemlos hinzugefügt werden. Alternativ können die generierte Annotationen auch im JSON-LD-Format (W3C, 2014c) für den Austausch mit anderen Systemen ausgegeben werden.

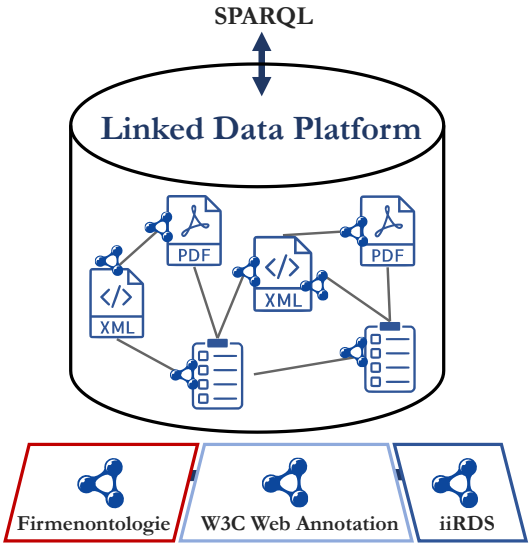

Abbildung 31: Einheitlicher semantischer Zugriff über SPARQL.
Die Linked Data Platform basiert auf Konzepten aus Web Annotation, iiRDS und einer individuellen Firmen-Ontologie, die miteinander verknüpft werden.

4.5.7 Anwendung

Die fortschreitende Automatisierung der industriellen Fertigung führt zu einer hohen Abhängigkeit von zuverlässig arbeitenden Produktionsanlagen. Im Gegensatz zu den Bemühungen der Industrie, die Produktion zu automatisieren und zu digitalisieren, hängt der Instandhaltungsprozess jedoch immer noch stark von der Kompetenz und Erfahrung einzelner Personen ab. Mit der zunehmenden Vielfalt der installierten Maschinentypen und -konfigurationen wird die kontextsensitive Bereitstellung von Informationen zu einem Wettbewerbsvorteil und das dazugehörige Wissensmanagement zur Notwendigkeit.

[93] vgl. http://marmotta.apache.org/ [01.02.19]

Unternehmen verwenden bereits Metadaten, um verschiedenste Inhalte mit zusätzlichen Informationen anzureichen. Doch um das vorhandene Wissen eines Unternehmens effektiv zu nutzen, müssen statische Dokumente mit weniger formalisierten Quellen kombiniert werden, die sich im Lebenszyklus eines Produktes auch verändern können. Darüber hinaus müssen Informationsmodule aus verschiedenen Unternehmen untereinander ausgetauscht werden (z. B. von Zulieferer zu Kunden), um Wissen schnell und einfach zugänglich zu machen.

Linked-Data-Technologien ermöglichen die einfache Verknüpfung beliebiger Informationen mit Hilfe von Webstandards und damit den universellen und unkomplizierten Zugriff darauf. Die Beschränkung auf ein standardisiertes Annotationsvokabular garantiert darüber hinaus den unternehmensweiten Austausch von technischem Wissen und ein gemeinsames Verständnis von der Bedeutung der Metadaten. Gerade in modernen Lieferketten mit vielen Komponentenherstellern und hoher Produktvariabilität ist die gezielte Bereitstellung von Informationen in geeigneter Größe entscheidend.

Eine der Hauptanwendungen für die semantische Annotation heterogener Datenquellen sind Content-Delivery-Portale (CDP), die sich auf die gezielte Bereitstellung von Informationen für Kunden, interne Mitarbeiter und Techniker konzentrieren (ZIEGLER & BEIER, 2015). Die steigende Nachfrage nach CDPs führt zur Entwicklung konkurrierender Standards für Inhalte und Metadatenformate. Ein unabhängiges und offenes Informationssystem als Grundlage für spezialisierte Systeme kann eine ungewollte Lieferantenbindung verhindern und die Austauschbarkeit von Daten erhöhen. In der vorgestellten Lösung können Kunden die Daten entweder durch ein Traversieren des Konzeptgraphen auf die Daten zugreifen oder durch strukturierte Abfragen via SPARQL (vgl. Codebeispiel 6, der **UNION**-Teil kombiniert die Annotationen aus der automatisierten Klassifizierung und denen der unterstützten Berichtserstellung).

```
SELECT ?text WHERE {
  ?annotation oa:hasBody iirds:Maintenance .
  ?annotation oa:hasBody :offset_unit .
  ?annotation oa:hasTarget ?target .
  { ?target oa:refinedBy ?textselector . }
  UNION
  { ?target oa:hasSelector ?selector .
    ?selector oa:refinedBy ?textselector . }
    ?textselector oa:exact ?text .
  FILTER (regex(str(?text), "verschmutzt", "i")) }
```

Codebeispiel 6: SPARQL-Abfrage mit Annotationsfilter.
Dargestellt ist ein Beispiel für die Suche nach Informationen in der Produktlebenszyklusphase „Wartung“ (iirds:Maintenance) zu einer verschmutzten Offset-Einheit.

4.5.8 Implementierung

Funktionsfähige Prototypen sind sowohl für die semantische Autovervollständigung[94] als auch für die automatisierte Klassifizierung[95] verfügbar und öffentlich zugänglich (siehe z. B. Abbildung 32 für die Segmentierungs-GUI).

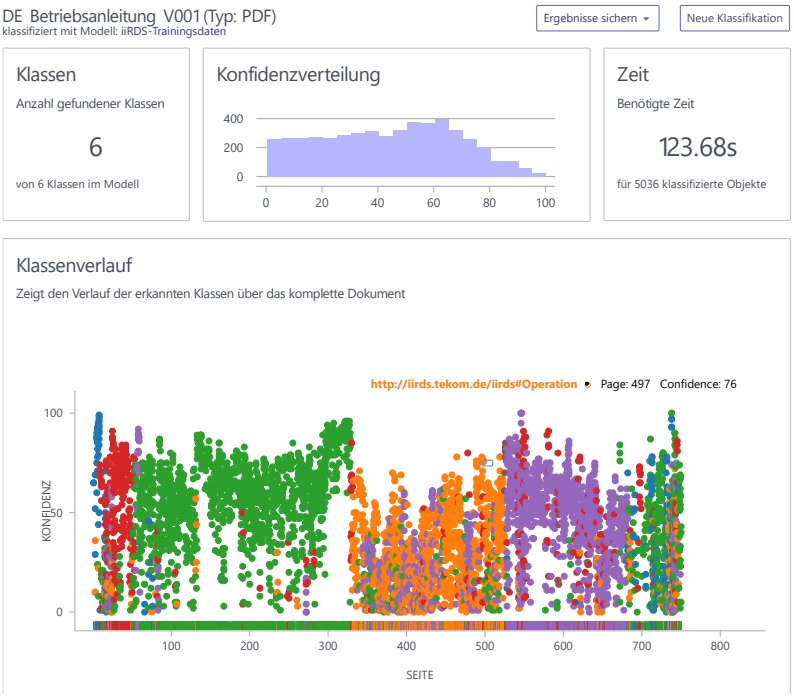

Abbildung 32: Screenshot. Oberfläche für die automatisierte Segmentierung. Mit iiRDS-Annotationen für ein Echtdatenbeispiel.

4.5.9 Evaluierung

Die vorgestellte Lösung wurde in einem zweistufigen Prozess evaluiert. Zunächst werden die semantische Autovervollständigung und die automatisierte Klassifizierung von Technischer Dokumentation gesondert betrachtet. In einem zweiten Experiment wird das kombinierte Informationsframework bewertet, indem relevante Dokumente sowohl aus kontrollierten Quellen (Module oder Dokumente) als auch aus unkontrollierten Quellen (Serviceberichte) abgefragt werden.

---

[94] vgl. https://github.com/sebbader/SemanticAutocompletion (Quellcode) [01.02.19]

[95] vgl. https://github.com/j-oe/semantics-demo (Quellcode) [01.02.19];
eine gehostete Demo ist verfügbar unter: http://semantics.fastclass.de/ [01.02.19]

4.5.9.1 Unterstützte Berichterstellung

Für die Auswertung der unterstützten Berichterstellung wurden die eingegebenen Texte und Metadaten im Hinblick auf die korrekte Erwähnung der beteiligten Techniker und Kunden, der betroffenen Maschinen und Baugruppen sowie der gefundenen Fehlercodes und der durchgeführten Aktionen analysiert. Ohne die semantische Autovervollständigung werden durchschnittlich $3{,}13 \pm 0{,}49$ Konzepte korrekt benannt. Im Gegensatz dazu werden $3{,}92 \pm 0{,}95$ Konzepte in Berichten mit aktivierter semantischer Autovervollständigung korrekt erwähnt. Auch bei einer Lockerung der Bewertungskriterien sind $3{,}90 \pm 0{,}47$ Konzepte ähnlich im Sinne einer falschen Reihenfolge von Begriffen oder fehlenden Textteilen (z. B. nur „670" statt „ICS 670"), während der Durchschnitt der nahezu richtigen Konzepte mit aktivierter Autovervollständigung bei etwa $4{,}41 \pm 0{,}62$ liegt. Da jeder Servicebericht nur ein oder zwei Sätze enthält, hilft die mitgelieferte Funktionalität vor allem ungeübten Anwendern, die richtigen Benennungen besser zu finden.

4.5.9.2 Annotation von Technischer Dokumentation

Die automatisierte Klassifizierung und Segmentierung und damit die Qualität der erzeugten semantischen Annotationen unterliegt der Klassifikatorleistung. Um die Ausgabequalität zu messen, wurde eine 10-fache Kreuzvalidierung durchgeführt und die durchschnittliche Genauigkeit (SOKOLOVA & LAPALME, 2009) der vorhergesagten Klassifikationen berechnet, was zu $85{,}3\% \pm 2{,}5$ für Informationstypen ($n = 6$) und $82{,}5\% \pm 2{,}1$ für produktbezogene Klassen ($n = 28$) führte. Das verwendete Trainingsset enthielt 3.686 XML-basierte Module, die von technischen Redakteuren manuell klassifiziert wurden. Eine geringe Genauigkeit in der Klassifizierung beeinflusst auch die verbreiteten Qualitätsmetriken im Information Retrieval: Precision und Recall.[96] Um dies zu berücksichtigen, werden die Testabfragen zusätzlich gegen manuell klassifizierte Inhalte aus den Trainingsdaten durchgeführt, um eine Baseline der Relevanz zu erhalten

4.5.9.3 Integriertes Informationsframework

Das Gesamtsystem, das aus beiden Modulen besteht, wird mit zehn Abfragen getestet, die auf realen Wartungsanfragen bzw.-aufgaben von Serviceanbietern basieren. Für die Evaluierung werden Precision und Recall von Stichwort-Abfragen (engl.: „Keyword queries") mit einer erweiterten Abfrage verglichen, die mit einer Filterung von semantischen Annotationen ergänzt wurde. Hauptaspekte für die Suche sind die Art der gesuchten Informationen

[96] Zur Definition von Precision und Recall siehe Abschnitt 2.6.

(Dokumente über Wartung, technische Daten, Betrieb usw.)[97] und die beschriebene Baugruppe oder Komponente des Produkts.

Es werden drei Varianten für jede Abfrage verglichen. Zunächst wird eine einfache Stichwort-Abfrage auf dem SPARQL-Endpunkt der Linked Data Platform mit der SPARQL-**`FILTER`**-Funktionalität und einer RegEx[98]-Suche durchgeführt. Der zweite Abfragetyp kombiniert sowohl das Schlüsselwort als auch den Filterterm in einer zusätzlichen Filteranweisung. Beide Abfragetypen können auf jeder Wissensbasis durchgeführt werden und profitieren nicht von den vorgeschlagenen automatisierten Annotationen. Infolgedessen können die erzielten Ergebnisse als Baseline für den dritten Abfragetyp dienen; dieser nutzt die Ergebnisse des integrierten Informationsframeworks. Codebeispiel 6 enthält eine solche Abfrage, bei der die Schlüsselwortsuche - über eine Filteranweisung - durch die Aussage in der zweiten Zeile weiter eingeschränkt wird. Dabei werden in diesem Beispiel nur Informationsmodule mit einer **`iirds:Maintenance`**-Annotation betrachtet.

Eine einfache Stichwortsuche ohne zusätzliche Filterung liefert die meisten Ergebnisse. Folglich liegt der Recall dieser Abfrage immer sehr nahe bei 100%. Da aber viele abgerufene Texte keine relevanten Informationen in Bezug auf das Informationsbedürfnis enthalten, kann nur ein niedriger Precision-Wert erreicht werden. Tatsächlich liegt das durchschnittliche F_1-Maß[99] bei nur etwa $0{,}46$. Die Suche nach Vorkommen des Schlüsselwortes in Kombination mit einem zusätzlichen Kontextterm (Filterkriterium) führt zu unterschiedlichen Beobachtungen. Dadurch ist die Anzahl der zurückgelieferten Ergebnisse sehr gering, da das Filterkriterium in der Regel nicht oft in den Texten selbst vorkommt. So enthalten z. B. Dokumente und Berichte über Wartungsmaßnahmen in der Regel nicht in jedem Absatz den Term "Wartung".[100] Dies führt zu einer deutlich reduzierten Anzahl von Ergebnissen, was die Precision erhöht, aber zu einem geringen Recall führt. Insgesamt wird ein F_1-Maß von nicht mehr als $0{,}42$ erreicht.

---

[97] Dies entspricht den iiRDS-Konzepten **`iirds:ProductLifeCyclePhase`** und **`iirds:InformationSubject`**.

[98] RegEx steht für reguläre Ausdrücke, die in der Suche eingesetzt werden können, um ein bestimmtes Zeichenmuster abzufangen.

[99] Kombination aus Precision und Recall, definiert als $F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$.

[100] Außnahmen sind Sonderfälle wie Kopf- und Fußzeilen oder wiederkehrende Überschriften.

*Tabelle 13: Ergebnisse für verschiedene Abfragearten. Durchgeführt innerhalb des integrierten Informationsframeworks mit zehn typischen Queries für Servicetechniker. Beste Ergebnisse sind **fett** hervorgehoben.*

Abfrageart	Metrik	Query 1	Query 2	Query 3	Query 4	Query 5	Query 6	Query 7	Query 8	Query 9	Query 10	Ø
Nur Stichwort	Precision	0,88	0,15	0,19	0,05	0,60	0,15	0,03	0,85	0,07	0,87	0,383
	Recall	1	1	1	1	1	1	1	1	1	1	**1**
	F_1	0,93	0,26	0,32	0,09	0,75	0,26	0,07	0,92	0,13	0,93	0,465
Stichwort und Kontextterm	Precision	1	1	1	1	0	0	1	1	1	0,75	**0,775**
	Recall	0,36	0,56	0,64	0,46	0	0	0,11	0,55	0,14	0,23	0,305
	F_1	0,53	0,72	0,78	0,63	0	0	0,19	0,71	0,25	0,35	0,416
Stichwort und Annotationsfilter	Precision	1	1	1	1	1	0,67	1	1	1	1	**0,967**
	Recall	0,93	0,54	0,57	0,54	1	1	0,89	0,18	0,86	0,92	0,743
	F_1	0,96	0,70	0,73	0,70	1	0,80	0,94	0,31	0,92	0,96	0,803

Die Kombination einer Stichwortsuche mit der Filterung von standardisierten Annotationen führt zu einem höheren Recall als jede der Baseline-Abfragen. Obwohl einige falsche Ergebnisse zurückgeliefert werden (was die Precision im Vergleich zum zweiten Abfragetyp verringert), wird dies durch die Steigerung des Recall kompensiert. Mit einer durchschnittlichen Precision von 0,96 und einem durchschnittlichen Recall von 0,74 schlägt das resultierende F_1-Maß von 0,80 deutlich die Baseline-Abfragen. Dieses Ergebnis bestätigt die Annahme, dass ein Informationsframework wie das hier vorgeschlagene - auch wenn es sich noch in einem prototypischen Stadium befindet - einen wesentlichen Unterschied bei den Ergebnissen bewirken kann.

Alle Suchabfragen und Ergebnismetriken sind in einem GitHub-Repository veröffentlicht.[101] Da die konkreten Resultate der Abfragen vertrauliche Daten enthalten, können diese nicht zugänglich gemacht werden.

4.5.10 Fazit

Durch die Nutzung von Linked-Data-Prinzipien und standardisierten Vokabularen konnte ein standardisiertes Modell für die semantische Annotation von digitaler Technischer Dokumentation und Serviceberichten entwickelt werden. Es wurden zwei Ansätze erfolgreich kombiniert, um diese Herausforderung zu meistern: Für unstrukturierte Berichte wurde eine ontologiebasierte Autovervollständigung für die unterstützte Berichtserstellung verwendet; für kontrollierte Inhalte wurden Methoden des maschinellen Lernens benutzt, um bestehende Technische Dokumentation durch automatisierte Klassifizierung von Modulen und Segmentierung von dokumentenbasierten Formaten semantisch zugänglich zu machen. Beide Ansätze werden mit aktuellen Linked-Data-Standards zu einem integrierten Informationsframework kombiniert.

Eine erste Evaluierung des vorgeschlagenen Modells zeigt, dass standardisierte semantische Annotationen den Umgang mit heterogenen Datentypen verbessern und damit die Einsatzzeit von Servicetechnikern reduzieren können. Durch den höheren Recall bei der Informationsrecherche werden wichtige Informationen schneller gefunden und Techniker können mehr Zeit für die eigentliche Tätigkeit verwenden, anstatt heterogene Wissensquellen mühevoll zu durchsuchen.

4.5.11 Ausblick

In zukünftigen Arbeiten soll die Flexibilität des vorgeschlagenen Modells durch das Hinzufügen weiterer externer Datenquellen validiert werden. Um die Vorteile von Linked Data vollständig zu nutzen, könnten die Annotationsobjekte mit Wissen aus frei verfügbaren Quellen wie der *Linked Open Data Cloud* angereichert werden. Ebenso kann ein schnelles Modell zur Anbindung von geschlossenen Unternehmens-Ontologien bereitgestellt werden. Darüber hinaus soll der Fokus künftiger Arbeit auf komplexeren Abfragen liegen, die mehrere Annotationen kombinieren und mit einer leistungsfähigen Volltextsuche verbinden. Es ist außerdem geplant, das vorgeschlagene Framework als *Backend* für Serviceinformationssysteme im Praxiseinsatz zu implementieren.

[101] https://github.com/j-oe/semantics-queries [01.02.19]

5 Zusammenfassung und Ausblick

5.1 Zusammenfassung

In der vorliegenden Arbeit wurde die Optimierung des semantischen Informationszugriffs auf Technische Dokumentation eingehend erforscht. Praxisrelevanz erhält das Thema durch die Tatsache, dass mit verbesserten Retrieval-Eigenschaften von Content die Recherchezeiten verkürzt, viele Informationsprozesse automatisiert und neuartige Anwendungen realisiert werden können. Im aktuellen Umfeld der Digitalen Transformation und Industrie 4.0 ermöglicht eine entsprechend optimierte Datenbasis die Bereitstellung von digitalen Informationsservices, welche internen und externen Anwendern bereitgestellt werden können.

Als Grundlage wurden für den Bereich der Technischen Dokumentation domänenspezifische Konzepte und Optimierungsmöglichkeiten gesammelt und zusammen mit Methoden und softwaregestützten Umsetzungen in eine Übersicht eingeordnet („CoSMOS“). Ausgehend von diesem Modell mit den Ebenen Content, Struktur, Metadaten und Ontologien wurden vier konkrete Untersuchungen im Detail vorgestellt, die mit Hilfe von Maschinellem Lernen und Linked-Data-Technologien die Optimierungen für den semantischen Informationszugriff automatisieren.

Schwerpunkt aller durchgeführten Arbeiten war die domänenspezifische Anwendung im Bereich der Technischen Dokumentation, zum einen was Eigenschaften der Textsorte und deren Form betrifft, zum anderen aber auch was rechtliche Rahmenbedingungen oder typische Anwendungsfälle angeht. Sowohl das entstandene Modell als auch die Detailergebnisse der einzelnen Untersuchungen können so ein umfassendes Bild der Optimierungsmöglichkeiten zeichnen und als Basis für weitere Forschung dienen.

5.1.1 Zusammenfassung der Untersuchungen

Die durchgeführten Untersuchungen zur Automatisierung werden mit Bezug auf ihre Auswirkungen beim Informationszugriff und ihre domänenspezifischen Ausprägungen wie folgt zusammengefasst:

- Unkontrollierte Varianten und Duplikate können mit Hilfe einer semantischen Ähnlichkeitsanalyse aus dem Datenbestand entfernt werden. Dadurch kann vermieden werden, dass Anwender bei einer Suche mit ähnlichen oder doppelten Ergebnissen konfrontiert und damit verunsichert werden. Die in der Technischen Dokumentation häufig vorkommenden (beabsichtigten) Content-Varianten können durch eine semantische Gewichtung bestimmter Textteile identifiziert und von ungewollten Varianten unterschieden werden.

- Monolithische Dokumente, die in der Regel als PDF-Version der gedruckten Dokumentation archiviert werden, sind weit verbreitet, aber hindern Anwender an einem granularen Informationszugriff. Um die oft hunderte Seiten langen PDF-Dokumente besser zugänglich zu machen, kann eine automatisierte Segmentierung thematisch zusammenhängende Seitenbereiche erkennen und mit semantischen Konzepten annotieren, um so eine Suche einschränken zu können.

- Content ohne Metadaten lässt sich nur auf Basis des darin vorkommenden Texts durchsuchen (Volltextsuche), häufig werden aber für eine effektive Suche zusätzliche Facetten wie Art oder Thema der Information benötigt, um Ergebnisse zu filtern. Eine automatisierte Klassifizierung weist automatisiert klassifizierende Metadaten zu und berücksichtig dabei Eigenheiten der Technischen Dokumentation auf sprachlicher und technischer Ebene zur Optimierung der Ergebnisse.

- Verteilte Informationsquellen mit heterogenen Inhalten und Metadatenformaten erschweren eine effektive Suche enorm. Durch die kombinierte Anwendung der vorgestellten Methoden und einer einheitlichen semantischen Annotation im Rahmen eine übergreifenden Ontologie kann eine Informationsintegration erreicht werden, die es erlaubt, strukturierte Suchen über den gesamten Datenbestand durchzuführen und damit die Präzision von Abfragen erheblich zu verbessern, was besonders im Bereich des industriellen Service von großem Nutzen ist.

Im Rahmen der Arbeit wurde seit 2016 der semantische Austauschstandard für digitale Technische Dokumentation *iiRDS* mitentwickelt. Erkenntnisse aus den Untersuchungen sind in die Spezifikation miteingeflossen, wie z. B. die Selektor-Logik für die Annotation von Seitenbereichen (BECKER u. a., 2018, Abschn. 6.3.1), Forschungen zur automatisierten Zuweisung von Konzepten (OEVERMANN, 2017b) oder die Integration mit anderen Metadatenstandards über definierte Mappings (IIRDS CONSORTIUM, 2018).

Obwohl die Untersuchungen isoliert betrachtet werden können, verfolgen sie ein gemeinsames Ziel: die semantische Informationsaufbereitung zum Zwecke des Information Retrieval. Allen Betrachtungen zuteil ist die Spezialisierung auf die Domäne Technische Dokumentationen mit den dort verbreiteten Methoden und Werkzeugen unter Berücksichtigung der gesetzten Rahmenbedingungen. Aus diesem Grund sind in vielen Fällen Verfahren gewählt worden, die zum einen gut anpassbar und zum anderen leicht implementierbar (und somit evaluierbar) sind. Da sich der Bereich des Maschinellen Lernens sehr schnell bewegt, wurde der Fokus hierbei auf etablierte Methoden gelegt, deren Funktionsweise nachvollziehbare Anpassungen zulässt.

5.1.2 Übergreifende Ergebnisse

Die folgenden übergreifenden Ergebnisse konnten im Rahmen dieser Arbeit ermittelt werden:

- Technische Dokumentation kann auf vielen Ebenen für einen semantischen Informationszugriff optimiert werden, begonnen bei grundlegenden Texteigenschaften über Struktur und Auszeichnung bis hin zur Einordnung in ein Beziehungsnetz (vgl. Abschnitt 3.2).
- Die Optimierung des Contents für das Information Retrieval beginnt mit manuellen Methoden, die dann softwaregestützt kontrolliert, automatisiert oder analysiert werden können (vgl. Abschnitt 3.1).
- Durch Anwendung der Kosinusähnlichkeit lassen sich effiziente Ähnlichkeitsanalysen mit Webtechnologien umsetzen und mit einer an den redaktionellen Workflow angepassten Benutzeroberfläche erfolgreich in Industrieprojekten einsetzen (vgl. Abschnitt 4.2.6).
- Die Mikrostrukturen von semantischen XML-Informationsmodellen können für eine semantische Gewichtung bei Ähnlichkeitsanalysen verwendet werden, um effizient und zuverlässig gewollte von ungewollten Varianten zu unterscheiden. (vgl. Abschnitt 4.2.5).
- Als Merkmale für eine statistische Sprachverarbeitung von Technischer Dokumentation liefern „Bag-of-n-grams“-Modelle mit einzelnen nichtlemmatisierten Wörtern und Wortgruppen ($n = \{1,2,3\}$) die besten Ergebnisse. Ähnlich gute Ergebnisse bei besserer Performance liefert die Konfiguration $n = 2$ (vgl. Abschnitt 4.3.6.1).
- Für die automatisierte Multiklassen-Klassifizierung von Technischer Dokumentation eignet sich als Merkmalsgewichtung am besten die vorgestellte $\mathrm{TF} - \mathrm{ICF} - \mathrm{CF}$-Methode (vgl. Abschnitt 4.3.6.3).
- Als zuverlässiger Indikator für den erreichbaren Automatisierungsgrad bei einer automatisierten Klassifizierung dient das vorgestellte Konfidenzmaß. Es kann dazu verwendet werden, Ergebnisse mit hoher Konfidenz automatisiert weiterzuverarbeiten, während Klassifikationen mit geringer Konfidenz eine menschliche Kontrolle benötigen (vgl. Abschnitt 4.3.7.4).
- Bei guten Trainingsdaten können über verhältnismäßig simple Verfahren bei der automatisierten Klassifizierung Ergebnisse von bis zu 92% Genauigkeit (*mean accuracy*) mit den vorliegenden Echtdatensätzen erzielt werden (vgl. Abschnitt 4.3.7.1).

- Durch die Anwendung von vorhergesagten Klassifikationen und deren Konfidenz lassen sich unstrukturierte Dokumente bei ausreichender Genauigkeit und ohne die Auswertung von Formatierungsinformationen automatisiert segmentieren (vgl. Abschnitt 4.4).
- Automatisch erkannte Segmente in einer PDF-Datei lassen sich mit iiRDS als semantische Annotationen außerhalb der Originaldatei modellieren und austauschen, was in der Technischen Dokumentation sehr wichtig sein kann (vgl. Abschnitt 4.4.4.7).
- Die vorgestellten Methoden lassen sich durch Linked-Data-Konzepte mit anderen Verfahren (z. B. semantischer Autovervollständigung) kombinieren, um einen einheitlichen Zugriff auf mehrere heterogene Datenquellen zu ermöglichen (vgl. Abschnitt 4.5).
- Strukturierte Suchen mit semantischen Annotationen liefern bei typischen Informationsrecherchen von Servicetechnikern präzisere Ergebnisse als reine Volltextsuchen (vgl. Abschnitt 4.5.9.3).

Abschließend lässt sich sagen, dass Content der Technischen Dokumentation spezielle Anforderungen hat, was die Optimierung hinsichtlich eines semantischen Informationszugriff betrifft.

5.1.3 Kritische Betrachtung

Bei dem zu Beginn eingeführten CoSMOS-Modell handelt es sich im engeren Sinne um ein Schema, in das sich Methoden und Anwendungen einordnen und bewerten lassen. Da die zugrunde liegenden Annahmen auf etablierten Konzepten der Technischen Dokumentation basieren, dient es als strukturierte Übersicht und Zusammenstellung von Optimierungen. Jedoch stellen die Ebenen oder die darin referenzierten Methoden keine Neuerungen an sich dar. In dieser Arbeit wird das CoSMOS-Modell vor allem dafür verwendet, die selbst durchgeführten Untersuchungen in einem Kontext zu verorten.

Bei den durchgeführten Evaluierungen wurden Echtdaten verwendet, die von Firmen bereitgestellt wurden. Dadurch konnten praxisrelevante Ergebnisse ermittelt werden, die auf Grund der Vertraulichkeit der Daten jedoch nicht unmittelbar durch Dritte reproduziert werden können. Zur Vergleichbarkeit der domänenspezifischen Ansätze in der Technischen Dokumentation werden deshalb frei verfügbare Referenz-Datensätze benötigt. Solche Benchmarks sind in anderen Feldern bereits üblich. Wie schon in der Zusammenfassung erwähnt, wurden für die Untersuchungen nicht immer *State-of-the-Art* ML-Technologien verwendet, da der Fokus auf einer nachvollziehbaren Anpassung der Parameter lag. Hier sollten in zukünftigen Arbeiten auch neuere Ansätze (z. B. künstliche neuronale Netze) zum Einsatz kommen.

5.2 Ausblick

Die Ausweitung der Untersuchungen auf weitere Methoden, die in der einleitenden Übersicht genannt sind, kann das Gesamtbild auf die Optimierungen für den semantischen Informationszugriff vervollständigen. Speziell die Auswirkungen von Struktur und Standardisierungsmethoden auf die Effizienz einer Informationsrecherche sind im Gegensatz zu Metadaten und Semantik noch wenig erforscht. Hier sollte neben Assistenz- und Analysewerkzeugen auch das vorhandene Potenzial zur Automatisierung ausgenutzt werden.

Bestehende Kooperationen mit anderen Forschungsgruppen zur Evaluierung der Effekte, die Strukturmuster, Informationsmodelle und Sprachen auf das Information Retrieval haben, werden ebenso weiter ausgebaut wie die Untersuchungen im Bereich der Qualitätsmetriken von Content und Klassifikationssystemen. In weiterführenden Experimenten ist geplant, die verwendeten ML-Methoden (z. B. zur automatisierten Klassifizierung) mit modernen Frameworks und alternativen Lernmethoden zu vergleichen. Der Einsatz der bereits entstandenen Prototypen in der Praxis kann zusätzliche Optimierungspotenziale bezüglich neuer Anwendungsfälle liefern.

Eine Anpassung der untersuchten Verfahren auf andere domänenspezifische Anwendungen oder Inhaltsarten, wie z. B. Patente, Normen oder Ausschreibungsunterlagen, ist ein logischer nächster Schritt für zukünftige Forschung. Ähnliche Herausforderungen sind hierbei der semantische Zugriff auf eine große Menge an unstrukturierten Inhalten mit speziellen Textcharakteristiken und die automatisierte Annotation mit ontologischen Konzepten. Die Ergebnisse dieser Arbeit können dabei als Grundlage dienen.

6 Literaturverzeichnis

6.1 Publikationen im Rahmen der Dissertation

Peer-Review

BADER, Sebastian, OEVERMANN, Jan (2017): Semantic Annotation of Heterogeneous Data Sources: Towards an Integrated Information Framework for Service Technicians. In: *Proceedings of the 13th International Conference on Semantic Systems* (S. 73–80). SEMANTiCS2017, Amsterdam, Netherlands: ACM
DOI:10.1145/3132218.3132221

OEVERMANN, Jan (2016): Reconstructing Semantic Structures in Technical Documentation with Vector Space Classification. In: *Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems* (Bd. 1695). SEMANTiCS 2016, Leipzig: CEUR-WS
URN:nbn:de:0074-1695-3

OEVERMANN, Jan (2018): Semantic PDF Segmentation for Legacy Documents in Technical Documentation. In: *Procedia Computer Science. Proceedings of the 14th International Conference on Semantic Systems* (Bd. 137, S. 55–65). SEMANTiCS 2018, Vienna, Austria: Elsevier
DOI:10.1016/j.procs.2018.09.006

OEVERMANN, Jan, LÜTH, Christoph (2018): Semantically Weighted Similarity Analysis for XML-based Content Components. In: *Proceedings of the ACM Symposium on Document Engineering 2018* (S. 1–4). DocEng '18, Halifax, Canada: ACM
DOI:10.1145/3209280.3229098

OEVERMANN, Jan, ZIEGLER, Wolfgang (2016): Automated Intrinsic Text Classification for Component Content Management Applications in Technical Communication. In: *Proceedings of the 2016 ACM Symposium on Document Engineering* (S. 95–98). DocEng '16, Vienna, Austria: ACM
DOI:10.1145/2960811.2967153

OEVERMANN, Jan, ZIEGLER, Wolfgang (2018): Automated Classification of Content Components in Technical Communication. *Computational Intelligence*, *34* (1), S. 30–48: Wiley: New Jersey, USA.
DOI:10.1111/coin.12157

Fachbeiträge

OEVERMANN, Jan (2017): Künstliche Intelligenz und intelligente Textservices. In: Jörg HENNIG, Marita TJARKS-SOBHANI (Hrsg.): *Intelligente Information* (S. 82–94). Stuttgart: tcworld
ISBN: 978-3-944449-60-9

OEVERMANN, Jan (2018): Technical Documentation 4.0 - How the Latest Trends in Manufacturing Lead to New Paradigms for Technical Writers. *tcworld*, (February 2018), S. 12–15
Online verfügbar unter: URL: http://www.tcworld.info/e-magazine/technical-communication/article/technical-documentation-40/ [08.02.19]

Andere Veröffentlichungen

BECKER, Frauke, KREUTZER, Martin, NUDING, Win, OEVERMANN, Jan, PARSON, Ulrike, SAPARA, Jürgen, u. a. (Hrsg.) iiRDS Specification - intelligent information Request and Delivery Standard - tekom Standard - 18 April 2018. (2018) Online verfügbar unter: URL: https://iirds.org [08.02.19]

OEVERMANN, Jan (2016): Intelligente Klassifizierung von technischen Inhalten – Automatisierung und Anwendungspotenziale. In: *Tagungsband der tekom Jahrestagung 2016*. tekom, Stuttgart: tcworld

OEVERMANN, Jan (2017a): iiRDS voll automatisch – mit Machine Learning zum intelligenten Austauschformat. In: *Tagungsband der tekom Jahrestagung 2017*. tekom, Stuttgart: tcworld

OEVERMANN, Jan (2017b): Smart Content Delivery – Von Dokumentation zu intelligenter Information mit Machine Learning. In: *Tagungsband der tekom Frühjahrestagung 2017*. tekom, Stuttgart: tcworld

OEVERMANN, Jan, FLESCHUTZ-BALAREZO, Timo (2018): Großputz im CMS! Semantische Ähnlichkeitsanalyse für XML-Module. In: *Tagungsband der tekom Jahrestagung 2018*. tekom, Stuttgart: tcworld

Verfügbarkeit

Alle hier gelisteten Publikation (teilweise als Preprint) können auch auf der Website des Autors heruntergeladen werden: http://janoevermann.de

6.2 Vollständige Bibliographie

2006/42/EG „EU-Maschinenrichtlinie“ – Richtlinie 2006/42/EG des Europäischen Parlaments und des Rates vom 17. Mai 2006 über Maschinen und zur Änderung der Richtlinie 95/16/EG (Neufassung). (2006)

ADOBE SYSTEMS (Hrsg.) (2001): *PDF reference: Adobe portable document format version 1.4*. Boston, USA: Addison-Wesley

ADRIAN, Benjamin (2018): Knowledge Packs: Wie Anwendungen durch Daten intelligenter werden. *EMPOLIS Blog*. Online verfügbar unter: URL: https://www.empolis.com/blog/kuenstliche-intelligenz/empolis-knowledge-packs-technik/ [29.01.2019]

ALLAN, James, HARPER, David J., HIEMSTRA, Djoerd, HOFMANN, Thomas, HOVY, Eduard, KRAAIJ, Wessel, u. a. (2003): Challenges in Information Retrieval and Language Modeling. In: *Report of a Workshop Held at the Center for Intelligent Information Retrieval* (Bd. 37, S. 31–47). SIGIR Forum, Amherst, USA: ACM

ALLEN, Jeffrey (1999): Adapting the Concept of „Translation Memory“ to „Authoring Memory“ for a Controlled Language Writing Environment. In: *Proceedings of the 21. International Conference on Translating and the Computer* (Bd. 10–11). London, UK

ANDERSEN, Rebekka (2011): Component Content Management - Shaping the Discourse through Innovation, Diffusion, Research and Reciprocity. *Technical Communication Quarterly*, *20* (4), S. 384–411

ANSI Z535.6 American National Standard for Product Safety Information in Product Manuals, Instructions, and Other Collateral Materials. (2006)

ASD-STE10 ASD Simplified Technical English – Specification ASD-STE100 European Community Trade Mark No. 017966390 – International Specification for the Preparation of Technical Documentation in a Controlled Language. (2017) Online verfügbar unter: URL: http://www.asd-ste100.org/ [07.02.2019]

ATA iSPEC 2200 (2014): *iSpec 2200: Information Standards for Aviation Maintenance*. Online verfügbar unter: URL: https://publications.airlines.org/ [11.01.2019]

AUTEXIER, Serge, MÜLLER, Normen (2010): Semantics-based Change Impact Analysis for Heterogeneous Collections of Documents. In: *Proceedings of the 10th ACM Symposium on Document Engineering* (S. 97). DocEng ’10, Manchester, UK: ACM

BADER, Sebastian, OEVERMANN, Jan (2017): Semantic Annotation of Heterogeneous Data Sources: Towards an Integrated Information Framework for Service Technicians. In: *Proceedings of the 13th International Conference on Semantic Systems* (S. 73–80). SEMANTiCS 2017, Amsterdam, Netherlands: ACM

BAEZA-YATES, R., RIBEIRO-NETO, Berthier (1999): *Modern Information Retrieval*. New York, USA: Addison-Wesley

BAILIE, Rahel Anne, HUSET, Jeffrey (2015): The Effect of CMS Technology on Writing Styles and Processes: Two Case Studies. *IEEE Transactions on Professional Communication*, *58* (3), S. 309–327

BAINES, Tim S., LIGHTFOOT, Howard W., BENEDETTINI, Ornella, KAY, John M. (2009): The Servitization of Manufacturing: A Review of Literature and Reflection on Future Challenges. *Journal of Manufacturing Technology Management*, *20* (5), S. 547–567

BECKER, Frauke, GÖTTEL, Sebastian, KREUTZER, Martin, NUDING, Win, OEVERMANN, Jan, PARSON, Ulrike, u. a. (2018): *iiRDS Specification - intelligent information Request and Delivery Standard - tekom Standard - 18 April 2018*. (TEKOM, Hrsg.) Online verfügbar unter: URL: https://iirds.org [11.01.2019]

BELKIN, Nicholas J. (2008): Some(what) Grand Challenges for Information Retrieval. In: *ECIR KEYNOTE* (Bd. 42, S. 47–54). European Conference on Information Retrieval, Glasgow, Scotland: ACM

BERNERS-LEE, Tim (2009): *Linked Data - Design Issues*. Online verfügbar unter: URL: https://www.w3.org/DesignIssues/LinkedData.html [01.02.2019]

BERNERS-LEE, Tim, HENDLER, James, LASSILA, Ora (2001): The Semantic Web. *Scientific american*, *284* (5), S. 34–43

BERNSTEIN, Yaniv, ZOBEL, Justin (2005): Redundant Documents and Search Effectiveness. In: *Proceedings of the 14th ACM International Conference on Information and Knowledge Management* (S. 736–743). CIKM '05, Bremen: ACM

BIRICIK, Göksel, DIRI, Banu, SÖNMEZ, Ahmet Coşkun (2009): A New Method for Attribute Extraction with Application on Text Classification. In: *Proceedings of the Fifth International Conference on Theory and Application of Soft Computing, Computing with Words and Perception* (S. 1–4). ICSCCW 2009, Famagusta, Cyprus: IEEE

BIRICIK, Göksel, DIRI, Banu, SÖNMEZ, Ahmet Coşkun (2012): Abstract Feature Extraction for Text Classification. *Turkish Journal of Electrical Engineering & Computer Sciences*, *20* (Sup. 1), S. 1137–1159

BLUMBERG, Donald F. (1994): Strategies for Improving Field Service Operations Productivity and Quality. *The Service Industries Journal*, *14* (2), S. 262–277

BRANTS, Thorsten, CHEN, Francine, TSOCHANTARIDIS, Ioannis (2002): Topic-based Document Segmentation with Probabilistic Latent Semantic Analysis. In: *Proceedings of the Eleventh International Conference on Information and Knowledge Management* (S. 211–218). CIKM '02, McLean, USA: ACM

BROUGHTON, Vanda (2006): The Need for a Faceted Classification as the Basis of all Methods of Information Retrieval. (Andy DAWSON, Hrsg.)*Aslib Proceedings: New Information Perspectives*, *58* (1/2), S. 49–72

BRUSH, AJ, BARGERON, David, GUPTA, Anoop, CADIZ, Jonathan J (2001): Robust Annotation Positioning in Digital Documents. In: *Proceedings of the SIGCHI conference on Human factors in computing systems* (S. 285–292). ACM

BRYNJOLFSSON, Erik, MCAFEE, Andrew (2016): *The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies*. New York, USA: W.W. Norton & Company

BUNDESMINISTERIUM FÜR BILDUNG UND FORSCHUNG (2014): Die neue Hightech-Strategie – Innovationen für Deutschland. Online verfügbar unter: URL: https://www.bmbf.de/pub_hts/HTS_Broschure_Web.pdf [05.12.2018]

BUSH, Vannevar (1945): As We May Think. *The Atlantic Monthly*, *176* (1), S. 101–108

CALDAS, Carlos H., SOIBELMAN, Lucio, HAN, Jiawei (2002): Automated Classification of Construction Project Documents. *Journal of Computing in Civil Engineering*, *16* (4), S. 234–243

CAPONI, Alessandro, DI IORIO, Angelo, VITALI, Fabio, ALBERTI, Paolo, SCATÁ, Marcello (2018): Exploiting Patterns and Templates for Technical Documentation. In: *Proceedings of the ACM Symposium on Document Engineering 2018* (S. 1–9). DocEng '18, Halifax, Canada: ACM

CARDINAELS, Kris, MEIRE, Michael, DUVAL, Erik (2005): Automating Metadata Generation: the Simple Indexing Interface. In: *Proceedings of the 14th International Conference on World Wide Web* (S. 548). WWW '05, Chiba, Japan: ACM Online verfügbar unter: URL: http://portal.acm.org/citation.cfm?doid=1060745.1060825 [22.01.2019]

CARLSON, Christopher N. (2003): Information Overload, Retrieval Strategies and Internet User Empowerment. In: Leslie HADDON (Hrsg.): *The Good, the Bad and the Irrelevant* (Bd. 1, S. 169–173). COST 269, Helsinki, Finland: UIAH

CARSTENSEN, Kai-Uwe (2010): *Computerlinguistik und Sprachtechnologie: eine Einführung*. Heidelberg: Spektrum

CASCINI, Gaetano, FANTECHI, Alessandro, SPINICCI, Emilio (2004): Natural Language Processing of Patents and Technical Documentation. In: *International Workshop on Document Analysis Systems* (S. 508–520). DAS 2004, Berlin, Heidelberg: Springer

CASTELLS, Pablo, FERNÁNDEZ SÁNCHEZ, Miriam, VALLET WEADON, David Jordi (2007): An Adaptation of the Vector-Space Model for Ontology-based Information Retrieval. *IEEE Transactions on Knowledge and Data Engineering*, *19* (2), S. 261–272

CHAO, Hui, FAN, Jian (2004): Layout and Content Extraction for PDF Documents. In: *Document Analysis Systems VI. DAS 2004* (S. 213–224). Berlin, Heidelberg: Springer

CLARKE, Charles LA, KOLLA, Maheedhar, CORMACK, Gordon V, VECHTOMOVA, Olga, ASHKAN, Azin, BÜTTCHER, Stefan, MACKINNON, Ian (2008): Novelty and Diversity in Information Retrieval Evaluation. In: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (S. 659–666). ACM

COLAS, Fabrice, PACLÍK, Pavel, KOK, Joost N., BRAZDIL, Pavel (2007): Does SVM Really Scale Up to Large Bag of Words Feature Spaces? In: Michael R. BERTHOLD, John SHAWE-TAYLOR, Nada LAVRAČ (Hrsg.): *Advances in Intelligent Data Analysis VII* (Bd. 4723, S. 296–307). Berlin, Heidelberg: Springer

CROFT, W Bruce (1986): User-specified Domain Knowledge for Document Retrieval. In: *Proceedings of the 9th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (S. 201–206). SIGIR 1986, Pisa, Italy: ACM

DAIBER, Joachim, JAKOB, Max, HOKAMP, Chris, MENDES, Pablo N. (2013): Improving Efficiency and Accuracy in Multilingual Entity Extraction. In: *Proceedings of the 9th International Conference on Semantic Systems* (S. 121–124). I-SEMANTICS 2013, New York, USA: ACM

DATTOLO, Antonina, DI IORIO, Angelo, DUCA, Silvia, FELIZIANI, Antonio Angelo, VITALI, Fabio (2007): Structural Patterns for Descriptive Documents. In: *International Conference on Web Engineering* (S. 421–426). Berlin, Heidelberg: Springer

DEERWESTER, Scott, DUMAIS, Susan T, FURNAS, George W, LANDAUER, Thomas K, HARSHMAN, Richard (1990): Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, *41* (6), S. 391–407

DÉJEAN, Hervé, MEUNIER, Jean-Luc (2006): A System for Converting PDF Documents into Structured XML Format. In: *International Workshop on Document Analysis Systems* (S. 129–140). DAS 2006, Berlin, Heidelberg: Springer

DI IORIO, Angelo, PERONI, Silvio, POGGI, Francesco, VITALI, Fabio (2012): A First Approach to the Automatic Recognition of Structural Patterns in XML Documents. In: *Proceedings of the 2012 ACM Symposium on Document Engineering* (S. 85–94). DocEng '12, New York, USA: ACM

DI IORIO, Angelo, PERONI, Silvio, POGGI, Francesco, VITALI, Fabio (2014): Dealing with Structural Patterns of XML Documents. *Journal of the Association for Information Science and Technology*, *65* (9), S. 1884–1900

DILL, Stephen, EIRON, Nadav, GIBSON, David, GRUHL, Daniel, GUHA, R, JHINGRAN, Anant, u. a. (2003): SemTag and Seeker: Bootstrapping the Semantic Web via Automated Semantic Annotation. In: *Proceedings of the 12th international conference on World Wide Web* (S. 178–186). WWW '03, Budapest, Hungary: ACM

DIMOU, Anastasia, VERBORGH, Ruben, SANDE, Miel Vander, MANNENS, Erik, VAN DE WALLE, Rik (2015): Machine-Interpretable Dataset and Service Descriptions for Heterogeneous Data Access and Retrieval. In: *Proceedings of the 11th International Conference on Semantic Systems* (S. 145–152). SEMANTICS '15, Vienna, Austria: ACM

DIN 5008 Schreib- und Gestaltungsregeln für die Textverarbeitung. (2011)

DIN 31623-1 Indexierung zur inhaltlichen Erschließung von Dokumenten; Begriffe, Grundlagen. (1988)

DIN EN 61355-1 Klassifikation und Kennzeichnung von Dokumenten für Anlagen, Systeme und Ausrüstungen - Teil 1: Regeln und Tabellen zur Klassifikation. (2009)

DORFHUBER, Jeanette, ZIEGLER, Wolfgang (2017): Content Relevance Analytics - Was lehren Delivery Portale über unseren Content und die Nutzer? In: *Tagungsband der tekom Jahrestagung 2017*. Stuttgart: tcworld

DREWER, Petra, ZIEGLER, Wolfgang (2011): *Technische Dokumentation*. (1. Aufl.). Würzburg: Vogel

DUDEN (2018): *Content (Wörterbucheintrag)*. Online verfügbar unter: URL: http://www.duden.de/rechtschreibung/Content [13.12.2018]

DUDEN (2019): *Semantik (Wörterbucheintrag)*. Online verfügbar unter: URL: https://www.duden.de/rechtschreibung/Semantik [10.01.2019]

DZBOR, Martin, MOTTA, Enrico, DOMINGUE, John (2004): Opening Up Magpie via Semantic Services. In: *Lecture Notes in Computer Science* (Bd. 3298, S. 635–649). The Semantic Web – ISWC 2004, Berlin, Heidelberg: Springer

EHRLINGER, Lisa, WÖß, Wolfram (2016): Towards a Definition of Knowledge Graphs. In: *Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems* (Bd. 1695, S. 4). SEMANTiCS 2016, Leipzig: CEUR-WS Online verfügbar unter: URL: http://ceur-ws.org/Vol-1695/paper4.pdf [22.01.2019]

ELL, Basil, HARTH, Andreas (2014): A Language-Independent Method for the Extraction of RDF Verbalization Templates. In: *Proceedings of the 8th International Natural Language Generation Conference (INLG)* (S. 26–34). Philadelphia, USA: Association for Computational Linguistics

ERTEL, Wolfgang (2016): *Grundkurs Künstliche Intelligenz: eine praxisorientierte Einführung*. (4., überarbeitete Auflage.). Wiesbaden: Springer Vieweg

EVANS, Michael P. (2007): Analysing Google Rankings Through Search Engine Optimization Data. (Steven FURNELL, Hrsg.)*Internet Research*, *17* (1), S. 21–37

FANG, Jing, TANG, Zhi, GAO, Liangcai (2011): Reflowing-Driven Paragraph Recognition for Electronic Books in PDF. In: Gady AGAM, Christian VIARD-GAUDIN (Hrsg.): *Document Recognition and Retrieval XVIII, part of the IS&T-SPIE Electronic Imaging Symposium*. SPIE 2011, San Jose, USA: SPIE

FERNÁNDEZ, Miriam, CANTADOR, Iván, LÓPEZ, Vanesa, VALLET, David, CASTELLS, Pablo, MOTTA, Enrico (2011): Semantically Enhanced Information Retrieval: An Ontology-based Approach. *Web Semantics: Science, Services and Agents on the World Wide Web*, *9* (4), S. 434–452

FORMAN, George (2004): A Pitfall and Solution in Multi-Class Feature Selection for Text Classification. In: *Proceedings of the Twenty-first International Conference on Machine Learning* (S. 38–46). ICML '04, Banff, Canada: ACM

FREITAS, André, CURRY, Edward, OLIVEIRA, Joao Gabriel, O'RIAIN, Sean (2012): Querying Heterogeneous Datasets on the Linked Data Web: Challenges, Approaches, and Trends. *IEEE Internet Computing*, *16* (1), S. 24–33

FURTH, Sebastian (2018): *Linkable Technical Documentation*. Universität Würzburg, Dissertation. Abgerufen von urn:nbn:de:bvb:20-opus-174185.

FURTH, Sebastian, BAUMEISTER, Joachim (2015): On the Semantification of 5-Star Technical Documentation. In: *Proceedings of the LWA 2015 Workshops: KDML, FGWM, IR, and FGDB* (S. 264–271). LWA 2015, Trier: CEUR-WS Online verfügbar unter: URL: http://ceur-ws.org/Vol-1458/F08_CRC52_Furth.pdf [04.02.2019]

GAMMA, Erich, HELM, Richard, JOHNSON, Ralph, VLISSIDES, John (1995): *Design Patterns: Elements of Reusable Object-oriented Software*. Boston, USA: Addison-Wesley

GHTF/SG1/N70 Label and Instructions for Use for Medical Devices. (2011) Online verfügbar unter: URL: http://www.imdrf.org/docs/ghtf/final/sg1/technical-docs/ghtf-sg1-n70-2011-label-instruction-use-medical-devices-110916.pdf [11.01.2019]

GOLUB, Koraljka (2006): Automated Subject Classification of Textual Web Documents. *Journal of Documentation*, *62* (3), S. 350–371

GRAHLMANN, Knut, HILHORST, Cocky, VAN AMERONGEN, Sander, HELMS, Remko, BRINKKEMPER, Sjaak (2010): Impacts of Implementing Enterprise Content Management Systems. In: *Proceedings of the 18th European Conference on Information Systems* (S. 103). ECIS 2010, Pretoria, South Africa

GRUBER, Thomas R. (1993): A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, *5* (2), S. 199–220

GUHA, R. V., BRICKLEY, Dan, MACBETH, Steve (2016): Schema.org: Evolution of Structured Data on the Web. *Communications of the ACM*, *59* (2), S. 44–51

HARARI, Yuval Noaḥ (2018): *Homo Deus: eine Geschichte von Morgen*. (Andreas WIRTHENSOHN, Übers.) (1. Auflage.). München: C.H.Beck

HASLHOFER, Bernhard, SANDERSON, Robert, SIMON, Rainer, VAN DE SOMPEL, Herbert (2012): Open Annotations on Multimedia Web Resources. *Multimedia Tools and Applications*, *70* (2), S. 847–867

HASSAN, Tamir (2018): Towards a Universally Editable Portable Document Format. In: *Proceedings of the ACM Symposium on Document Engineering* (S. 11:1–11:4). DocEng '18, Halifax, Canada: ACM

HAYNES, David (2004): *Metadata for Information Management and Retrieval*. London, UK: Facet

HEIDEL, Roland, HOFFMEISTER, Michael, HANKEL, Martin, DÖBRICH, Udo (2017): *Industrie 4.0 Basiswissen RAMI 4.0: Referenzarchitekturmodell mit Industrie 4.0-Komponente*. (DEUTSCHES INSTITUT FÜR NORMUNG, Hrsg.) (1. Auflage.). Berlin, Wien, Zürich: Beuth

HENNIG, Jörg, TJARKS-SOBHANI, Marita (Hrsg.) (2017): *Intelligente Information*. Stuttgart: tcworld

HEPP, Martin, LEUKEL, Joerg, SCHMITZ, Volker (2007): A Quantitative Analysis of Product Categorization Standards: Content, Coverage, and Maintenance of eCl@ss, UNSPSC, eOTD, and the RosettaNet Technical Dictionary. *Knowledge and Information Systems*, *13* (1), S. 77–114

HJØRLAND, Birger (2013): Facet Analysis: The Logical Approach to Knowledge Organization. *Information Processing & Management*, *49* (2), S. 545–557

HOSNY, Mahmoud M., EL-BELTAGY, Samhaa R., ALLAM, Mahmoud E. (2015): Unsupervised Data Driven Taxonomy Learning. In: *Proceedings of the First International Conference on Arabic Computational Linguistics* (S. 9–14). ACLing 2015, Cairo, Egypt: IEEE

HYVÖNEN, Eero, MÄKELÄ, Eetu (2006): Semantic Autocompletion. In: *The Semantic Web – ASWC 2006*. *Lecture Notes in Computer Science* (Bd. 4185, S. 739–751). ASWC 2006, Berlin, Heidelberg: Springer

IEC 82079-1 Preparation of Instructions for Use - Structuring, Content and Presentation. (2012)

IIRDS CONSORTIUM (2018): Implementation Guide - Generating VDI 2770 from iiRDS 1.0 (First Draft). self-published

ISO 704 Terminology Work - Principles and Methods. (2009)

ISO 9001 Quality Management Systems - Requirements. (2008)

ISO 15836-1 Information and Documentation -- The Dublin Core Metadata Element Set -- Part 1: Core Elements. (2017)

ISO 25964-1 Information and Documentation -- Thesauri and Interoperability with other Vocabularies -- Part 1: Thesauri for Information Retrieval. (2011)

ISO 26162 Systems to Manage Terminology, Knowledge and Content - Design, Implementation and Maintenance of Terminology Management Systems. (2012)

KACHKACH, Ahmed (2018): Problem-solving with ML: Automatic Document Classification. *Google Cloud Blog* Online verfügbar unter: URL: https://cloud.google.com/blog/products/gcp/problem-solving-with-ml-automatic-document-classification [29.01.2019]

KILLORAN, John B. (2013): How to Use Search Engine Optimization Techniques to Increase Website Visibility. *IEEE Transactions on Professional Communication*, *56* (1), S. 50–66

KINCAID, J Peter, KINCAID, Calliopi D, KNIFFIN, JD, THOMAS, Margaret, LANG, Sheau (1991): Intelligent Authoring Aids for Technical Instructional Materials Written in Controlled English. *Journal of Artificial Intelligence in Education*, *2* (3), S. 77

KIRYAKOV, Atanas, POPOV, Borislav, TERZIEV, Ivan, MANOV, Dimitar, OGNYANOFF, Damyan (2004): Semantic Annotation, Indexing, and Retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web*, *2* (1), S. 49–79

KO, Youngjoong (2012): A Study of Term Weighting Schemes Using Class Information for Text Classification. In: *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '12, Portland, USA: ACM

KOHAVI, Ron (1995): A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In: *Proceedings of the 14thInternational Joint Conference on Artificial Intelligence* (Bd. 2, S. 1137–1145). IJCAI'95, Montreal, Canada: Morgan Kaufmann

KOTSAKIS, Evangelos (2002): Structured Information Retrieval in XML Documents. In: *Proceedings of the 2002 ACM Symposium on Applied Computing* (S. 663–667). SAC'2002), New York, USA: ACM

KRÜGER, Manfred, ZIEGLER, Wolfgang (2008): Standards für strukturierte technische Informationen - ein Überblick. In: Jürgen MUTHIG (Hrsg.): *Standardisierungsmethoden für die technische Dokumentation* (S. 11–40). Lübeck: Schmidt-Römhild

KUHLTHAU, Carol Collier (2005): Accommodating the User's Information Search Process: Challenges for Information Retrieval System Designers. *Bulletin of the American Society for Information Science and Technology*, *25* (3), S. 12–16

LAHTINEN, Timo (2000): *Automatic Indexing: an Approach Using an Index Term Corpus and Combining Linguistic and Statistical Methods*. Universität Helsinki, Dissertation. Online verfügbar unter: URL: http://ethesis.helsinki.fi/julkaisut/hum/yleis/vk/lahtinen/ [28.01.2019]

LALMAS, Mounia (2009): XML Information Retrieval. In: Marcia J. BATES, Mary Niles MAACK (Hrsg.): *Encyclopedia of Library and Information Sciences* (3rd Edition.)

LAN, Man, TAN, Chew-Lim, LOW, Hwee-Boon, SUNG, Sung (2005): A Comprehensive Comparative Study on Term Weighting Schemes for Text Categorization with Support Vector Machines. In: *14th International World Wide Web Conference* (S. 1032–1033). WWW '05, Chiba, Japan: ACM

LE, Quoc, MIKOLOV, Tomas (2014): Distributed Representations of Sentences and Documents. In: *Proceedings of the 31st International Conference on International Conference on Machine Learning* (Bd. 32, S. II-1188-II–1196). ICML'14, Beijing, China: ACM

LEASE, Matthew (2018): Fact Checking and Information Retrieval. In: *Proceedings of the First Biennial Conference on Design of Experimental Search & Information Retrieval Systems* (Bd. 2167). DESIRES 2018, Bertinoro, Italy: CEUR-WS Online verfügbar unter: URL: http://ceur-ws.org/Vol-2167/short1.pdf [04.02.2019]

LEMNITZER, Lothar, ZINSMEISTER, Heike (2006): *Korpuslinguistik: eine Einführung*. Tübingen: Narr

LEY, Martin (2018): Informationen erhalten Bedeutung. *technische kommunikation*, *40* (08/2018), S. 50–55

LEZIUS, Wolfgang (2000): Morphy – German Morphology, Part-of-Speech Tagging and Applications. In: *Proceedings of the 9th EURALEX International Congress* (S. 619–623). EURALEX 2000, Stuttgart: Institut für Maschinelle Sprachverarbeitung

LI, Yuhua, MCLEAN, David, BANDAR, Zuhair A., O'SHEA, James D., CROCKETT, Keeley (2006): Sentence Similarity Based on Semantic Nets and Corpus Statistics. *IEEE Transactions on Knowledge and Data Engineering, 18* (8), S. 1138–1150

LIU, Mingyong, YANG, Jiangang (2012): An Improvement of TFIDF Weighting in Text Categorization. In: *International Conference on Computer Technology and Science (ICCTS 2012)*. ICCTS 2012, New Delhi, India

MANNING, Christopher D., RAGHAVAN, Prabhakar, SCHÜTZE, Hinrich (2008): *Introduction to Information Retrieval*. New York, USA: Cambridge Univ. Press

MANNING, Christopher D., SCHÜTZE, Hinrich (1999): *Foundations of Statistical Natural Language Processing*. Cambridge, USA: MIT Press

MASCARDI, Viviana, CORDÌ, Valentina, ROSSO, Paolo (2007): A Comparison of Upper Ontologies. In: *Dagli Oggetti agli Agenti. 8th AI*IA/TABOO Joint Workshop „From Objects to Agents": Agents and Industry: Technological Applications of Software Agents* (Bd. 2007, S. 55–64). WOA 2007, Genova, Italy

MCDONALD, Ryan, CRAMMER, Koby, PEREIRA, Fernando (2005): Flexible Text Segmentation with Structured Multilabel Classification. In: *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing* (S. 987–994). HLT '05, Vancouver, Canada: Association for Computational Linguistics

MENYCHTAS, Andreas, KONSTANTELI, Kleopatra G. (2012): Fault Detection and Recovery Mechanisms and Techniques for Service Oriented Infrastructures. In: Dimosthenis KYRIAZIS, Theodora VARVARIGOU, Konstanteli KLEOPATRA G. (Hrsg.): *Achieving Real-Time in Distributed Computing: From Grids to Clouds*. Hershey, USA: IGI Global

MERZ, Tobias (2019): Linguistisches SEO – Textverständlichkeit in der Suchmaschinenoptimierung. *congree Blog*. Online verfügbar unter: URL: https://www.congree.com/blog/linguistisches-seo/ [28.01.2019]

METZLER, Donald, DUMAIS, Susan, MEEK, Christopher (2007): Similarity Measures for Short Segments of Text. In: *European Conference on Information Retrieval. Lecture Notes in Computer Science* (Bd. 4425, S. 16–27). ECIR 2007, Berlin, Heidelberg: Springer

MEUNIER, Jean-Luc (2010): *Physical and Logical Strucutre Recognition of PDF Documents*. University of Fribourg (Switzerland), Dissertation. Online verfügbar unter: URL: www.bloechle.ch/jean-luc/pub/Bloechle_Thesis.pdf [01.02.2019]

MIHALCEA, Rada, CORLEY, Courtney, STRAPPARAVA, Carlo (2006): Corpus-based and Knowledge-based Measures of Text Semantic Similarity. In: *Proceedings of the 21st National Conference on Artificial Intelligence* (Bd. Vol. 1, S. 775–780). AAAI'06, Boston, USA: AAAI Press

MUTHIG, Jürgen (Hrsg.) (2008): *Standardisierungsmethoden für die technische Dokumentation*. Lübeck: Schmidt-Römhild

NAZEMI, Azadeh, MURRAY, Iain, MCMEEKIN, David A (2014): Practical Segmentation Methods for Logical and Geometric Layout Analysis to Improve Scanned PDF Accessibility to Vision Impaired. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, *7* (4), S. 23–26

OASIS The DocBook Schema, Committee Draft 5.0. (2008) Online verfügbar unter: URL: http://www.docbook.org/specs/docbook-5.0-spec-cd-03.html [01.02.2019]

OASIS DITA Version 1.2 Specification. (2010) Online verfügbar unter: URL: http://docs.oasis-open.org/dita/v1.2/spec/DITA1.2-spec.html [01.02.2019]

OBERLE, Claudia, ZIEGLER, Wolfgang (2012): Content Intelligence for Content Management Systems. *tcworld e-magazine*, (12) Online verfügbar unter: URL: http://www.tcworld.info/rss/article/content-intelligence-for-content-management-systems/ [18.01.2019]

OBERLE, Claudia, ZIEGLER, Wolfgang (2013): Bestimmung von CMS-Kennzahlen mit der REx-Methode. In: *Tagungsband der tekom Frühjahrstagung 2013* (S. 72–73). Stuttgart: tcworld

OEVERMANN, Jan (2016a): Intelligente Klassifizierung von technischen Inhalten – Automatisierung und Anwendungspotenziale. In: *Tagungsband der tekom Jahrestagung 2016*. Stuttgart: tcworld

OEVERMANN, Jan (2016b): Reconstructing Semantic Structures in Technical Documentation with Vector Space Classification. In: *Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems* (Bd. 1695). SEMANTiCS 2016, Leipzig: CEUR-WS Online verfügbar unter: URL: http://ceur-ws.org/Vol-1695/paper5r1.pdf [07.01.2019]

OEVERMANN, Jan (2017a): Künstliche Intelligenz und intelligente Textservices. In: Jörg HENNIG, Marita TJARKS-SOBHANI (Hrsg.): *Intelligente Information* (S. 82–94). Stuttgart: tcworld

OEVERMANN, Jan (2017b): iiRDS voll automatisch – mit Machine Learning zum intelligenten Austauschformat. In: *Tagungsband der tekom Jahrestagung 2017*. Stuttgart: tcworld

OEVERMANN, Jan (2018a): Technical Documentation 4.0 - How the Latest Trends in Manufacturing Lead to New Paradigms for Technical Writers. *tcworld*, (February 2018), S. 12–15

OEVERMANN, Jan (2018b): Semantic PDF Segmentation for Legacy Documents in Technical Documentation. In: *Procedia Computer Science. Proceedings of the 14th International Conference on Semantic Systems* (Bd. 137, S. 55–65). SEMANTiCS 2018, Vienna, Austria: Elsevier

OEVERMANN, Jan, FLESCHUTZ-BALAREZO, Timo (2018): Großputz im CMS! Semantische Ähnlichkeitsanalyse für XML-Module. In: *Tagungsband der tekom Jahrestagung 2018*. Stuttgart: tcworld

OEVERMANN, Jan, LÜTH, Christoph (2018): Semantically Weighted Similarity Analysis for XML-based Content Components. In: *Proceedings of the ACM Symposium on Document Engineering 2018* (S. 1–4). DocEng '18, Halifax, Canada: ACM

OEVERMANN, Jan, ZIEGLER, Wolfgang (2016): Automated Intrinsic Text Classification for Component Content Management Applications in Technical Communication. In: *Proceedings of the 2016 ACM Symposium on Document Engineering* (S. 95–98). DocEng '16, Vienna, Austria: ACM

OEVERMANN, Jan, ZIEGLER, Wolfgang (2018): Automated Classification of Content Components in Technical Communication. *Computational Intelligence*, *34* (1), S. 30–48

PAAß, Gerhard, KONYA, Iuliu (2011): Machine Learning for Document Structure Recognition. In: Alexander MEHLER, Kai-Uwe KÜHNBERGER, Henning LOBIN, Harald LÜNGEN, Angelika STORRER, Andreas WITT (Hrsg.): *Modeling, Learning, and Processing of Text Technological Data Structures* (Bd. 370, S. 221–247). Berlin, Heidelberg: Springer

PARSON, Ulrike, SAPARA, Jürgen, ZIEGLER, Wolfgang (2017): iiRDS for Technical Writers - Introduction to the Metadata. In: *Tagungsband der tekom Jahrestagung 2017*. Stuttgart: tcworld

PEREZ-BELTRACHINI, Laura, SAYED, Rania Mohamed, GARDENT, Claire (2016): Building RDF Content for Data-to-Text Generation. In: *Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers* (S. 1493–1502). COLING 2016, Osaka, Japan

PODDIG, Thorsten (1992): *Künstliche Intelligenz und Entscheidungstheorie*. Wiesbaden: Deutscher Universitätsverlag

PUPPE, Frank (2013): *Einführung in Expertensysteme*. (2., illustriert Aufl.). Berlin, Heidelberg: Springer

RAVICHANDRAN, Deepak, HOVY, Eduard (2002): Learning Surface Text Patterns for a Question Answering System. In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (S. 41–47). ACL '02, Philadelphia, USA: Association for Computational Linguistics

REICHENBERGER, Klaus (2010): Grundlagen semantischer Netze. In: *Kompendium semantische Netze* (S. 3–19). Berlin, Heidelberg: Springer

REUßNER, Lukas (2018): *Classification of Technical Documentation*. Hochschule München, Bachelor-Thesis. Studiengang Technische Kommunikation.

RING, Martin, LÜTH, Christoph, GLÄBE, Ralf (2009): FactBook 2.0. [nicht mehr erreichbar] Online verfügbar unter: URL: http://factbook.informatik.uni-bremen.de/

ROCKLEY, Ann, COOPER, Charles (2012): *Managing Enterprise Content: A Unified Content Strategy*. (2nd ed.). Berkeley, USA: New Riders

ROCKLEY, Ann, KOSTUR, Pamela, MANNING, Steve (2003): *Managing Enterprise Content: A Unified Content Strategy*. Berkley, USA: New Riders

ROGERS, Steven K, KABRISKY, Matthew, BAUER, KENNETH, OXLEY, Mark (2003): Computing Machinery and Intelligence Amplification. *Computational Intelligence: The Experts Speak*, *3*, S. 25–44

S1000D S1000D Issue 4.2 – International Specification for Technical Publications Using a Common Source Database. (2017) Online verfügbar unter: URL: http://public.s1000d.org/Downloads/ [01.02.2019]

SCHAFFNER, Michael (2017): Industrie 4.0 als Motor für „intelligente Information". In: Jörg HENNIG, Marita TJARKS-SOBHANI (Hrsg.): *Intelligente Information* (S. 111–124). Stuttgart: tcworld

SCHAFFNER, Michael (2019): Industrie 4.0: Technische Redakteure werden zu Semantikmodellierern: Digitalisierung verändert die Arbeitswelt in der Technischen Kommunikation. In: Burghard HERMEIER, Thomas HEUPEL, Sabine FICHTNER-ROSADA (Hrsg.): *Arbeitswelten der Zukunft* (S. 107–129). Wiesbaden: Springer Fachmedien

SCHWEITZER, Eric, AURICH, Jan C. (2010): Continuous Improvement of Industrial Product-Service Systems. *CIRP Journal of Manufacturing Science and Technology*, *3* (2), S. 158–164

SEBASTIANI, Fabrizio (2002): Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, *34* (1), S. 1–47

SEELING, Christian, BECKS, Andreas (2005): Semantische Interaktionspfade: Ein Multiview System für betriebswirtschaftliche Entscheidungsprobleme. In: *Beiträge der 35. Jahrestagung der Gesellschaft für Informatik e.V. (GI)* (Bd. 2, S. 99–103). INFORMATIK 2005 - Informatik LIVE!, Bonn

SHAH, Urvi, FININ, Tim, JOSHI, Anupam, COST, R Scott, MATFIELD, James (2002): Information Retrieval on the Semantic Web. In: *Proceedings of the Eleventh International Conference on Information and Knowledge Management* (S. 461–468). CIKM'02, McLean, USA: ACM

SHETH, Amit, ARPINAR, I Budak, KASHYAP, Vipul (2004): Relationships at the Heart of Semantic Web: Modeling, Discovering, and Exploiting Complex Semantic Relationships. In: *Enhancing the Power of the Internet* (S. 63–94). Berlin, Heidelberg: Springer

SINGHAL, Amit (2001): Modern Information Retrieval: A Brief Overview. *IEEE Data Engineering Bulletin*, *24* (4), S. 35–43

SINGHAL, Amit (2012): Introducing the Knowledge Graph: Things, not Strings. *Official Google Blog* Online verfügbar unter: URL: https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html [22.01.2019]

SOFEAN, Mustafa, ARAS, Hidir, ALRIFAI, Ahmad (2018): A Workflow-Based Large-Scale Patent Mining and Analytics Framework. In: Robertas DAMAŠEVIČIUS, Giedrė VASILJEVIENĖ (Hrsg.): *Proceedings of the International Conference on Information and Software Technologies* (Bd. 920, S. 210–223). ICIST 2018, Berlin, Heidelberg: Springer

SOKOLOVA, Marina, LAPALME, Guy (2009): A Systematic Analysis of Performance Measures for Classification Tasks. *Information Processing & Management*, *45* (4), S. 427–437

SOTO, Axel J., MOHAMMAD, Abidalrahman, ALBERT, Andrew, ISLAM, Aminul, MILIOS, Evangelos, DOYLE, Michael, u. a. (2015): Similarity-based Support for Text Reuse in Technical Writing. In: *Proceedings of the 2015 ACM Symposium on Document Engineering* (S. 97–106). DocEng'15, Lausanne, Switzerland: ACM

STAAB, Steffen, STUDER, Rudi (Hrsg.) (2009): *Handbook on Ontologies*. (2. Aufl.). Berlin, Heidelberg: Springer

STEURER, Stephan (2017): Dynamische Information und ihre Bereitstellung. In: Jörg HENNIG, Marita TJARKS-SOBHANI (Hrsg.): *Intelligente Information* (S. 125–133). Stuttgart: tcworld

STRAUB, Daniela (2016): *Branchenkennzahlen für die Technische Dokumentation 2016*. Stuttgart: tcworld Online verfügbar unter: URL: https://www.tekom.de/fileadmin/Dokumente/de/tekom_2016-07-01_Branchenkennzahlen_2016_DE.pdf [01.02.2019]

STVILIA, Besiki, GASSER, Les, TWIDALE, Michael B., SMITH, Linda C. (2007): A Framework for Information Quality Assessment. *Journal of the American Society for Information Science and Technology*, *58* (12), S. 1720–1733

TEKOM E.V. (2017a): iiRDS Specification - intelligent information Request and Delivery Standard - First Public Working Draft, 03 April 2017. [Obsolet]

TEKOM E.V. (2017b): iiRDS RDF Schema - First Public Working Draft, 03 April 2017. [Obsolet]

UREN, Victoria, CIMIANO, Philipp, IRIA, José, HANDSCHUH, Siegfried, VARGAS-VERA, Maria, MOTTA, Enrico, CIRAVEGNA, Fabio (2006): Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art. *Journal of Web Semantics*, *4* (1), S. 14–28

VDI 2770 Richtlinie VDI 2770 - Blatt 1 „Betrieb verfahrenstechnischer Anlagen; Mindestanforderungen an digitale Herstellerinformationen für die Prozessindustrie; Grundlagen" (Technische Regel Entwurf). (2018)

VERBORGH, Ruben (2019): Re-decentralizing the Web, for Good This Time. In: Oshani SENEVIRATNE, James HENDLER (Hrsg.): *Linking the World's Information: Tim Berners-Lee's Invention of the World Wide Web*. ACM. Accepted for publication. Online verfügbar unter: URL: https://ruben.verborgh.org/articles/redecentralizing-the-web/ [01.02.2019]

VON LEPEL, Florian, FINKLER, Stephan (2010): Semantic Architecture for Managing Information Through Structured Storage and Retrieval. US Patent Nr. US 7,725.499 B1 (STAR AG).

W3C SPARQL 1.1 Query Language - W3C Recommendation 21 March 2013. (2013) Online verfügbar unter: URL: https://www.w3.org/TR/2013/REC-sparql11-query-20130321/ [21.01.2019]

W3C RDF 1.1 Primer – W3C Working Group Note 25 February 2014. (2014) Online verfügbar unter: URL: http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/ [01.02.2019]

W3C RDF 1.1 N-Quads – A line-based syntax for RDF datasets – W3C Recommendation 25 February 2014. (2014) Online verfügbar unter: URL: https://www.w3.org/TR/2014/REC-n-quads-20140225/ [08.02.2019]

W3C JSON-LD 1.0 -- A JSON-based Serialization for Linked Data -- W3C Recommendation 16 January 2014. (2014) Online verfügbar unter: URL: https://www.w3.org/TR/2014/REC-json-ld-20140116/

W3C Linked Data Platform 1.0 – W3C Recommendation 26 February 2015. (2015) Online verfügbar unter: URL: https://www.w3.org/TR/2015/REC-ldp-20150226/ [01.02.2019]

W3C Web Annotation Vocabulary – W3C Recommendation 23 February 2017. (2017) Online verfügbar unter: URL: https://www.w3.org/TR/2017/REC-annotation-vocab-20170223/ [01.02.2019]

W3C Web Annotation Data Model – W3C Recommendation 23 February 2017. (2017) Online verfügbar unter: URL: https://www.w3.org/TR/2017/REC-annotation-model-20170223/ [01.02.2019]

WACHE, Holger, VÖGELE, Thomas J., VISSER, Ubbo, STUCKENSCHMIDT, Heiner, SCHUSTER, Gerhard, NEUMANN, H., HÜBNER, Sebastian (2001): Ontology-based Integration of Information - A Survey of Existing Approaches. In: *Workshop on Ontologies and Information Sharing* (S. 108–117). IJCAI'01, Seattle, USA

WADHWA, Neelanshi, PATERIYA, Rajesh Kumar, SHRIVASTAVA, Sonika (2019): A Hybrid Query Recommendation Technique in Information Retrieval. In: Pradeep Kumar SINGH, Marcin PAPRZYCKI, Bharat BHARGAVA, Jitender Kumar CHHABRA, Narottam Chand KAUSHAL, Yugal KUMAR (Hrsg.): *Futuristic Trends in Network and Communication Technologies* (Bd. 958, S. 165–175). FTNCT 2018, Solan, India: Springer

WELLER, Kathrin (2009): Ontologien: Stand und Entwicklung der Semantik für WorldWideWeb. *LIBREAS. Library Ideas # 15: A Semiotic Turn? & Open Access und Geisteswissenschaften*, S. 5–12

YAMAUCHI, Yutaka, WHALEN, Jack, BOBROW, Daniel G. (2003): Information Use of Service Technicians in Difficult Cases. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (S. 81–88). CHI '06, Montreal, Canada: ACM

ZENG, Qing, KOGAN, Sandra, ASH, Nachman, GREENES, Robert A, BOXWALA, Aziz A (2002): Characteristics of Consumer Terminology for Health Information Retrieval. *Methods of Information in Medicine*, *41* (04), S. 289–298

ZHENG, Bweijunl, ZHANG, Wei, FENG, Xiaoyu Fu Boqin (2013): A Survey of Faceted Search. *Journal of Web Engineering, 12* (1&2), S. 041–064

ZIEGLER, Wolfgang (2013a): Content-Management in der Technischen Kommunikation. Ein Überblick. In: Jörg HENNIG, Marita TJARKS-SOBHANI (Hrsg.): *Content Management und Technische Kommunikation* (S. 11–25). Stuttgart: tcworld

ZIEGLER, Wolfgang (2013b): Alles muss raus! Content Delivery für Informationsportale. In: *Tagungsband der tekom Jahrestagung 2013* (S. 47–48). Wiesbaden: tcworld

ZIEGLER, Wolfgang (2015): Content Management und Content Delivery. Powered by PI-Class. In: *Tagungsband der tekom Jahrestagung 2015*. Stuttgart: tcworld

ZIEGLER, Wolfgang (2016): Drivers and Concepts of Content Management Systems in the Age of Globalization and Mass Customization. *Frontier, Official Journal of Japan Technical Communicators Association JTCA*, S. 15–26

ZIEGLER, Wolfgang (2017): Metadaten für intelligenten Content. In: Jörg HENNIG, Marita TJARKS-SOBHANI (Hrsg.): *Intelligente Information* (S. 51–66). Stuttgart: tcworld

ZIEGLER, Wolfgang (2018a): Man muss auch austeilen können. *technische kommunikation*, *40. Jg* (04), S. 15–21

ZIEGLER, Wolfgang (2018b): Content Delivery and Content Analytics for Industrial Applications - The Upcoming Era of Digital Information Services. *Frontier, Official Journal of Japan Technical Communicators Association JTCA*

ZIEGLER, Wolfgang, BEIER, Heiko (2014): Alles muss raus. *technische kommunikation*, *40. Jg.* (6), S. 50–55

ZIEGLER, Wolfgang, BEIER, Heiko (2015): Content Delivery Portals: The Future of Modular Content. *tcworld e-magazine*, (02)

ZIEGLER, Wolfgang, STEURER, Stephan (2010): Mit PI-Mod dokumentieren - Standardisiertes Informationsmodell für den Anlagen- und Maschinenbau. *technische kommunikation*, *32. Jg.* (6), S. 51–55

ZVEI Industrie 4.0: Das Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0). (2016) Online verfügbar unter: URL: https://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/zvei-faktenblatt-rami.html [10.02.2019]

Anmerkung: Alle Hyperlinks auf Internetquellen werden in nachgestellten eckigen Klammern mit dem Datum des letzten Besuchs gekennzeichnet

7 Anhang

7.1 Übersicht Quellcode[102] und Demos

Automatisierte Klassifizierung

Quellcode: https://github.com/j-oe/coin-demo
Demo: http://coin.fastclass.de/

Automatisierte Segmentierung

Quellcode: https://github.com/j-oe/segments
Demo: http://segments.fastclass.de/

Semantische Ähnlichkeitsanalyse

Quellcode: https://github.com/j-oe/semsim
Demo: http://semsim.fastclass.de/

7.2 Codebeispielverzeichnis

Codebeispiel 1: (a) – Erste Formulierungsvariante mit 110 V (Ausschnitt). 52
Codebeispiel 2: (b) – Erste Formulierungsvariante mit 220 V (Ausschnitt). 52
Codebeispiel 3: (c) – Zweite Formulierungsvariante mit 110 V (Ausschnitt). 53
Codebeispiel 4: (d) – Zweite Formulierungsvariante mit 220 V (Ausschnitt). 53
Codebeispiel 5: Semantische Annotation eines Abschnitts in iiRDS. 98
Codebeispiel 6: SPARQL-Abfrage mit Annotationsfilter. 127

7.3 Tabellenverzeichnis

Tabelle 1: CoSMOS-Ebene Content 33
Tabelle 2: CoSMOS-Ebene Struktur 35
Tabelle 3: CoSMOS-Ebene Metadaten 38
Tabelle 4: CoSMOS-Ebene Ontologien und semantische Netze 40
Tabelle 5: Tests zur Effizienz der Ähnlichkeitsanalyse. 55
Tabelle 6: Datensätze für Trainings- und Testdaten (autom. Klassifizierung). 64
Tabelle 7: Genauigkeit für verschiedene n-Gramme als Merkmale 72
Tabelle 8: Genauigkeit für verschiedene Gewichtungsmethoden. 74
Tabelle 9: Gesamtergebnisse der automatisierten Klassifizierung. 78
Tabelle 10: Validierung der Methode zur Konfidenzmessung. 80
Tabelle 11: Baseline für Segmentierung. 100
Tabelle 12: Ergebnisse der automatisierten Segmentierung. 101
Tabelle 13: Ergebnisse für verschiedene Abfragearten. 131

[102] Der Quellcode wird jeweils unter der MIT-Lizenz bereitgestellt.

7.4 Abbildungsverzeichnis

Abbildung 1: PI-Klassifikationsraum nach Ziegler 19
Abbildung 2: Spannungsfeld des semantisch Informationszugriffs. 25
Abbildung 3: Ausschnitt Screenshot, sog. Rich Snippet. 26
Abbildung 4: Machine Learning als Teilgebiet der Künstlichen Intelligenz. 27
Abbildung 5: Arten des möglichen Informationszugriffs (CoSMOS-Ebenen). 30
Abbildung 6: Empfohlene Optimierungen im CoSMOS-Modell. 41
Abbildung 7: Standardisierungsgrade der TD nach Ziegler. 43
Abbildung 8: CoSMOS-Modell mit abgeleiteten Reifegraden von TD 43
Abbildung 9: 5-STAR Technical Documentation nach Furth. 44
Abbildung 10: Zuordnung der Untersuchungen zu CoSMOS-Ebenen. 45
Abbildung 11: Schematischer Datenfluss bei der Ähnlichkeitsanalyse. 50
Abbildung 12: Screenshot der Benutzeroberfläche der Ähnlichkeitsanalyse. 56
Abbildung 13: Versuchsaufbau der automatisierten Klassifizierung von Daten. 66
Abbildung 14: Visualisierung der Konfidenz-Hypothese. 76
Abbildung 15: Visualisierung der Chunking-Hypothese. 89
Abbildung 16: Schematischer Datenfluss im Segmentierungsprozess. 91
Abbildung 17: Chunking-Methodologie. 93
Abbildung 18: Beispiel für Konfidenz-Hypothese. 93
Abbildung 19: Beispiel- Segmentierung mit iiRDS-Annotationen. 95
Abbildung 20: Beispiel für Konfidenzkurve mit Algorithmus. 97
Abbildung 21: Visualisierung der Segmentierungsergebnisse. 102
Abbildung 22: Beispiel für Segment- vs. Konfidenzverhalten. 105
Abbildung 23: Segment-Ergebnisse eine Test-PDFs (Set A). 106
Abbildung 24: Segment-Ergebnisse eine Test-PDFs (Set B). 107
Abbildung 25: Kategorisierte Informationsquellen für Servicetechniker. 110
Abbildung 26: Vorgehensweise für unterschiedliche Informationsquellen. 111
Abbildung 27: Modell des integrierten Informationsframeworks. 112
Abbildung 28: Annotationsmodell der semantischen Autovervollständigung. 116
Abbildung 29: Annotationsmodell der automatisierten Klassifizierung. 121
Abbildung 30: Annotationsmodell der automatisierten Segmentierung. 122
Abbildung 31: Einheitlicher semantischer Zugriff über SPARQL. 126
Abbildung 32: Screenshot. Oberfläche für die automatisierte Segmentierung. 128

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich

1. die Arbeit ohne unerlaubte fremde Hilfe angefertigt habe,
2. keine anderen als die von mir angegebenen Quellen und Hilfsmittel benutzt habe

und

3. die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Karlsruhe, den 25. Juni 2019

_____

Jan Oevermann

Hinweis zu Fremdanteilen

Bei den in dieser Arbeit wiedergegeben Publikationen wurden folgende Teile in der Originalversion von anderen Autoren verfasst:

- OEVERMANN & LÜTH (2018) wiedergegeben in Kapitel 4.2:
  Teile des Abschnitts 4.2.3 und ergänzende Anmerkungen
  (Gesamt ca. 5-10 %)
- OEVERMANN & ZIEGLER (2018) wiedergegeben in Kapitel 4.3:
  Abschnitt 4.3.5.2 und ergänzende Anmerkungen
  (Gesamt ca. 5-10 %)
- BADER & OEVERMANN (2017) wiedergegeben in Kapitel 4.5:
  Abschnitte 4.5.4, 4.5.9.1 und 4.5.9.3; Teile der Abschnitte 4.5.1, 4.5.2, 4.5.3, 4.5.7, 4.5.10 und 4.5.11 und ergänzende Anmerkungen
  (Gesamt ca. 50 %)

Karlsruhe, den 25. Juni 2019

_____

Jan Oevermann