

Semantische Interoperabilität digitaler Zwillinge als Basis für  
automatisierte Erkundungsmechanismen

**Dissertation**

zur Erlangung des akademischen Grades

Doktoringenieur

(Dr.-Ing.)

von **M.Eng, Maximilian Both**

geb. am 14.02.1993 in Köln

genehmigt durch die Fakultät für Elektrotechnik und Informationstechnik  
der Otto-von-Guericke-Universität Magdeburg

Gutachter:

Prof. Dr.-Ing. Christian Diedrich

Prof. Dr.-Ing. Tobias Kleinert

Prof. Dr. rer. nat. Jochen Müller

Promotionskolloquium am 28. Juni 2024

## Kurzfassung

Eines der Ziele von Industrie 4.0 sind dynamische, flexible und weltweit vernetzte Wertschöpfungsnetzwerke. Um dies zu erreichen sind Systeme erforderlich, die autonom miteinander interagieren. Basis ist die Interoperabilität der digitalen Zwillinge der verschiedenen Assets dieser Systeme. Hierzu müssen die einzelnen Zwillinge die untereinander ausgetauschten Informationen verstehen können - sie müssen semantisch interoperabel sein. Die Erreichung von semantischer Interoperabilität stellt derzeit eine große Herausforderung im Bereich der Industrie 4.0 dar. Verschiedene Unternehmen und Branchen verwenden heterogene Semantik in ihren Vokabularen, was die Realisierung von semantischer Interoperabilität erschwert. Um semantische Interoperabilität zu erreichen, müssen die verwendeten Vokabulare aufeinander abgebildet werden.

In dieser Arbeit wird ein Konzept zur Erreichung semantischer Interoperabilität digitaler Zwillinge entwickelt. Es werden sowohl homogene als auch heterogene Sprachräume berücksichtigt. Für den homogenen Bereich wird ein lebenszyklusorientiertes Vorgehensmodell zur einheitlichen Ableitung von Industrie 4.0 Teilmodellen entworfen. Auf Basis dieses Modells werden Teilmodelle für Pumpen abgeleitet. Für den heterogenen Bereich wird eine Methode entwickelt, um die unterschiedlichen Beschreibungen der Vokabulare von digitalen Zwillingen automatisiert auf einen Standard abzubilden. Language-Models und Sentence-Embeddings werden eingesetzt, um die Elemente der Vokabulare der digitalen Zwillinge automatisiert mit denen anderer Kommunikationsteilnehmer abzugleichen, aufeinander abzubilden und so semantische Interoperabilität zu erreichen.

Um Bestandsanlagen in Industrie 4.0-Netzwerke zu integrieren und trotz heterogener Semantik ihre Informationen zu erkunden, müssen sie durch digitale Zwillinge repräsentiert und die Betriebsinformationen auf einen semantischen Standard abgebildet werden. In dieser Arbeit wird ein Konzept entwickelt, das digitale Zwillinge für Bestandsanlagen automatisiert generiert und ihre Informationen auf einen semantischen Standard abbildet. Dieses Konzept wird für Bestandsanlagen der technischen Gebäudeausrüstung umgesetzt. Um die heterogenen Betriebsinformationen zu verarbeiten, wird ein vierstufiger Klassifizierungsalgorithmus entwickelt. Dieser verarbeitet die Informationen mittels Language-Models und bildet sie auf einen Standard ab. Die nun semantisch einheitlichen Informationen der digitalen Zwillinge können automatisiert von verschiedenen Anwendungen erkundet

werden. Es wird ein prototypisches technisches Monitoring implementiert, das die Informationen der digitalen Zwillinge erkundet und auf Basis dieser Informationen automatisierte Visualisierungen generiert.

## Abstract

Industrie 4.0 aims for dynamic, flexible, and globally connected value-added systems. Achieving this necessitates systems that autonomously interact, relying on interoperable digital twins representing various assets. Semantic interoperability, enabling these twins to comprehend exchanged information, poses a significant challenge in Industrie 4.0 due to diverse vocabularies and semantics used across companies and industries.

In this work a concept to achieve semantic interoperability among digital twins, addressing both homogeneous and heterogeneous language areas, is developed. For homogeneous areas, a life cycle-oriented process model facilitates consistent derivation of Industrie 4.0 submodels, exemplified through pump submodels. For the heterogeneous area, a method is being developed to automatically map the different descriptions of the vocabularies of digital twins onto a standard. Language models and sentence embeddings are used to automatically compare the elements of the digital twin vocabularies with those of other communication participants, map them to one another and thus achieve semantic interoperability.

Integrating existing systems into Industrie 4.0 networks requires representing them as digital twins and mapping their operational data to a semantic standard. This work presents a concept to automatically generate digital twins for existing systems and map their information to a semantic standard. Demonstrated through existing technical building equipment systems, this concept employs a four-stage classification algorithm to process heterogeneous operational data using language models, resulting in standardized information across digital twins. This uniform data enables automated exploration by various applications. Furthermore, a prototype for technical monitoring is implemented, leveraging digital twin information to generate automated visualizations.

# Inhaltsverzeichnis

<b>Kurzfassung</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Struktur der Arbeit . . . . .	5
<b>2 Grundlagen</b>	<b>7</b>
2.1 Soziale Interaktion und Kommunikation . . . . .	7
2.2 Linguistik . . . . .	8
2.3 Computerlinguistik . . . . .	11
2.3.1 Stochastische Ansätze . . . . .	12
2.3.2 Transformer Architektur . . . . .	13
2.3.3 Pretrained Language Models . . . . .	15
2.3.4 Text Classification . . . . .	16
2.3.5 Paraphrase Identification . . . . .	18
2.3.6 Wissensgraphen in Kombination mit Language Models . . . . .	18
2.4 Interoperabilität . . . . .	19
2.4.1 Definition Interoperabilität . . . . .	20
2.4.2 Ebenen der Interoperabilität . . . . .	21
2.4.3 Aktueller Stand: Erfüllung der Interoperabilitätsebenen . . . . .	25
2.4.4 Semantische Interoperabilität . . . . .	26
2.5 Europäische Klimapolitik . . . . .	27
2.5.1 Europäische Klimaziele . . . . .	27
2.5.2 Maßnahmen zur Steigerung der Energieeffizienz . . . . .	28
2.5.3 Überwachung der Energieeffizienz in Gebäuden . . . . .	30
<b>3 Anforderungsanalyse</b>	<b>32</b>
3.1 Anforderungen an ein automatisiertes Monitoring . . . . .	32
3.2 Diskussion Anforderungsanalyse . . . . .	35
<b>4 Stand der Technik</b>	<b>38</b>
4.1 Digitale Zwillinge . . . . .	38

4.2	Interoperabilität im Kontext Industrie 4.0 . . . . .	40
4.2.1	Referenzarchitekturmodell Industrie 4.0 . . . . .	40
4.2.2	Struktur der Verwaltungsschale . . . . .	41
4.2.3	Semantische Interoperabilität von Verwaltungsschalen . . . . .	44
4.2.4	Industrie 4.0 Service Model . . . . .	45
4.2.5	BaSyx 4.0 – Industrie 4.0 Operating System . . . . .	45
4.3	Kommunikationsprotokolle der Gebäudeautomation . . . . .	46
4.4	Klassifizierung heterogener Gebäudeautomationsdaten . . . . .	47
4.5	Diskussion - Aktueller Stand der Interoperabilität im Bereich Industrie 4.0	48
<b>5</b>	<b>Semantische Interoperabilität - Homogene Semantik</b>	<b>55</b>
5.1	Entwurf von Industrie 4.0 Teilmodellen – Vorgehensmodell . . . . .	55
5.1.1	Lebenszyklusphasen als Grundlage von Teilmodellen . . . . .	56
5.1.2	Klassifikation von Assets . . . . .	58
5.1.3	Kombination der grundlegenden Assetfunktionen und Assetklassifikation . . . . .	60
5.2	Teilmodelle für Pumpen . . . . .	61
5.2.1	Vorgehensweise Entwicklung von Teilmodellen . . . . .	61
5.2.2	Teilmodelle Design, System-Requirements und Implementation des Anwendungsfalls Konfiguration . . . . .	63
5.2.3	Teilmodelle Condition-Based-, Preventive- und Breakdown-Maintenance des Anwendungsfalls Instandhaltung . . . . .	64
5.2.4	Teilmodelle des Anwendungsfalls Betrieb . . . . .	65
5.2.5	Integration der Teilmodellinhalte in ECLASS . . . . .	68
5.3	Fazit homogene Semantik . . . . .	69
<b>6</b>	<b>Semantische Interoperabilität - Heterogene Semantik</b>	<b>70</b>
6.1	NLP als Basis für semantische Interoperabilität von AAS . . . . .	70
6.1.1	Datensatz mit heterogenen Beschreibungen . . . . .	71
6.1.2	Sentence Embeddings zur Verarbeitung heterogener Semantik . . . . .	72
6.1.3	Erweiterung mit Entity Embeddings . . . . .	76
6.1.4	Large Language Models zur Verarbeitung heterogener Semantik . . . . .	79
6.1.5	Metadaten als zusätzliche Klassifizierung . . . . .	80
6.1.6	Evaluation . . . . .	85
6.2	Semantic Matching als Industrie 4.0 Service . . . . .	90
6.2.1	Spezifikation des Industrie 4.0 Interfaces für Semantic Matching . . . . .	90
6.2.2	Umsetzung des Interfaces als Industrie 4.0 Service . . . . .	93
6.2.3	Prototypische Implementierung des Semantic Matching Service . . . . .	96
6.3	Fazit heterogene Semantik . . . . .	102

<b>7</b>	<b>Digitale Zwillinge für Bestandsanlagen</b>	<b>103</b>
7.1	Konzept zur Generierung digitaler Zwillinge für Bestandsanlagen . . . . .	103
7.1.1	Spezifikation des branchenunabhängigen Konzepts . . . . .	104
7.1.2	Umsetzung des allgemeinen Konzepts für Anlagen der technischen Gebäudeausrüstung . . . . .	106
7.1.3	Ableitung von Teilmodellen für Bestandsanlagen auf Basis grundlegenden Lebenszyklusprozesse . . . . .	108
7.2	Abbildung des BACnet Protokolls auf die Verwaltungsschale . . . . .	109
7.2.1	BaSyx Data Bridge als Basis für die Abbildung von BACnet auf die Verwaltungsschale . . . . .	109
7.2.2	Teilmodell BACnet Datapoints Information . . . . .	109
7.3	NLP zur automatisierten Abbildung semantisch heterogener Datenpunkte .	110
7.3.1	Entwurf vierstufiger Klassifizierungsprozess . . . . .	111
7.3.2	Datensatz zum Training der Modelle . . . . .	112
7.3.3	Training und Evaluation der Modelle . . . . .	115
7.4	Digitale Zwillinge technischer Systeme als Grundlage für ein technisches Monitoring . . . . .	127
7.4.1	Teilmodell NLP Classification Result . . . . .	127
7.4.2	Generierung Strukturinformationen . . . . .	128
7.4.3	Teilmodelle Operating Information . . . . .	129
7.4.4	Teilmodelle Technical Monitoring . . . . .	130
7.5	Fazit Bestandsanlagen . . . . .	131
<b>8</b>	<b>Prototypische Realisierung im Anwendungsfall Technisches Monitoring</b>	<b>133</b>
8.1	Generierung von Verwaltungsschalen für Automationsstationen . . . . .	134
8.2	Verarbeitung heterogener Datenpunkte . . . . .	134
8.3	Technische Monitoring Anwendung . . . . .	135
<b>9</b>	<b>Zusammenfassung und Ausblick</b>	<b>143</b>
9.1	Zusammenfassung . . . . .	143
9.2	Wissenschaftlicher Beitrag dieser Arbeit . . . . .	146
9.3	Ausblick . . . . .	147
	<b>Literatur</b>	<b>150</b>
	<b>Ehrenerklärung</b>	<b>169</b>
	<b>Eigene Veröffentlichungen</b>	<b>170</b>
<b>A</b>	<b>Definitionen ChatGPT</b>	<b>172</b>

<b>B Beste Modelle im Bereich Sentence Embeddings</b>	<b>173</b>
<b>C Übersicht Datensatz auf Datenpunktebene</b>	<b>174</b>
<b>D Entwicklung des F1-Score und Evaluation-Loss der Modelle</b>	<b>188</b>
<b>E Wahrheitsmatrizen der besten Modelle</b>	<b>200</b>

## Abkürzungsverzeichnis

<b>AAS</b>	Asset Administration Shell
<b>AID</b>	Asset-Interface-Description
<b>AIMC</b>	Asset-Interfaces-Mapping-Configuration
<b>AKS</b>	Anlagenkennzeichnungsschlüssel
<b>AMEV</b>	Arbeitskreis Maschinen- und Elektrotechnik staatlicher und kommunaler Verwaltungen
<b>AMon</b>	Anlagenmonitoring
<b>API</b>	Programmierschnittstelle
<b>BACnet</b>	Building Automation and Control Network
<b>CD</b>	Concept-Description
<b>CEL</b>	Cross-Entropy-Loss
<b>CLM</b>	Casual-Language-Modeling
<b>DDC</b>	Direct Digital Control
<b>DTDL</b>	Digital Twins Definition Language
<b>EE</b>	Entity-Embedding
<b>EED</b>	Richtlinie zur Energieeffizienz
<b>EMon</b>	Energiemonitoring
<b>EMS</b>	Energiemanagementsystem
<b>EnPI</b>	Energieleistungskennzahl
<b>EPBD</b>	Richtlinie über die Gesamtenergieeffizienz von Gebäuden
<b>ERMon</b>	Einregulierungsmonitoring

<b>ESG</b>	Environmental, Social and Governance
<b>EU</b>	Europäische Union
<b>FFN</b>	Feed-Forward-Network
<b>GA</b>	Gebäudeautomation
<b>GBMon</b>	Gebäude- und Behaglichkeitsmonitoring
<b>GPU</b>	Graphics-Processing-Unit
<b>HTTP</b>	Hypertext Transfer Protocol
<b>I4.0</b>	Industrie 4.0
<b>ID</b>	Identifikationsnummer
<b>IDiS</b>	Initiative Digitale Standards
<b>IDTA</b>	Industrial Digital Twin Association
<b>IEC CDD</b>	IEC Common Data Dictionary
<b>IoT</b>	Internet of Things
<b>IP</b>	Internet Protokoll
<b>IRDI</b>	International Registration Data Identifier
<b>IRI</b>	Internationalized Resource Identifier
<b>ISIC</b>	International Standard Industrial Classification of All Economic Activities
<b>KG</b>	Knowledge-Graph
<b>KI</b>	Künstliche Intelligenz
<b>KNN</b>	Künstliche Neuronale Netze
<b>LAC</b>	Language Acquisition Device
<b>LLM</b>	Large-Language-Model
<b>LM</b>	Language-Model
<b>LON</b>	Local Operating Network
<b>LSTM</b>	Long-Short-Term-Memory
<b>LZMon</b>	Langzeitmonitoring

<b>ML</b>	Maschinelles Lernen
<b>MLM</b>	Masked-Language-Modeling
<b>MNRL</b>	Multiple-negatives-Ranking-Loss
<b>NASA</b>	National Aeronautics and Space Administration
<b>NGSI-LD</b>	Next Generation Services Interface-Linked Data
<b>NLI</b>	Natural-Language-Inference
<b>NLP</b>	Natural Language Processing
<b>PH</b>	Project Haystack
<b>PI</b>	Paraphrase-Identification
<b>PLM</b>	Pretrained-Language-Model
<b>QQP</b>	Quora Question Pairs
<b>RAMI 4.0</b>	Referenzarchitekturmodell Industrie 4.0
<b>RE</b>	Relation-Embedding
<b>RLT</b>	Raumlufttechnik
<b>RNN</b>	Recurrent-Neural-Network
<b>SDK</b>	Software Development Kit
<b>SE</b>	Submodel-Element
<b>SEm</b>	Sentence-Embedding
<b>seq2seq</b>	Sequence-to-Sequence
<b>SM</b>	Semantic-Matching
<b>SMC</b>	Submodel-Element-Collection
<b>SPS</b>	Speicherprogrammierbare Steuerung
<b>TC</b>	Text-Classification
<b>TGA</b>	Technische Gebäudeausrüstung
<b>TMon</b>	Technisches Monitoring
<b>UML</b>	Unified Modeling Language

<b>URL</b>	Universal Resource Locator
<b>WE</b>	Word-Embedding
<b>WoT</b>	Web of Things

# Abbildungsverzeichnis

1.1	Automatisierter Zugriff auf Informationen semantisch homogener und heterogener AAS innerhalb eines Netzwerks . . . . .	4
2.1	Transformer Architektur nach [36] . . . . .	14
3.1	Visuelle Darstellung der spezifizierten Anforderungen . . . . .	36
4.1	RAMI 4.0 nach [119] . . . . .	40
4.2	Überblick über das Metamodell der AAS nach [11] . . . . .	41
4.3	Abstrakte SE-Klasse mit Subklassen nach [11] . . . . .	42
4.4	Data Specification Template IEC 61360 nach [11] . . . . .	43
4.5	Aktuelle Herausforderungen im Bereich der I4.0 hinsichtlich des Erreichens von Interoperabilität . . . . .	50
5.1	Zugeordnete Lebenszyklusprozesse der ISO 15288 zu dem Lebenszyklusmodell aus RAMI 4.0 . . . . .	56
5.2	Beispiele für abgeleitete Teilmodelle von den grundlegenden Assetfunktionen	58
5.3	Beispiel einer Klassifikation von Assets . . . . .	59
5.4	Wiederverwendbarkeit von SEs . . . . .	60
5.5	Verortung der Anwendungsfälle im Lebenszyklusmodell . . . . .	61
5.6	Abgeleitete Funktionen von Betreiben . . . . .	62
5.7	Mögliche Regelungsarten für Pumpen . . . . .	63
5.8	Abstrakte Klasse Measurements . . . . .	66
5.9	Teilmodell Archiving . . . . .	67
5.10	SMC Discrete Input . . . . .	68
6.1	Semantische Interoperabilität von AAS mittels SM . . . . .	71
6.2	Ablauf des Trainings von SEms . . . . .	73
6.3	Ablauf Evaluation SEms . . . . .	75
6.4	Ablauf Evaluation Kombination SEms mit statischen EEs . . . . .	78
6.5	Ablauf Evaluation LLMs . . . . .	80
6.6	Ablauf Evaluation SEms mit Metadaten . . . . .	84
6.7	Verarbeitung von AAS als Vorbereitung des SM . . . . .	94
6.8	SM auf AAS-Ebene . . . . .	95
6.9	SM auf SE-Ebene . . . . .	96

---

6.10	Bereitstellen von AAS für das SM . . . . .	97
6.11	Informationen des SM-Service . . . . .	98
6.12	SM innerhalb einer AAS . . . . .	99
6.13	Visualisierung von drei möglichen Informationen des Dokumentationsmanagements . . . . .	100
6.14	Übersicht Ergebnisse der gesuchten SEs . . . . .	100
6.15	Ergebnis eines NLP-basierten Matchings . . . . .	101
6.16	Ergebnis eines Matchings auf Basis gleicher Semantic-IDs . . . . .	101
7.1	Integration von Bestandsanlagen in ein I4.0-Wertschöpfungsnetzwerk . . . . .	104
7.2	Generierung von Strukturinformationen für Bestandsanlagen . . . . .	107
7.3	Visualisierung Teilmodell BACnet Datapoints Information . . . . .	110
7.4	Überblick über den vierstufigen Klassifizierungsprozess mit den Stufen Grundfunktion, Unterfunktion, Entität und Datenpunkt . . . . .	111
7.5	Beispielhafte Klassifizierung eines Datenpunkts bestehend aus Object Name und Description . . . . .	111
7.6	Beispiel für die entwickelte Methodik zur Klassifizierung eines Datenpunkts mithilfe von NLI . . . . .	116
7.7	Entwicklung des F1-Score und Evaluation-Loss in Kombination mit zunehmender Anzahl von Trainingsbeispielen (gBERT Grundfunktionen) . . . . .	121
7.8	Entwicklung des F1-Score und Evaluation-Loss in Kombination mit zunehmender Anzahl von Trainingsbeispielen (gBERT Datenpunkte) . . . . .	121
7.9	Wahrheitsmatrizen mit korrekt und inkorrekt klassifizierten Datenpunkten . . . . .	126
7.10	Visualisierung des Teilmodells NLP Classification Result . . . . .	127
7.11	Überblick über das entwickelte Konzept für Strukturinformationen . . . . .	128
7.12	Teilmodell Technical Monitoring Heating Circuit AMEV . . . . .	131
8.1	Visuelle Darstellung der Softwarearchitektur einschließlich einzelner Komponenten . . . . .	133
8.2	Eingabemaske Registrierung Unternehmen . . . . .	135
8.3	Visualisierung registrierter Liegenschaften und Gebäude . . . . .	136
8.4	AAS des Unternehmens, Liegenschaften und Gebäude im Backend . . . . .	136
8.5	AAS der Liegenschaften und Gebäude im Frontend . . . . .	137
8.6	Visualisierung der erkannten Grundfunktionen eines Gebäudes . . . . .	137
8.7	Visualisierung eines Heizkreises mit zugehörigen Komponenten . . . . .	138
8.8	Beispielhafte AAS einer Komponente mit Datenpunkten . . . . .	138
8.9	AAS einer Komponente mit dazugehörigen Teilmodellen im Backend . . . . .	139
8.10	Trend der Vor- und Rücklauftemperatur eines Heizkreises . . . . .	140
8.11	Visualisierung des Monitorings zweier Anlagen . . . . .	141

A.1	Definition Interoperabilität des Chatbots ChatGPT . . . . .	172
A.2	Definition semantische Interoperabilität des Chatbots ChatGPT . . . . .	172
B.1	Screenshot der aktuell besten Modelle im Bereich SEms und Semantic Search	173
D.1	F1-Score und Evaluation-Loss der Ebene Grundfunktion . . . . .	188
D.2	F1-Score und Evaluation-Loss der Ebene Wärme versorgen . . . . .	189
D.3	F1-Score und Evaluation-Loss der Ebene Luft versorgen . . . . .	190
D.4	F1-Score und Evaluation-Loss der Ebene Medien versorgen . . . . .	191
D.5	F1-Score und Evaluation-Loss der Ebene Kälte versorgen . . . . .	192
D.6	F1-Score und Evaluation-Loss der Ebene Sichern . . . . .	193
D.7	F1-Score und Evaluation-Loss der Ebene Wärme erzeugen . . . . .	194
D.8	F1-Score und Evaluation-Loss der Ebene Wärme verteilen . . . . .	195
D.9	F1-Score und Evaluation-Loss der Ebene Luft bereitstellen . . . . .	196
D.10	F1-Score und Evaluation-Loss der Ebene Luft verteilen . . . . .	197
D.11	F1-Score und Evaluation-Loss der Ebene Kälte erzeugen . . . . .	198
D.12	F1-Score und Evaluation-Loss der Ebene Datenpunkte . . . . .	199
E.1	Wahrheitsmatrizen der Ebene Grundfunktion . . . . .	200
E.2	Wahrheitsmatrizen der Ebene Wärme versorgen . . . . .	201
E.3	Wahrheitsmatrizen der Ebene Luft versorgen . . . . .	202
E.4	Wahrheitsmatrizen der Ebene Medien versorgen . . . . .	203
E.5	Wahrheitsmatrizen der Ebene Kälte versorgen . . . . .	204
E.6	Wahrheitsmatrizen der Ebene Sichern . . . . .	205
E.7	Wahrheitsmatrizen der Ebene Wärme erzeugen . . . . .	206
E.8	Wahrheitsmatrizen der Ebene Wärme verteilen . . . . .	207
E.9	Wahrheitsmatrizen der Ebene Luft bereitstellen . . . . .	208
E.10	Wahrheitsmatrizen der Ebene Luft verteilen . . . . .	209
E.11	Wahrheitsmatrizen der Ebene Kälte erzeugen . . . . .	210

## Tabellenverzeichnis

2.1	Teilbereiche der Linguistik . . . . .	9
2.2	Vergleich Definitionen Interoperabilitätsebenen . . . . .	24
4.1	Überblick über die Ansätze zur automatisierten Klassifizierung von Datenpunkten auf Basis der Semantik . . . . .	47
4.2	Verschiedene Semantic-IDs des ECLASS Standards für die maximale Umgebungstemperatur . . . . .	52
6.1	Beispiele für Paraphrasen der Merkmale der Teilmodelle . . . . .	72
6.2	Parameter des erweiterten Pretraining . . . . .	74
6.3	Übersicht über kombinierte LMs mit EEs . . . . .	78
6.4	Mögliche Metadaten der AAS . . . . .	81
6.5	Übergeordnete Kategorien Einheit . . . . .	82
6.6	Übergeordnete Kategorien Datentyp . . . . .	83
6.7	Übersicht der Basismodelle . . . . .	85
6.8	Ergebnisse SM auf dem Evaluationsdatensatz . . . . .	87
6.9	Übersicht über das SM-Interface . . . . .	90
6.10	Operation Semantic-Query-AAS . . . . .	91
6.11	Operation Semantic-Query-Submodel-Element . . . . .	92
6.12	Operation Semantic-Query-Match-Object . . . . .	93
7.1	Übersicht über den erstellten Datensatz mit Namen und Anzahl der Vorkommnisse aus den drei Ebenen: Grundfunktion, Unterfunktion, Entität . . . . .	113
7.2	Variation von Hyperparametern für jeden Klassifizierungsschritt, die zum Trainieren verschiedener Modelle verwendet werden . . . . .	117
7.3	Übersicht über die Anzahl der Trainingsbeispiele und anderer Hyperparameter, die zum Erreichen der angegebenen Schwellenwerte erforderlich sind . . . . .	118
7.4	Mittelwert und Median der Anzahl der Trainingsbeispiele pro Label, die zum Erreichen der angegebenen Schwellenwerte erforderlich sind . . . . .	120
7.5	Ergebnisse für den Validierungsdatensatz des undersampled Ansatzes mit 50, 100 und 200 Beispielen pro Klasse . . . . .	123
7.6	Übersicht über die besten Ergebnisse (Validierungsdatensatz) mit verwendeten Hyperparametern der verschiedenen Klassifizierungsstufen . . . . .	124

C.1 Übersicht der Komponenten mit möglichen Datenpunkten . . . . . 174



# 1 Einleitung

In diesem Kapitel wird die Motivation und der Aufbau der Arbeit beschrieben. Im ersten Abschnitt wird die Arbeit in einen größeren Kontext eingebettet, indem aktuelle Herausforderungen Deutschlands und Europas erläutert werden. Dabei werden die Themenfelder herausgearbeitet, zu denen diese Arbeit einen Beitrag leistet. Der zweite Abschnitt gibt einen Überblick über die Gliederung der Arbeit und beschreibt kurz, welche Themen in den einzelnen Kapiteln behandelt werden.

## 1.1 Motivation

Die digitale sowie technologische Souveränität Deutschlands sichern und die Potenziale der Digitalisierung nutzen (1) und den Klimaschutz vorantreiben (2) sind zwei von sechs definierten Missionen der „Zukunftsstrategie für Forschung und Innovation“ der aktuellen Bundesregierung [1]. Sowohl der Klimawandel als auch die Digitalisierung haben Einfluss auf verschiedene Branchen und gelten als Querschnittsthemen, die die Wirtschaft und das Privatleben betreffen. Verschiedene Maßnahmen werden von der Bundesregierung ergriffen, um Fortschritte in diesen Bereichen zu erzielen. Das übergeordnete Ziel der Klimastrategie Deutschlands ist die Reduktion der Netto-Treibhausgasemissionen im Jahr 2045 auf null. Als Zwischenziel sollen bis 2030 die Emissionen um 65% im Vergleich zum Jahr 1990 gesenkt werden [2]. Diese Reduktion kann nur branchenübergreifend erreicht werden, wofür verschiedene Fördermaßnahmen und Ziele definiert wurden. Eine Schlüsselrolle bei der Erreichung der Klimaziele kommt dabei dem Gebäudebetrieb zu. 36% des Endenergieverbrauchs und 28% der Treibhausgasemissionen entfallen auf den Betrieb von Gebäuden [3]. Um die allgemeinen Ziele der Bundesregierung für 2030 und 2045 zu erreichen, muss der Gebäudebetrieb nachhaltiger und energieeffizienter werden. Die Bundesregierung setzt auf digitale Innovationen, um Deutschlands Nachhaltigkeitsziele zu erreichen. Im Aktionsplan „Natürlich.Digital.Nachhaltig“ werden entsprechende Maßnahmen dargestellt [4]. Dazu werden drei Handlungsfelder beschrieben, welche digitale Technologien und nachhaltige Entwicklung verbinden sollen, um eine digitale, nachhaltige Zukunft zu ermöglichen. Digitalisierung ist demnach eine Schlüsselkomponente, damit Deutschland seine Klimaziele erreichen kann.

Der Bereich der Digitalisierung wird in Deutschland ebenfalls von der Bundesregierung gefördert. Die aktuelle „Digitalstrategie“ der Bundesregierung identifiziert drei Handlungsfelder, die in den kommenden Jahren verstärkt bearbeitet werden sollen. Im Handlungsfeld „Innovative Wirtschaft, Arbeitswelt, Wissenschaft und Forschung“ werden unter anderem die Stärkung der Position der Industrie 4.0 (I4.0), der Einsatz digitaler Zwillinge, sowie die Umsetzung der Strategie für Künstliche Intelligenz (KI) als zentrale Aspekte beschrieben [5]. Der Einsatz digitaler Zwillinge wird in verschiedenen Bereichen wie Gebäuden, urbanen Gebieten und der Industrie vorangetrieben. Die Weiterentwicklung des KI-Standorts Deutschland zählt zu den dringendsten Aufgaben der Bundesregierung. Insbesondere seit dem „Chat-GPT-Moment“ im November 2022 hat die Regierung beschlossen, Entwicklungen im Bereich der KI stärker zu fördern. Konkret sollen Forschungsaktivitäten gestärkt, die Infrastruktur ausgebaut, sowie der Transfer von Forscherkenntnissen in die Wirtschaft erleichtert werden [6].

Das Thema I4.0 wird in Deutschland seit 2011 im Zuge der „Hightech-Strategie“ gefördert [7]. Bisher starre und fest definierte Wertschöpfungsketten werden in der I4.0 dynamisch und flexibel und ergeben unternehmensübergreifende Wertschöpfungsnetzwerke [8]. Im Leitbild 2030 der „Plattform Industrie 4.0“ werden die drei Handlungsfelder *Souveränität*, *Nachhaltigkeit* und *Interoperabilität* als zentrale Elemente bei der Umsetzung der I4.0 definiert [9]. Das Handlungsfeld der Nachhaltigkeit beschreibt unter anderem, dass eine moderne industrielle Wertschöpfung auch einen Beitrag zum Klimaschutz leisten muss. Das zweite Handlungsfeld beschreibt, dass Interoperabilität zwischen allen Akteuren eines Wertschöpfungsnetzwerks gegeben sein muss, damit diese Netzwerke dynamisch und flexibel gestaltet werden können [9]. Als Basis für Interoperabilität wurde die *Verwaltungsschale* (Asset Administration Shell (AAS)) entwickelt, welche die Umsetzung des digitalen Zwillings der I4.0 ist [10]. Dynamische und flexible Netzwerke sollen mit Hilfe von AAS realisiert werden. Hierfür sind *aktive AAS* erforderlich, die ihre Umgebung erkunden, mit anderen AAS kommunizieren, eigenständige Interaktionen initiieren und Verhandlungen abschließen können. AAS sollen eigenständig handeln, um dynamische Wertschöpfungsnetzwerke zu schaffen, die nicht mehr an starre Verknüpfungen einzelner Teilnehmer gebunden sind. Das Metamodell der AAS bildet die Basis für die einheitliche Strukturierung von Informationen [11]. Durch die Verwendung standardisierter Schnittstellen kann einheitlich auf diese Informationen zugegriffen werden [12]. Darüber hinaus soll die Entwicklung einer *I4.0-Sprache* eigenständige Interaktionen der AAS ermöglichen [13]. Damit AAS miteinander interagieren können, müssen sie die Sprache ihres Interaktionspartners verstehen. Wie beim Menschen kann eine erfolgreiche Interaktion nur stattfinden, wenn die Sprache des Gegenübers korrekt erfasst wird. Andernfalls ist eine Interaktion nur schwer umzusetzen. Um ein gemeinsames Verständnis zwischen AAS zu erreichen, werden zurzeit *Teilmodelle* standardisiert [14]. Teilmodelle beschreiben unterschiedliche Aspekte eines Assets und enthalten die Informationen in Form standardisierter Elemente. Diese Elemente verweisen

wiederum auf Vokabulare mit standardisierter Semantik. Wenn eine Interaktion zwischen AAS stattfindet, die dieselben Teilmodelle mit einheitlich standardisierter Semantik verwenden, ist die Grundlage für eine automatisierte Interaktion gegeben. Die AAS sind semantisch interoperabel.

Wie zu Beginn beschrieben sollen im Rahmen von I4.0 dynamische Wertschöpfungsnetzwerke entstehen, die unternehmensübergreifend und global ausgerichtet sind. Aufgrund der Vielzahl an Teilnehmern und Domänen innerhalb eines I4.0-Wertschöpfungsnetzwerks ist jedoch zu erwarten, dass eine durchgehend einheitliche semantische Beschreibung nur in seltenen Fällen möglich sein wird. Es wird demnach nicht nur ein einziges Vokabular geben, das von AAS verwendet wird, sondern verschiedene Vokabulare (standardisierte, herstellerepezifische etc.), welche untereinander semantisch heterogen ausgeprägt sind. Ähnlich wie Menschen werden nicht alle AAS dieselbe Sprache sprechen. Um trotz Sprachbarrieren eine Interaktion zwischen Menschen zu ermöglichen, kommen Übersetzer zum Einsatz. Früher wurden Übersetzungen ausschließlich von Menschen (Dolmetscher oder durch das Erlernen einer Fremdsprache) durchgeführt. Heute gibt es verschiedene Tools (bspw. „Google Translate“ [15] oder „DeepL“ [16]), die eine Übersetzung verschiedener Sprachen ermöglichen. Im übertragenen Sinne werden Sprachen wie Englisch dabei auf andere Sprachen wie Deutsch abgebildet. Dadurch wird ein gemeinsames Verständnis über die ausgetauschten Informationen gewährleistet, wodurch Interaktionen zwischen Menschen ermöglicht werden. Im Bereich der I4.0 existiert derzeit noch kein automatisierter Übersetzungsmechanismus, der verschiedene Sprachen aufeinander abbilden kann. Der aktuelle Forschungsansatz der I4.0 zur Interaktion von Systemen basiert auf homogenen semantischen Beschreibungen oder auf einem manuellen Mapping heterogener Beschreibungen [11]. Dies führt zu einem erhöhten Engineeringaufwand bei der Konfiguration, und erschwert die Erstellung dynamischer Netzwerke. Semantische Interoperabilität von AAS ist demnach aktuell nur gegeben, wenn eine homogene Semantik verwendet wird. Der Einsatz unbekannter und semantisch heterogener Vokabulare innerhalb eines Netzwerks verhindert momentan die semantische Interoperabilität von AAS.

In dieser Arbeit wird die Frage untersucht, wie semantische Interoperabilität zwischen semantisch homogenen und heterogenen AAS erreicht werden kann, um automatisierte Erkundungsmechanismen und Interaktionen innerhalb semantisch heterogener Netzwerke zu ermöglichen (vgl. Abbildung 1.1).

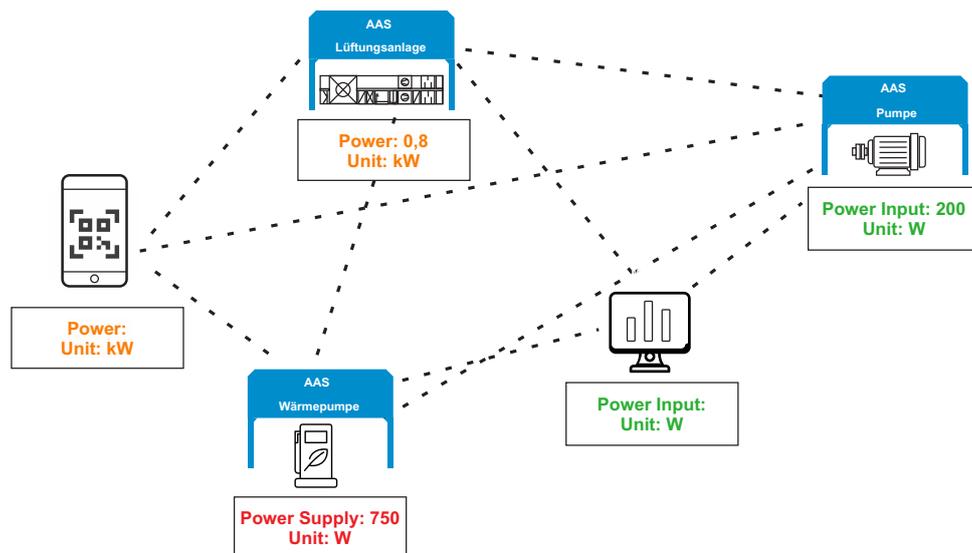


Abbildung 1.1: Automatisierter Zugriff auf Informationen semantisch homogener und heterogener AAS innerhalb eines Netzwerks

In der Abbildung sind verschiedene technische Komponenten dargestellt, die durch AAS repräsentiert werden und miteinander interagieren. Die Verwendung unterschiedlicher Namen zur Bezeichnung der Leistungsaufnahme in verschiedenen AAS führt jedoch zu einer Heterogenität der Beschreibungen. Trotzdem können die AAS aufgrund eines Übersetzungsmechanismus miteinander interagieren. Darüber hinaus können Anwendungen die AAS erkunden und auf die Informationen zugreifen. In den Anwendungen wird ebenfalls ein Vokabular verwendet, das teilweise heterogen und teilweise homogen zu den Vokabularen der AAS ist.

Um automatisierte Interaktionen und Erkundungsmechanismen zu ermöglichen, werden in dieser Arbeit drei zusammenhängende Konzepte entwickelt. Erstens wird ein Vorgehensmodell ausgearbeitet, das die Ableitung von Teilmodellen verschiedener Assets anhand eines lebenszyklusorientierten Modells ermöglicht. Auf Grundlage dieses Modells werden konkrete Teilmodelle für Pumpen erstellt. Zweitens wird eine Methode entwickelt, die es ermöglicht, semantisch heterogene Beschreibungen auf ein Zielvokabular abzubilden. Hierbei werden Verfahren aus dem Bereich des *Natural Language Processing (NLP)*, einem Teilgebiet der KI, angewendet. Die entwickelte Methodik ermöglicht die automatisierte Abbildung heterogener Semantik, wodurch manuelles Engineering vermieden und die Grundlage für automatisierte Erkundungsmechanismen und Interaktionen geschaffen wird. Drittens wird eine Methode entwickelt, um Bestandsanlagen durch AAS zu repräsentieren. Dadurch können sie grundsätzlich in ein I4.0-Netzwerk integriert und erkundet werden. Es wird ein Klassifikationsverfahren entwickelt, das die semantisch heterogenen Datenpunkte der Anlagen mittels NLP auf ein Zielvokabular abbildet. Im Rahmen dieser Arbeit werden die Grundzüge dieser Methode unabhängig von einem spezifischen Industriezweig entworfen und im Anschluss konkret für die Technische Gebäudeausrüstung (TGA) umgesetzt. Es

wird ein prototypisches Technisches Monitoring (TMon) entworfen und implementiert, das die Informationen der AAS erkundet und in die Anwendung integriert.

Zu Beginn der Einleitung wurde erläutert, dass eine erfolgreiche Bekämpfung des Klimawandels, sowie die Weiterentwicklung der Bereiche KI und Digitalisierung wichtige Elemente für die künftige Ausrichtung Deutschlands darstellen. Diese Arbeit verfolgt den Ansatz, dass der Einsatz von Methoden der KI und digitaler Technologien dazu beitragen kann, die Klimawende im Gebäudesektor zu unterstützen. Monitoring-Anwendungen sind die Grundlage, um Optimierungspotenziale in Gebäuden aufzuzeigen und diese ressourcenschonender und effizienter zu betreiben. Eine einfache Implementierung dieser Anwendungen erfordert einen hohen Grad an Automatisierung, der durch die Abbildung der Anlagen der TGA als AAS und die Verarbeitung semantisch heterogener Beschreibungen durch NLP ermöglicht wird. Es wird ein grundlegendes Konzept zur automatisierten Abbildung heterogener Beschreibungen entwickelt. Dieses ermöglicht semantische Interoperabilität zwischen AAS, wodurch Erkundungsmechanismen sowie automatisierte Interaktionen in verschiedenen Bereichen umgesetzt werden können. Dieses Konzept wird konkret im Bereich des TMon von Anlagen der TGA umgesetzt. Auf diese Weise werden die beiden Ansätze KI und AAS kombiniert, um einen klimafreundlicheren Betrieb von Gebäuden zu ermöglichen.

## 1.2 Struktur der Arbeit

Diese Arbeit ist in neun Kapitel gegliedert. Im zweiten Kapitel werden die Grundlagen der Arbeit beschrieben. Zunächst wird die Funktionsweise der Interaktionen zwischen Menschen und die Rolle der Sprache hierbei erläutert. Da in dieser Arbeit Methoden aus dem Bereich des NLP verwendet werden, werden auch die Grundlagen der Computerlinguistik dargestellt. Anschließend werden die Bereiche Interoperabilität und semantische Interoperabilität beschrieben. Um die Rolle von Gebäuden bei der Erreichung der deutschen und europäischen Klimaziele aufzuzeigen, wird die europäische Klimapolitik beschrieben. Es wird erörtert, welche Vorgaben in Gebäuden umgesetzt werden sollen.

Aufbauend auf den Grundlagen wird in Kapitel 3 eine Analyse der Anforderungen durchgeführt, welche erfüllt sein müssen, damit Systeme interoperabel miteinander interagieren können und ihre Informationen erkundbar sind.

Kapitel 4 beschreibt den Stand der Technik. Dabei wird das Konzept des digitalen Zwillings erläutert und es wird aufgezeigt, wie aktuell Interoperabilität im Bereich der I4.0 erreicht werden soll. Des Weiteren werden bestehende Methoden zur Verarbeitung heterogener Datenpunkte in der TGA vorgestellt. Zum Abschluss des Kapitels wird der aktuelle Stand hinsichtlich Interoperabilität im Bereich der I4.0 erörtert. In dieser Diskussion werden die Grundlagen aus Kapitel 2 und die Anforderungen des dritten Kapitels sowie der Stand der

Technik miteinander verknüpft. Dabei werden Herausforderungen aufgezeigt, die derzeit im Hinblick auf Interoperabilität bestehen, und inwieweit diese Arbeit dazu beiträgt, diese zu erfüllen. In diesem Abschnitt wird die zentrale Forschungsfrage der Arbeit erörtert.

In den Kapiteln 5, 6 und 7 werden die drei entwickelten Konzepte dieser Arbeit dargestellt. Jedes Kapitel beginnt mit einer Erläuterung des Ansatzes und der Methodik, gefolgt von der konkreten Umsetzung des entwickelten Konzepts. In Kapitel 5 wird das Vorgehensmodell vorgestellt, welches zur Ableitung von Teilmodellen genutzt werden kann. Auf Basis des Modells werden konkrete Teilmodelle für Pumpen ausgeprägt.

In Kapitel 6 wird das Konzept zur automatisierten Verarbeitung heterogener Semantik mithilfe von NLP-Methoden als Basis für semantische Interoperabilität erläutert. Im ersten Teil des Kapitels wird der entwickelte Ansatz, die trainierten Modelle sowie die Evaluation beschrieben. Der zweite Teil beschäftigt sich damit, wie die Methode in ein I4.0-Interface integriert werden kann und wie dieses als Service umgesetzt wird. Eine erste prototypische Implementierung des Services wird realisiert.

Das Konzept zur Integration bestehender Anlagen in I4.0-Umgebungen und zur Verarbeitung heterogener Datenpunktbeschreibungen mithilfe von NLP wird in Kapitel 7 dargestellt.

Im achten Kapitel wird die prototypische Implementierung einer TMon-Anwendung beschrieben. Diese demonstriert die Verwendung der entwickelten Methode in der Praxis. Zu diesem Zweck wurde eine Anwendung entworfen, die eine NLP-Pipeline zur Klassifizierung von Datenpunkten startet. Darauf aufbauend werden Strukturinformationen generiert und digitale Zwillinge eines Gebäudes erstellt. Die Monitoring-Anwendung erkundet diese Informationen und erstellt basierend darauf Monitoring-Charts.

Im neunten Kapitel wird die Arbeit zusammengefasst und es werden verbleibende Herausforderungen in Bezug auf Interoperabilität sowie zukünftige Forschungsmöglichkeiten beschrieben.

## 2 Grundlagen

Im Mittelpunkt der Arbeit steht die semantische Interoperabilität digitaler Zwillinge, um darauf aufbauend automatisierte Erkundungsmechanismen und Interaktionen zu ermöglichen. Um dies zu erreichen, müssen bestimmte Anforderungen erfüllt sein, die in Kapitel 3 erarbeitet werden. In den folgenden fünf Unterkapiteln werden die Grundlagen beschrieben, aus denen die Anforderungen abgeleitet werden. Zunächst wird definiert, was Interaktion ist und wie sie zwischen Menschen abläuft (Kapitel 2.1). Da Menschen in der Lage sind, miteinander zu interagieren, es aber auch hier bestimmter Anforderungen bedarf, um dies zu ermöglichen, wird zunächst untersucht, wie die Interaktion zwischen Menschen abläuft. Ein wichtiger Bereich der Interaktion ist dabei die Sprache. In den Kapiteln 2.2 und 2.3 werden Aspekte der Linguistik und Computerlinguistik beschrieben, um zu zeigen, wie Mensch und Computer Sprache verstehen. Kapitel 2.4 behandelt das Thema der Interoperabilität zwischen technischen Komponenten. Ist diese gegeben, können die Informationen der Komponenten automatisiert erkundet und in Anwendungen eingebunden werden. In dieser Arbeit werden Monitoring-Anwendungen für technische Komponenten der TGA entworfen. Monitoring-Anwendungen sind ein wichtiges Instrument, um die europäischen und deutschen Klimaziele zu erreichen. Das Kapitel 2.5 beschreibt den aktuellen Stand der europäischen Klimapolitik und wie diese in Gebäuden umgesetzt werden soll.

### 2.1 Soziale Interaktion und Kommunikation

Im Allgemeinen bezeichnet Interaktion das Zusammenspiel von zwei oder mehr Gegebenheiten [17]. Handelt es sich bei diesen Gegebenheiten um zwei oder mehr Personen, spricht man von *sozialer Interaktion* [17]. Soziale Interaktion wird definiert als die Prozesse wechselseitiger Beeinflussung zwischen Menschen, sowie die daraus resultierenden Veränderungen von Verhaltensweisen und Einstellungen [18]. Zentral ist hierbei der Aspekt der wechselseitigen Beeinflussung. Es existiert nicht nur ein Sender und ein Empfänger, sondern beide Beteiligten sind sowohl Sender als auch Empfänger und beeinflussen sich gegenseitig.

Von der Interaktion zu unterscheiden ist die *Kommunikation* zwischen Menschen. Kommunikation wird als ein Prozess definiert, bei dem Informationen von einer Person zu

einer anderen übertragen werden. Dabei wird zwischen verbaler (mündlich und schriftlich) und nonverbaler (Mimik, Gestik etc.) Kommunikation unterschieden. Während soziale Interaktion durch wechselseitige Beeinflussung definiert ist, ist dies bei der Kommunikation nicht zwingend der Fall [19]. Begrüßt beispielsweise ein Moderator seine Fernsehzuschauer, so ist dies ein kommunikativer, aber kein interaktiver Akt [20]. Da die Reaktion des Zuschauers dem Moderator verborgen bleibt, kann er wiederum nicht darauf reagieren, so dass die wechselseitige Beeinflussung als Grundvoraussetzung für Interaktion nicht gegeben ist. Es handelt sich also um eine einseitige Kommunikation des Moderators.

Um die Kommunikation zwischen Menschen formal zu beschreiben, gibt es verschiedene Modelle. Eines der ersten veröffentlichten Modelle ist das Modell von Shannon und Weaver, welches heute als „Sender-Empfänger-Modell“ bezeichnet wird [21]. Das Modell wurde ursprünglich für die Nachrichtentechnik entwickelt, um zu beschreiben, wie Nachrichten störungsfrei übertragen werden können und besteht aus fünf Komponenten. Die ersten beiden Komponenten sind der Sender (*Information-Source*) einer Nachricht und der Empfänger (*Destination*) dieser Nachricht. Weitere Komponenten sind der Kodierer (*Transmitter*) auf der Seite des Senders und der Dekodierer (*Receiver*) auf der Seite des Empfängers. Diese werden benötigt, um die Information in ein übertragbares Signal umzuwandeln, bzw. um das Signal wieder in eine verständliche Information umzuwandeln, z. B. Telefone. Für die Übertragung des Signals wird ein Kanal (*Channel*) benötigt. Es gibt Weiterentwicklungen des Modells, z. B. [22], die Grundannahmen des Modells haben jedoch bis heute Bestand und bieten eine gute Grundlage, um die Kommunikation zwischen Menschen zu beschreiben [23]. Das Modell geht jedoch nicht auf den Inhalt und die Bedeutung der übermittelten Botschaften ein. Ebenso wird nicht ausgeführt, wie der Mensch die Nachrichten versteht, sondern es werden nur die technischen Aspekte beschrieben. Was im menschlichen Gehirn passiert, um die Nachrichten zu bilden bzw. die Nachrichten zu verstehen, wird nicht betrachtet. Der Aspekt der semantischen Inhalte bleibt also außen vor. Das Forschungsgebiet, das sich mit der Wissenschaft der Sprachen befasst, wird als *Linguistik* oder *Sprachwissenschaft* bezeichnet [24]. Das folgende Kapitel gibt einen kurzen Überblick über dieses Forschungsgebiet und aktuelle Modelle.

## 2.2 Linguistik

In dieser Arbeit wird kein vollständiger Überblick über historische und aktuelle Theorien innerhalb der Linguistik gegeben, sondern lediglich Konzepte vorgestellt, welche die Grundlage für weiterführende Arbeiten bilden, die wiederum die Basis für diese Arbeit darstellen.

Die Linguistik lässt sich in verschiedene Teildisziplinen unterteilen. Innerhalb des Forschungsbereichs besteht jedoch keine vollständige Übereinstimmung bzgl. der einzelnen Kategorien, sodass hier leichte Unterschiede entstehen. Beispiele für eine Aufteilung sind in folgender Tabelle 2.1 gegeben.

Tabelle 2.1: Teilbereiche der Linguistik

Harm [25]	Carstensen et al. [26]
Lexikologie	-
Phonologie	Phonetik/ Phonologie
Morphologie	Morphologie
Syntax	Syntax
Textlinguistik	-
Semantik	Semantik
Pragmatik	Pragmantik

Die in [26] aufgeführten Teilgebiete der Linguistik werden auch als *grammatische Teiltheorien* bezeichnet [27]. Im Folgenden sind die Definitionen der einzelnen Teilgebiete nach [27] aufgeführt:

- Phonologie: In der Phonologie werden die Lautsysteme einer Sprache untersucht.
- Morphologie: In der Morphologie werden die kleinsten bedeutungstragenden Elemente einer Sprache untersucht.
- Syntax: In der Syntax wird der zulässige, strukturelle Aufbau von Sätzen untersucht.
- Semantik: In der Semantik wird die Bedeutung von Wörtern, Sätzen und Texten untersucht.
- Pragmatik: In der Pragmatik wird der Zweck von Sprache innerhalb des jeweiligen Kontextes untersucht.

Die einzelnen Teilgebiete bilden zusammen die Grundlage für Modelle, welche die Frage beantworten sollen, wie der Mensch Sprache versteht. Moderne Ansätze werden als *strukturelle Linguistik* bezeichnet und haben ihren Ursprung in Ferdinand de Saussures Werk „Cours de linguistique générale“ [28, 29]. Kern der von de Saussure eingeführten strukturellen Linguistik ist die Abkehr von der Untersuchung der Sprache als Ansammlung von Äußerungen hin zur Analyse der Sprache als System von Elementen und deren Beziehungen untereinander. De Saussure unterscheidet dabei zwischen der Sprache als System (*Langue*) und dem Sprechen (*Parole*). Die Unterscheidung zwischen der Sprache als System und dem Sprechen ermöglicht eine losgelöste Untersuchung der Sprache. De Saussure unterscheidet in seinem Werk zwischen der Inhaltsseite (*Signifié*) und der Ausdrucksseite (*Signifiant*) von Zeichen. Die Inhaltsseite eines Zeichens repräsentiert die Vorstellung, die

im Gehirn entsteht, wenn man ein bestimmtes Wort hört, die Ausdrucksseite steht für die Laute oder Graphen, die man beim Sprechen oder Schreiben dieses Wortes erzeugt [28]. Die Kernaspekte der Arbeiten de Saussures, die Untersuchung der Sprache als System von Elementen und Beziehungen, sowie die Unterscheidung zwischen Inhalts- und Ausdrucksseite, bilden die Grundlage des *Strukturalismus*. Diese Ideen wurden in den folgenden Jahren in der *Prager- und Kopenhagener Schule* aufgegriffen und weiterentwickelt [30]. In Abgrenzung zum Strukturalismus verfasste Noam Chomsky in den 50er und 60er Jahren mehrere Arbeiten, deren Kernaussagen heute unter den Begriffen *generative Grammatik* und *Transformationsgrammatik* bekannt sind.

Chomskys generative Grammatik beschreibt die Fähigkeit des Menschen, Sprache zu erwerben und zu produzieren. Nach Chomsky verfügt der Mensch über angeborene kognitive Strukturen, die genetisch vererbt werden. Diese angeborenen Strukturen bilden die Grundlage der Grammatik und Syntax jeder Sprache. Chomsky geht davon aus, dass es einen Spracherwerbsmechanismus gibt, den so genannten *Language Acquisition Device (LAC)*, der zumindest die Grundprinzipien der Sprachkompetenz des Sprechers bestimmt [31]. Zwar lernen Kinder im Zuge des Spracherwerbs die einzelnen Wörter, wie diese jedoch zu grammatikalisch korrekten Wörtern zusammengefügt werden, ist genetisch vererbt. Ziel der generativen Grammatik ist es, eine universelle Grammatik zu finden, die für möglichst viele Sprachen gilt und beschreibt, wie ein Sprecher einer beliebigen Sprache mit einer endlichen Anzahl von Regeln eine unendliche Anzahl von Sätzen produzieren kann. Korrekte Sätze werden durch das LAC mittels der *Tiefen-* und *Oberflächenstruktur*, sowie der *Transformationskomponente* gebildet [32]. Nach Chomsky beschreibt die Oberflächenstruktur eines Satzes die Reihenfolge und Anordnung der Wörter und Satzzeichen in der tatsächlichen Äußerung, die wir hören oder lesen können. Die Tiefenstruktur hingegen bezieht sich auf die zugrundeliegende grammatikalische Struktur eines Satzes, die nicht immer direkt sichtbar ist. Die Tiefenstruktur beschreibt die Abfolge von Wörtern und Satzteilen, wie sie in unserem mentalen Sprachmodell gespeichert ist. Die Transformationskomponente wandelt mittels Transformationsregeln die Tiefenstruktur in die Oberflächenstruktur um [32]. Ein wohlgeformter Satz ist syntaktisch und semantisch korrekt und die Oberflächenstruktur spiegelt die Tiefenstruktur ausreichend wider. Die Aufteilung in Oberflächen- und Tiefenstruktur erinnert hierbei an die Aufteilung der Sprache de Saussure's in die *Langue*, Sprache als System, und *Parole*, den gesprochenen Wörtern. Die *Langue* ist hierbei ähnlich dem Konzept der Tiefenstruktur, während die *Parole* die Oberflächenstruktur widerspiegelt. Um die Unterschiede zwischen dem tatsächlichen Sprachgebrauch (Oberflächenstruktur) und dem zugrunde liegende Sprachverständnis (Tiefenstruktur und Transformationsregeln) zu beschreiben, führte Chomsky die Begriffe *Performanz* und *Kompetenz* ein. Kompetenz bezieht sich auf das Sprachwissen eines Sprechers, einschließlich der Fähigkeit, eine unbegrenzte Anzahl von Sätzen zu bilden und zu verstehen. Chomsky argumentiert, dass Sprachkompetenz durch eine angeborene universelle Grammatik ermöglicht wird. Perfor-

manz hingegen bezieht sich auf den tatsächlichen Sprachgebrauch und die Faktoren, die ihn beeinflussen, wie z. B. Gedächtnisbeschränkungen oder die Notwendigkeit, schnell zu sprechen [32]. Performanzfaktoren können die Sprachproduktion und das Sprachverstehen beeinträchtigen, indem sie Fehlerquellen und Einschränkungen verursachen, die nicht in der Sprachkompetenz des Sprechers begründet sind. Chomsky betont, wie wichtig es ist, zwischen Kompetenz und Performanz zu unterscheiden, da Performanzprobleme nicht unbedingt auf mangelnde Sprachkompetenz zurückzuführen sind.

Die Arbeiten von Chomsky hatten großen Einfluss auf andere Teilgebiete der Linguistik, wie z. B. die *Neurolinguistik* und die *Computerlinguistik*. Das folgende Kapitel gibt einen Überblick über die Geschichte der Computerlinguistik sowie über aktuelle Methoden und Modelle.

## 2.3 Computerlinguistik

Ziel der Computerlinguistik ist es, Computer in die Lage zu versetzen, gesprochene und geschriebene menschliche Sprache zu verstehen [26]. In der Computerlinguistik werden linguistische und informatische Methoden, insbesondere der KI, zur Verarbeitung von Sprache eingesetzt [33]. Der Einsatz von Methoden der KI zur Verarbeitung von Sprache wird als NLP bezeichnet [26, 33]. Dieser Begriff hat sich in den letzten Jahren etabliert und wird daher auch in dieser Arbeit verwendet, wenn es um den Einsatz von Methoden der KI im Bereich der Computerlinguistik geht.

Die Ursprünge der Computerlinguistik reichen bis in die 40er und 50er Jahre des letzten Jahrhunderts zurück. Schon früh zeigte sich, dass zwei unterschiedliche Ansätze verfolgt wurden, um das Gebiet der Computerlinguistik zu bearbeiten: Zum einen wurden stochastische Verfahren, zum anderen symbolische Ansätze eingesetzt. In den folgenden Jahrzehnten wurden insbesondere die symbolischen Ansätze untersucht, während stochastische Verfahren als nicht geeignet für die Modellierung von Sprache angesehen wurden [26]. Ein Grund dafür waren insbesondere Veröffentlichungen von Chomsky, in denen er die These aufstellte, dass stochastische Ansätze nicht in der Lage seien, Sprache hinreichend genau zu modellieren, da sie die zugrunde liegende Tiefenstruktur der Sprache nicht verstünden [31]. Unter dem großen Einfluss von Chomsky wandten sich viele Linguisten symbolischen statt stochastischen Methoden zu [26, 33]. Außerdem standen in der Mitte des letzten Jahrhunderts nicht genügend Rechenleistung und Daten zur Verfügung, um stochastische Modelle von ausreichender Größe zu trainieren. Diese Entwicklung hat sich in den letzten Jahren jedoch umgekehrt. Steigende Rechenleistung [34], zunehmende Verfügbarkeit von Daten in Kombination mit neuen Möglichkeiten diese zu beziehen (bspw. [35]) und neue Entwicklungen hinsichtlich entwickelter Architekturen (bspw. [36]) haben dazu geführt,

dass stochastische Ansätze seit Beginn der 2000er wieder vermehrt in den Fokus gerückt sind. Die entwickelte Methode dieser Arbeit basiert ebenfalls auf stochastischen Ansätzen, weshalb die hierfür relevanten Grundlagen in den folgenden Kapiteln beschrieben werden.

### 2.3.1 Stochastische Ansätze

Im Jahr 1950 wurde der Aufsatz „Computing Machinery and Intelligence“ von Alan Turing veröffentlicht (1). In diesem Aufsatz wurde von Turing die Frage aufgeworfen, wie man feststellen kann, ob ein Programm intelligent ist. Um diese Frage zu beantworten wurde der *Turing-Test* entworfen. In diesem Test kommuniziert eine Testperson mit zwei Partnern: einer weiteren Person und einem Computer. Kann die Testperson nicht unterscheiden, wer von den beiden Partnern der Mensch ist, wird die Maschine als intelligent bezeichnet [37]. Der eigentliche Begriff der KI wurde das erste Mal auf der Dartmouth Conference 1956 verwendet (2) [38]. 1966 definierte einer der Initiatoren der Konferenz, Marvin Minsky, KI wie folgt (3):

Artificial Intelligence is the science of making machines do things that would require intelligence if done by men.

Diese drei Ereignisse gelten heute weithin als die Geburtsstunde der KI. Seitdem wurde dieses Feld weiter erforscht und verschiedene Ansätze entwickelt, wie KI erreicht werden kann. Ein Teilgebiet der KI ist das *maschinelle Lernen (ML)*. Dieses wird verwendet, um Maschinen in die Lage zu versetzen, in vorliegenden Daten Muster und Zusammenhänge zu erkennen und auf dieser Basis Vorhersagen zu treffen [39]. Eine häufig verwendete Architektur sind *Künstliche Neuronale Netze (KNN)*. KNN bestehen aus künstlichen Neuronen und sind den Neuronen in einem Nervensystem nachempfunden. Die einzelnen Neuronen eines KNN sind in verschiedenen Ebenen (*Layern*) angeordnet und miteinander verbunden. Die Layer werden in *Input-, Hidden- und Output-Layer* unterschieden. In den Input-Layer werden die zur Verfügung stehenden Daten eingelesen, im Output-Layer wird eine Vorhersage auf Basis dieser Daten ausgegeben. Der Hidden-Layer liegt zwischen dem Input- und Output-Layer des KNN und gibt die Informationen weiter. Wird in einem KNN mehr als ein Hidden-Layer verwendet, wird dieses üblicherweise als ein *Deep-Neural-Network* bezeichnet [40]. Je mehr Hidden-Layer es in einem Netzwerk gibt, umso tiefer ist dieses. Aus diesem Grund spricht man in diesem Zusammenhang von *Deep Learning* [41].

Seit 2001 haben auf KNN basierende Methoden Einzug in den Bereich des NLP erhalten, welche die Basis für die heutigen Durchbrüche darstellen. Erste Ansätze verwendeten ein einfaches KNN für die Aufgabe des *Language-Modeling*. Hierbei wird das KNN verwendet, um das nächste Wort einer Sequenz auf Basis der vorherigen Wörter der Größe  $n$

vorherzusagen [42]. Das Ergebnis dieses Trainings sind Vektoren, welche Wörter so repräsentieren, dass sie semantische Bedeutungen und Beziehungen der Wörter enthalten. Diese sogenannten *Word-Embeddings (WE)s* können bspw. verwendet werden, um zu erkennen ob ein Wortpaar wie Berlin und Deutschland die gleiche Beziehung zueinander hat wie Madrid und Spanien. Diese auf großen Mengen Text vortrainierten WEs konnten dann in verschiedenen NLP-Aufgabenfeldern, wie bspw. *Text-Classification (TC)*, eingesetzt werden [43, 44]. Auf dieser Basis trainierte WEs ordnen jedoch jedem Wort nur eine Bedeutung zu. Die Bedeutungen des Wortes Bank, die je nach Kontext variieren können, werden mit dieser Form der WEs nicht abgebildet.

Diese Problematik konnte behoben werden, in dem nicht mehr nur das einzelne Wort zur Erstellung der WEs betrachtet wurde, sondern die einzelnen WEs als eine Funktion des gesamten Satzes gebildet wurden [45, 46]. Diese kontextualisierten Wortdarstellungen wurden auf Basis von bi-direktionalen *Long-Short-Term-Memory (LSTM)*-Netzen [47] erstellt und haben zu weiteren Verbesserungen im Bereich des Language-Modeling geführt. Die vortrainierten WEs wurden in Aufgaben, wie der TC, jedoch weiterhin nur als fixe Parameter verwendet, was als sogenannter *Feature-based-Approach* bezeichnet wird. Um die WEs und somit die Bedeutung der Worte in der Aufgabe selbst weiter anzupassen, wurde die Methode des *Finetuning* entwickelt [48]. Diese Art des Trainings, *Pretraining* auf allgemeinen Texten und *Finetuning* auf spezifischen Datensätzen, wird als *Transfer-Learning* bezeichnet [49] und ist die Basis für die heute führenden Modelle im Bereich NLP-Bereich. Während sich die generelle Art des Transfer-Learning nicht verändert hat, wurden die Modelle jedoch weiterentwickelt. Hierbei wurden die LSTM-Modelle durch Modelle ersetzt, welche auf der *Transformer-Architektur* basieren [36].

### 2.3.2 Transformer Architektur

Transformer sind eine Erweiterung der *Sequence-to-Sequence (seq2seq)* Modelle. Klassische seq2seq-Modelle wurden entwickelt, um Sequenzen (Sätze) vom Typ A in Sequenzen vom Typ B umzuwandeln. Diese werden vor allem in der automatisierten Übersetzung eingesetzt. Diese ursprünglichen seq2seq-Modelle werden zu der Familie der *Encoder-Decoder-Modelle* gezählt, wobei sowohl der *Encoder*, als auch der *Decoder* auf *Recurrent-Neural-Networks (RNN)s* basieren. Hierbei bildet der Encoder eine Eingangssequenz als einen Vektor ab, welchen der Decoder übernimmt, und als eine Ausgangssequenz übergibt. Beispiele für solche Modelle sind in [50–52] gegeben. Die Schwächen dieser Modelle liegen in der Abbildung langer Abhängigkeiten zwischen einzelnen Wörtern eines Satzes und darin, dass die Verwendung von RNNs und die damit verbundene Verarbeitung sequentieller Daten eine Parallelisierung verhindert [36].

An diesen Punkten setzt das Transformer-Modell an. Das Transformer-Modell setzt sich ebenfalls aus Encoder- und Decoder-Blöcken zusammen. Innerhalb eines Transformer-Modells können mehrere dieser Blöcke übereinandergestapelt werden. Die Kombination eines einzelnen Encoder-Blocks (links) und Decoder-Blocks (rechts) ist in folgender Abbildung 2.1 dargestellt. Da die Transformer-Architektur die Basis für die Modelle dieser Arbeit bildet, wird sie näher beschrieben.

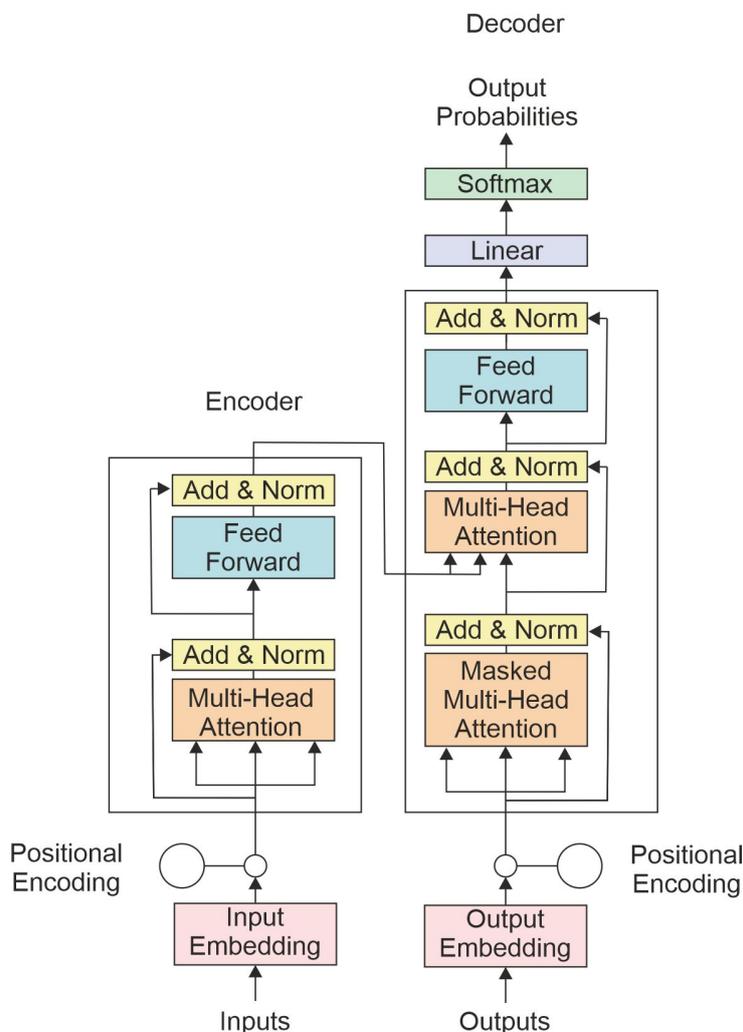


Abbildung 2.1: Transformer Architektur nach [36]

Jeder Encoder-Block setzt sich aus zwei Layern zusammen: Einem Layer, welcher als *Multi-Head-Self-Attention* bezeichnet wird, und einem einfachen Feed-Forward-Network (FFN) [36]. Im Training werden Satz-Paare an den Encoder und Decoder übergeben. Beispielsweise wird an den Encoder ein deutscher Satz, an den Decoder die Übersetzung des Satzes in Englisch übergeben. Die Embeddings des deutschen Satzes werden an den untersten Encoder-Block übergeben. Dort durchlaufen sie den Attention- und FFN-Layer und werden an den nächsten Block übergeben. Die Ergebnisse des obersten Encoder-Blocks werden an jeden Decoder-Block übergeben. Jeder Decoder-Block setzt sich wiederum aus drei Layern zusammen: Zusätzlich zu den gleichen Layern, aus denen sich der Encoder-

Block zusammensetzt, wird ein weiterer Layer eingeführt, welcher Self-Attention auf den Ausgang des Encoder-Layers ausführt. Der Attention-Mechanismus wird eingeführt, um die Abhängigkeit einzelner Wörter einer Sequenz von anderen Wörtern darzustellen [36]. Im Transformer-Modell wird die Self-Attention mehrmals durchgeführt, was zu der Bezeichnung Multi-Head-Attention führt. Der Ausgang eines Decoder-Blocks wird an den nächsthöheren Block übergeben. Der letzte Block gibt dann mittels einer *Softmax-Funktion* die Wahrscheinlichkeit für das nächste Wort der Sequenz aus.

### 2.3.3 Pretrained Language Models

Die ursprüngliche Transformer-Architektur setzt sich aus einem Encoder- und Decoder-Teil zusammen und wurde für seq2seq-Anwendungen (bspw. Übersetzungen) entworfen. Um die Vorzüge dieser Architektur (Parallelismus, Self-Attention) auch für andere Anwendungen einsetzen zu können, werden für State of the Art Modelle entweder der Encoder- [53–58] oder Decoder-Teil [59–61] verwendet und als *Pretrained-Language-Models (PLM)s* bezeichnet. Sowohl Encoder- als auch Decoder-Modelle werden nach dem Prinzip des Transfer-Learning trainiert: Pretraining auf allgemeinen Texten (bspw. [35]), mit anschließendem Finetuning auf eine spezifische Aufgabe. Wenn das Finetuning eines PLM für eine bestimmte Aufgabe durchgeführt wurde, wird es nicht mehr als PLM, sondern als Language-Model (LM) bezeichnet. Durch ihre unterschiedliche Architektur verläuft das Pretraining der Encoder- und Decoder-Modelle jedoch unterschiedlich.

Das Pretraining von Encoder-Modellen wird als *Masked-Language-Modeling (MLM)* bezeichnet. Die *Tokens* einzelner Textsequenzen werden parallel in das Modell eingespeist, so dass das Modell gleichzeitig auf alle Tokens zugreifen kann. Tokens repräsentieren Wörter oder Teile von Wörtern [33]. Um das Modell zu trainieren, werden einzelne Tokens, die das Modell vorhersagen soll, maskiert. Dadurch lernt das Modell, in welcher Beziehung die Tokens zueinander stehen [54]. Für das Pretraining der Modelle werden keine gelabelten Daten benötigt, da die Tokens vor jedem Durchlauf zufällig maskiert werden. Nachdem das Modell ein maskiertes Token vorhergesagt hat, kann es sich selbst überprüfen, indem die Maske entfernt wird und eine Überprüfung zwischen dem vorhergesagten und dem tatsächlichen Token durchgeführt wird. Aus diesem Grund wird diese Art des Trainings als *Self-Supervised* bezeichnet. Encoder-Modelle eignen sich besonders gut für NLP-Anwendungen, welche ein Verständnis des gesamten Satzes benötigen (bspw. TC, *Paraphrase-Identification (PI)*). Für das Finetuning der Modelle werden gelabelte Datensätze benötigt, welche speziell für eine der NLP-Anwendungen entwickelt werden.

Das Pretraining von Decoder-Modellen wird als *Casual-Language-Modeling (CLM)* bezeichnet. Decoder Modelle werden prinzipiell auf den gleichen Texten trainiert wie Encoder-Modelle. Durch ihre Architektur haben sie jedoch keinen Zugriff auf alle Tokens der

Eingangssequenz. Stattdessen werden die Tokens nacheinander an das Modell übergeben, wobei das Modell versucht auf Basis der vorherigen Tokens das nächste Token vorherzusagen [60]. Decoder-Modelle werden insbesondere für Text generierende Aufgaben eingesetzt.

Die NLP-Anwendungsfälle in denen die Modelle dieser Arbeit eingesetzt werden, sind TC und PI.

### 2.3.4 Text Classification

Der Bereich der TC beschäftigt sich mit der Klassifizierung von Texten. Das Ziel ist es, Texte zu verstehen und ihnen zuvor definierte Klassen zuzuordnen. Dabei handelt es sich um eine typische Anwendung von *Supervised-Learning*. Texte mit zugeordneten Labels dienen als Trainingsdaten, anhand derer der verwendete Algorithmus lernt, welche Art von Text welchem Label zugeordnet werden soll. Ein Anwendungsbereich ist beispielsweise die Gefühlsanalyse, die dazu eingesetzt werden kann festzustellen, ob Texte einen positiven oder negativen Ton haben [33].

Für das Finetuning von PLMs werden *Hyperparameter* eingesetzt, welche Auswirkungen auf die Leistung der Modelle haben. Die richtige Wahl der Hyperparameter wird als *Hyperparameter Optimierung* bezeichnet [62]. Der folgende Abschnitt beschreibt Hyperparameter, welche in dieser Arbeit und im Bereich der TC eingesetzt werden.

Beim Finetuning eines PLM werden die Trainingsdaten innerhalb von *Batches* an das Modell übergeben. Die *Batch-Size* gibt hierbei die Anzahl der Trainingsdaten an, die für jeden Optimierungsschritt gemeinsam verarbeitet werden. Die Größe der Batch-Size kann theoretisch von einem Trainingsbeispiel bis hin zum gesamten Trainingsdatensatz variiert werden. In der Praxis ist sie jedoch durch die eingesetzte Hardware limitiert. Durch den Einsatz des *Gradient-Accumulation-Verfahrens* kann die effektive Batch-Size jedoch erhöht werden. Beim Gradient-Accumulation-Verfahren werden die Gradienten der einzelnen Batches zwischengespeichert, um eine Aktualisierung der Gewichte basierend auf allen zwischengespeicherten Gradienten erst dann durchzuführen, wenn die definierte Zahl der Gradient-Accumulation erreicht ist [63]. Die *Anzahl der Epochen* gibt an wie oft der gesamte Trainingsdatensatz das Modell während des Trainings durchläuft. Wird eine zu hohe Anzahl an Epochen gewählt, führt dies im Allgemeinen zu einer Überanpassung (*Overfitting*) des Modells auf die Trainingsbeispiele, eine zu geringe Anzahl führt zu einer Unteranpassung (*Underfitting*) [64]. Während des Finetuning wird mittels einer *Verlustfunktion* der Verlust (*Loss*) eines Modells berechnet. Nach jedem Durchlauf wird dieser bestimmt und auf Basis dessen die Parameter des Modells für den nächsten Trainingsdurchlauf optimiert, sodass das Modell im nächsten Durchlauf eine bessere Vorhersage treffen kann. Je niedriger der Verlust

am Ende des Finetuning, desto besser sind die Vorhersagen des Modells [40]. Im Bereich TC wird für Klassifizierungen mit nur einem wahren Label (*Single-Label*) der *Cross-Entropy-Loss (CEL)* verwendet. Für ein Klassifizierungsproblem mit  $C$  Klassen wird die Differenz zwischen den tatsächlichen Klassen ( $p$ ) und den vorhergesagten Wahrscheinlichkeiten ( $q$ ) berechnet.

$$L(p_i, q_i) = - \sum_{i=1}^C p_i \cdot \log(q_i) \quad (2.1)$$

Um die trainierten Modelle bewerten zu können, werden verschiedene Metriken eingesetzt. Als Bewertungsmetrik im Bereich TC wird häufig die Genauigkeit (*Accuracy*) verwendet. Die Genauigkeit stellt die Anzahl der korrekten Vorhersagen der Gesamtzahl der Vorhersagen gegenüber [65].

$$\text{Genauigkeit} = \frac{\text{Anzahl korrekter Vorhersagen}}{\text{Gesamtzahl der Vorhersagen}} \quad (2.2)$$

Eine weitere Bewertungsmetrik ist der *F1-Score*. Der F1-Score ist ein Maß für die Leistung eines Klassifikationsmodells und berechnet sich aus dem harmonischen Mittelwert von Präzision (*Precision*) und Sensitivität (*Recall*). Durch die Kombination aus Präzision und Sensitivität kann der F1-Score gut für unausgeglichene Datensätze verwendet werden [65].

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.3)$$

Die Präzision gibt das Verhältnis der korrekten positiven Vorhersagen (*True-Positives*) zur Gesamtzahl der positiven Vorhersagen (*True-Positives* und *False-Positives*) an, während die Sensitivität das Verhältnis der korrekten positiven Vorhersagen zur Gesamtzahl der tatsächlich positiven Fälle (*True-Positives* und *False-Negatives*) repräsentiert [65].

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2.4)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2.5)$$

Bei einer Klassifizierung mit mehreren möglichen Labeln, muss der gewichtete Durchschnitt der F1-Scores aller Klassen gebildet werden, wobei der F1-Score jeder Klasse mit der Häufigkeit der Klasse im Datensatz gewichtet wird ( $w$ ).

$$\text{Gewichteter F1-Score} = \sum_{i=1}^N w_i \cdot F1 \quad (2.6)$$

### 2.3.5 Paraphrase Identification

Die Fähigkeit, semantisch gleiche Sätze zu erkennen, wird als PI bezeichnet [66]. Ein Beispiel dafür ist der Einsatz des Datensatzes Quora Question Pairs (QQP), für den vorhergesagt werden soll, ob Fragen die gleiche Bedeutung haben oder nicht [67]. Das Ziel in diesem Fall ist, mehrfaches Stellen der gleichen Frage in einem Forum zu verhindern. Dafür müssen unterschiedlich formulierte Sätze auf ihre semantische Äquivalenz geprüft werden, beispielsweise: „Sind Machine Learning und Künstliche Intelligenz das Gleiche?“ und „Gibt es einen Unterschied zwischen KI und ML?“. Um die Ähnlichkeit zweier Sätze mittels NLP bestimmen zu können, werden diese als Embeddings, in Form von Vektoren, dargestellt. Im Bereich PI werden zur Berechnung der Ähnlichkeit zweier Vektoren unter anderem die *Kosinus-Ähnlichkeit* und der *euklidische Abstand* verwendet. Mittels der Kosinus-Ähnlichkeit wird der Kosinus des Winkels zwischen zwei Vektoren gemessen. Hierbei wird das Skalarprodukt  $A \cdot B$ , sowie die Länge (euklidische Norm) der Vektoren A und B berechnet [68].

$$\text{Kosinus-Ähnlichkeit} = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \times \|\mathbf{B}\|} \quad (2.7)$$

Der euklidische Abstand berechnet die Wurzel aus der Summe der Quadrate der Differenzen zwischen den Punkten der beiden Vektoren [69].

$$\text{Euklidischer Abstand} = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} \quad (2.8)$$

Als Verlustfunktion wird im Bereich PI unter anderem der *Multiple-negatives-Ranking-Loss (MNRL)* verwendet [70]. Beim MNRL wird die Kosinus-Ähnlichkeit in Kombination mit dem CEL verwendet. Die Kosinus-Ähnlichkeit wird hierbei für jedes Satzpaar (sowohl positive als auch negative Paare berechnet). Die Ergebnisse werden mittels CEL mit den wahren Labeln verglichen und als Verlustfunktion verwendet.

### 2.3.6 Wissensgraphen in Kombination mit Language Models

Mithilfe der Transformer-Architektur und darauf basierender PLMs können Informationen aus Text extrahiert werden. Da Text jedoch durch unscharfe Formulierungen gegebenenfalls nicht präzise genug ist, um formale Beziehungen exakt auszudrücken, werden andere

Konzepte hierzu verwendet. Um das Wissen einer Domäne sowie Konzepte, Kategorien und deren Beziehungen formal darzustellen, werden Wissensgraphen (*Knowledge-Graphs (KG)s*) eingesetzt. Ein KG bildet das Wissen über reale Entitäten innerhalb einer bestimmten Domäne strukturiert in Form eines Graphen ab [71]. Sie können in Form von Tripel dargestellt werden, welche sich aus Subjekt (Entität), Prädikat (Relation) und Objekt (Entität) zusammensetzen. Als Basis für KGs können *Ontologien* dienen, welche die Klassen einer Domäne und deren mögliche Beziehungen untereinander beschreiben [72]. Das Wissen der Gebäudetechnik wird abhängig von dem Anwendungsfall in unterschiedlichen Ontologien dokumentiert. Zwei weit verbreitete Ontologien sind das *Brick Schema* [73] und *Project Haystack (PH)* [74]. Diese ermöglichen die Abbildung des Betriebs von kommerzieller Gebäudeautomation (GA) und TMon [75]. Dazu standardisieren sie die Repräsentation von physischen, logischen und virtuellen Assets in Gebäuden und deren Beziehung zueinander [73]. Nachfolgend werden die Ontologien Brick und PH als KGs interpretiert. Die Klassen der KGs werden als Entitäten bezeichnet. Ähnlich wie die Token eines Textes kann das in KGs repräsentierte Wissen durch Embeddings dargestellt werden. Die Embeddings unterteilen sich in *Entity-Embeddings (EE)s* und *Relation-Embeddings (RE)s*, die die Bedeutung der Entitäten respektive der Relationen abbilden [76]. Diese Embeddings können sowohl statisch als auch dynamisch sein. Basierend auf den Tripel der KGs können statische EEs mit Hilfe verschiedener Modelle erlernt werden [77]. Dynamische EEs hingegen können durch die gemeinsame Verarbeitung von Text und KGs, beispielsweise durch das kontextualisierte Knowledge-LM *CoLAKE* [78], trainiert werden. Dabei wird das Pretraining im Vergleich zum Pretraining des reinen PLM um die Methoden des *Masked-Entity-Modeling* und *Masked-Relation-Modeling* erweitert. Es werden also sowohl Token als auch Entitäten und Relationen maskiert. Dadurch wird die Semantik der Token in Form der WEs um die in EEs und REs enthaltenen Informationen aus dem KG erweitert [78].

## 2.4 Interoperabilität

In Kapitel 2.1 wird die Interaktion und Kommunikation zwischen Menschen beschrieben. Die Interaktion zwischen Menschen ist durch wechselseitige Beeinflussung geprägt, was zu Veränderungen im Verhalten führt. Die Kommunikation bildet die Grundlage für die Interaktion, wobei der Sender Informationen an den Empfänger überträgt. Bei verbaler Kommunikation erfolgt dies durch Sprache. Die Grundlagen dieser Konzepte lassen sich auch auf die Interaktion technischer Systeme übertragen. Um dynamische Wertschöpfungsnetzwerke zu kreieren, müssen Systeme miteinander interagieren. Hierbei müssen sie Nachrichten austauschen, deren Inhalt interpretieren und darauf basierend Aktionen durchführen, die wiederum Einfluss auf andere Systeme haben. Die Interoperabilität von Systemen spielt hierbei eine Schlüsselrolle (vgl. Kapitel 1.1). Im folgenden Abschnitt wer-

den zunächst Definitionen des Begriffs Interoperabilität vorgestellt. Anschließend wird eine eigene Definition erarbeitet. Darüber hinaus werden verschiedene Konzepte zur Beschreibung von Interoperabilitätsebenen dargestellt und verglichen, um anschließend auf das für diese Arbeit wesentliche Konzept der semantischen Interoperabilität näher einzugehen.

### 2.4.1 Definition Interoperabilität

Für den Begriff Interoperabilität existieren verschiedene Definitionen, von denen nachfolgend einige aufgeführt sind.

- **ISO/IEC 22123-1 [79]**: „Ability for two or more systems or applications to exchange information and to mutually use the information that has been exchanged.“
- **Positionspapier Interoperabilität Plattform Industrie 4.0 [10]**: „Der Begriff beschreibt die Fähigkeit von Systemen, Geräten oder Anwendungen miteinander zu kommunizieren und nahtlos zusammen zuarbeiten.“
- **Peter Wegner [80]**: „Interoperability is the ability of two or more software components to cooperate despite differences in language, interface, and execution platform.“
- **Diedrich et al. [81]**: „Interoperabilität zwischen den Elementen eines Systems ist der Integrationsgrad dieser Elemente bewertet an dem Beitrag der Interaktionen zwischen Elementen an der(n) vom System zu erfüllenden Aufgabe(n).“
- **ChatGPT**: „Interoperability refers to the ability of different systems, technologies, or software applications to communicate, exchange data and work together effectively. It is the capability of diverse systems and organizations to interact and exchange information seamlessly in a coordinated manner without restrictions or limitations.[...] In summary, interoperability is the ability of different systems and technologies to work together and communicate with each other effectively, enabling the exchange of information and data seamlessly without any restrictions or limitations.“

Die in dieser Arbeit entwickelte Methode zur Erreichung semantischer Interoperabilität basiert auf Modellen aus dem Bereich des NLP. Um die Potentiale dieser Modelle aufzuzeigen, wird bei der Auflistung der Definitionen neben von Menschen erstellten Definitionen auch eine Definition des *Chatbots* „ChatGPT“ aufgeführt. Die verschiedenen Fragen und Antworten an den Chatbot sind in Anhang A zu finden.

Die Analyse der verschiedenen Definitionen des Begriffs Interoperabilität zeigt, dass sie im Wesentlichen die gleichen Aspekte behandeln. Zunächst geht es bei Interoperabilität um den Austausch von Informationen zwischen zwei oder mehreren Entitäten. In den

verschiedenen Definitionen können diese Entitäten Systeme, Anwendungen oder Geräte sein. Um Interoperabilität zwischen den Entitäten zu erreichen, muss der Informationsaustausch so gestaltet sein, dass die jeweils andere Entität die Informationen verarbeiten kann. Ist dies nicht der Fall, kann Interoperabilität nicht erreicht werden. Über den reinen Informationsaustausch hinaus müssen die Entitäten auch in der Lage sein, nahtlos zusammenzuarbeiten. Die nahtlose Zusammenarbeit wird weiterhin so beschrieben, dass sie ohne Einschränkungen erreicht werden soll, auch wenn unterschiedliche Sprachen, Schnittstellen oder Plattformen verwendet werden. Hervorzuheben ist hier insbesondere der Aspekt, dass Interoperabilität auch dann gewährleistet sein muss, wenn die Entitäten unterschiedliche Sprachen verwenden. Während die anderen Definitionen diesen Punkt nicht näher spezifizieren, wird er bei Wegner explizit adressiert. Zusammengefasst bedeutet dies, dass nicht nur homogene Systeme (Systeme, die die gleichen Modelle, Sprachen, Schnittstellen verwenden) interoperabel zusammenarbeiten sollen, sondern auch heterogene Systeme. Zusätzlich zu den oben genannten Punkten wird in [81] gefordert, dass Entitäten interoperabel interagieren sollen, um eine gemeinsame Aufgabe zu erfüllen. Die gemeinsame Aufgaben- bzw. Funktionserfüllung ist also das übergeordnete Ziel, das durch die interoperable Interaktion von Entitäten erreicht werden soll.

Für diese Arbeit wird eine eigene Definition von Interoperabilität erstellt:

Interoperabilität ist die Fähigkeit zweier oder mehrerer homogener oder heterogener Systeme, Informationen so auszutauschen, dass diese Systeme nahtlos und uneingeschränkt zusammenarbeiten können, um eine gemeinsame Aufgabe zu erfüllen.

Um Interoperabilität zu erreichen und die verschiedenen Aspekte der Interoperabilität besser zu bündeln, wird das Konzept der Interoperabilität in der Literatur in verschiedene Ebenen unterteilt.

### **2.4.2 Ebenen der Interoperabilität**

Im folgenden Abschnitt werden verschiedene Modelle betrachtet, welche Ebenen der Interoperabilität beschreiben.

#### **Sheth [82]**

Eine frühe Unterteilung des Begriffs Interoperabilität in verschiedene Ebenen findet sich in [82]. Hier wird Interoperabilität in die vier Ebenen System- (System), syntaktische- (Syntactic), strukturelle- (Structural) und semantische- (Semantic) Interoperabilität unterteilt.

Die Ebene der System-Interoperabilität umfasst die Nutzung derselben Netzinfrastruktur. Interoperabel interagierende Entitäten müssen dasselbe Netzwerk verwenden, um miteinander kommunizieren zu können. Syntaktische Interoperabilität beschreibt, dass Systeme die gleichen Datenformate als Grundlage für einen erfolgreichen Datenaustausch verwenden. Die Ebene der strukturellen Interoperabilität baut auf der syntaktischen Ebene auf. Um strukturelle Interoperabilität zu erreichen, müssen die Entitäten nicht nur die gleichen Formate verwenden. Sie müssen auch die gleichen Modelle innerhalb dieser Formate verwenden, um Daten automatisiert austauschen zu können. Die oberste Ebene, die semantische Interoperabilität, beschreibt, dass die von den Entitäten ausgetauschten Informationen von der jeweils anderen Entität verstanden werden müssen.

### **ISO/IEC 21823 [83]**

Die Normenreihe ISO/IEC 21823 behandelt unterschiedliche Aspekte der Interoperabilität von *Internet of Things (IoT)-Systemen*. Part 1 der Reihe beschreibt das Framework zur Umsetzung von Interoperabilitätsszenarien und behandelt hierbei die Aspekte, welche hierfür erfüllt sein müssen. Es werden die fünf Aspekte Transport- (Transport), syntaktische- (Syntactic), semantische- (Semantic), verhaltensbezogene- (Behavioural) und rechtliche- (Policy) Interoperabilität unterschieden. Die Transport-Interoperabilität bezieht sich dabei auf den Datentransfer zwischen Entitäten. Sie beschreibt, dass interoperabel interagierende Entitäten die gleiche Netzwerkinfrastruktur verwenden müssen. Dies beinhaltet sowohl die Verwendung des gleichen physikalischen Mediums, als auch die Verwendung der gleichen Protokolle. Kann dies nicht gewährleistet werden, müssen Gateways als Übersetzer eingesetzt werden. Syntaktische Interoperabilität bedeutet, dass Entitäten die gleiche Syntax verwenden müssen, um Interoperabilität zu gewährleisten. Syntax bedeutet in diesem Kontext, dass die Entitäten die gleichen Informationsmodelle und Formate verwenden. Semantische Interoperabilität beschreibt hier die Fähigkeit der interagierenden Entitäten, die ausgetauschten Informationen zu verstehen. Hierzu sollen semantische Modelle verwendet werden, welche beiden Entitäten bekannt sind. Die verhaltensbezogene Interoperabilität beschreibt die Schnittstellen, deren Operationen, sowie Ein- und Ausgänge, die von den Entitäten bereitgestellt werden. Sie beschreibt das zu erwartende Verhalten bei der Nutzung der von den Entitäten angebotenen Schnittstellen. Verhaltensbezogene Interoperabilität liegt vor, wenn das tatsächliche Ergebnis bei der Nutzung einer solchen Schnittstelle mit dem erwarteten Verhalten übereinstimmt. Der letzte Aspekt der Interoperabilität betrifft die rechtliche Interoperabilität. Diese ist erreicht, wenn zwei Entitäten innerhalb des geltenden Rechtsrahmens (sowohl staatlich als auch organisatorisch) miteinander interagieren.

**Noura et al. [84]**

In [84] wird eine Taxonomie für Interoperabilität zwischen IoT-Geräten aufgestellt und es werden Herausforderungen beschrieben, die gelöst werden müssen, um Interoperabilität zu erreichen. Die Taxonomie besteht aus fünf Ebenen: Geräte- (Device), Netzwerk- (Network), syntaktische- (Syntactical), semantische- (Semantic) und Plattform- (Platform) Interoperabilität. Geräte-Interoperabilität beschreibt, dass IoT-Geräte, die unterschiedliche Kommunikationsprotokolle verwenden, dennoch in der Lage sein sollen, ihre Informationen auszutauschen. Aufgrund der großen Anzahl unterschiedlicher IoT-Geräte ist auch mit mehreren Netzwerken innerhalb eines IoT-Systems zu rechnen. Netzwerk-Interoperabilität bedeutet, dass Geräte, welche unterschiedliche Netzwerke verwenden, dennoch Nachrichten miteinander austauschen können. Syntaktische Interoperabilität bezieht sich auf die Interoperabilität in Bezug auf die verwendeten Formate und Modelle. Um syntaktische Interoperabilität zu erreichen, müssen IoT-Geräte die gleichen Formate und Modelle verwenden oder es müssen Schnittstellen definiert werden, die verschiedene Modelle aufeinander abbilden. Die Ebene der semantischen Interoperabilität bezieht sich darauf, dass Entitäten die ausgetauschten Daten verstehen müssen. Werden nicht die gleichen semantischen Modelle verwendet, so dass semantische Inkompatibilität vorliegt, ist eine automatisierte Interaktion zwischen IoT-Geräten nicht möglich. Die letzte Ebene, die Plattform-Interoperabilität, befasst sich mit dem Aspekt, dass verschiedene IoT-Plattformen derzeit nicht miteinander kompatibel sind. Entwickler von Schnittstellen zwischen IoT-Geräten und IoT-Plattformen müssen die spezifischen *Programmierschnittstellen (API)s* der Plattformen kennen, um Geräte in diese integrieren zu können. Für eine vollständige Interoperabilität muss eine plattformübergreifende Interoperabilität geschaffen werden, die es ermöglicht, Geräte über die gleichen Schnittstellen in verschiedene Plattformen einzubinden.

**Tolk et Muguira [85]**

In [85] werden fünf Ebenen (0 - 4) der Interoperabilität beschrieben, wobei der Schwerpunkt auf den ausgetauschten Daten liegt: systemspezifische Daten (Ebene 0), dokumentierte Daten (1), abgestimmte statische Daten (2), abgestimmte dynamische Daten (3), harmonisierte Daten (4). Auf Ebene 0 kann keine Form der Interoperabilität zwischen Entitäten hergestellt werden, da die Daten nur innerhalb des eigenen Systems verwendet werden und aufgrund der proprietären Systeme kein Austausch zwischen den Entitäten stattfinden kann. Interoperabilität der Ebene 1 bedeutet, dass Entitäten ein gemeinsames Informationsmodell verwenden und ihre Daten über eine Schnittstelle zur Verfügung stellen. Stufe 2 setzt voraus, dass die Entitäten zusätzlich ein gemeinsames Vokabular verwenden, so dass die ausgetauschten Informationen eindeutig verstanden werden können. Ebene 3 der abgestimmten dynamischen Daten bedeutet, dass die Systeme nicht nur das gleiche

Informationsmodell und das gleiche Vokabular verwenden, sondern dass auch ihr Verhalten bekannt ist. Während die Ebenen 2 und 3 definieren, wie Daten zu modellieren sind und was sie bedeuten, wird auf dieser Ebene das Verhalten der Entitäten modelliert. Die letzte Ebene beschreibt die Interoperabilität durch harmonisierte Daten. Harmonisierte Daten bedeutet in diesem Zusammenhang, dass konzeptuelle Modelle für eine bestimmte Domäne existieren müssen. Konzeptuelle Modelle werden hier verwendet, um die reale Welt als Ganzes zu modellieren. Dieser Grad an Interoperabilität kann nach [85] nicht durch technische Lösungen erreicht werden. In Tabelle 2.2 sind die verschiedenen Ebenen der Interoperabilität gegenübergestellt.

Tabelle 2.2: Vergleich Definitionen Interoperabilitätsebenen

Interoperabilitätsebene	Sheth	ISO/IEC 21823-1	Noura et al.	Tolk et Mugui-ra
Ebene 5	-	Rechtlich	Plattform	Harmonisiert
Ebene 4	-	Verhaltensbezogen	-	Dynamisch abgestimmt
Ebene 3	Semantisch	Semantisch	Semantisch	Statisch abgestimmt
Ebene 2	Strukturell Syntaktisch	Syntaktisch	Syntaktisch	Dokumentiert
Ebene 1	System	Transport	Netzwerk Geräte	
Ebene 0	-	-	-	System-spezifisch

Fasst man die vier Klassifikationen zusammen, so ergeben sich sechs Ebenen der Interoperabilität. Dabei überschneiden sich die vier Klassifikationen größtenteils, teilweise sind die Ebenen der Einteilungen jedoch auch zusammengefasst. Die unterste Stufe wird nur in [85] beschrieben und besagt, dass keine Interoperabilität möglich ist. Die Entitäten verwenden hier proprietäre Systeme und die Daten werden nur innerhalb des eigenen Systems verwendet. Die Ebene 1 wird in allen Definitionen beschrieben, jedoch mit unterschiedlichen Namen bezeichnet. Inhaltlich beschreibt diese Ebene, dass die Systeme die gleiche Netzwerkinfrastruktur und die gleichen Kommunikationsprotokolle verwenden. Bei unterschiedlichen Protokollen können Gateways eingesetzt werden, um zwischen den Protokollen zu übersetzen. Ebene 2 wird üblicherweise als syntaktische Interoperabilität bezeichnet. Inhaltlich bedeutet dies, dass die übertragenen Daten (Ebene 1) die gleiche Struktur aufweisen und die gleichen Modelle verwendet werden. Ebene 3 umfasst die semantische Interoperabilität. Semantische Interoperabilität wird erreicht, wenn die interagierenden Entitäten die Bedeutung der ausgetauschten Informationen verstehen. Allen Definitionen ist gemeinsam, dass semantische Interoperabilität nur erreicht werden kann, wenn das gleiche Vokabular bzw. die gleichen semantischen Modelle verwendet werden. Ebene 4 wird nur in [83] und [85] beschrieben und behandelt die verhaltensbezo-

gene Interoperabilität. Wenn Entitäten auf dem gleichen Modell basieren (syntaktische Interoperabilität) und das gleiche Vokabular verwenden (semantische Interoperabilität), können die Entitäten theoretisch per *Plug-and-Play* ausgetauscht werden. Wenn sie sich jedoch intern unterschiedlich verhalten um eine Funktion zu erfüllen, kann dies dennoch zu Problemen in der Anwendung führen. Ebene 4 ist demnach erfüllt, wenn das Verhalten der Entitäten bekannt ist. Ebene 5 behandelt in allen Klassifikationen unterschiedliche Aspekte der Interoperabilität, zur besseren Übersicht sind diese jedoch in der Tabelle in einer Ebene zusammengefasst. Während hier in [83] rechtliche Aspekte betrachtet werden, behandelt [84] die Interoperabilität zwischen Plattformen und [85] konzeptuelle Modelle.

Zusammenfassend kann festgestellt werden, dass insbesondere die Ebenen 1 (Transport), 2 (Syntax) und 3 (Semantik) in allen vier Klassifikationen behandelt werden. Im Folgenden werden Einschätzungen zum aktuellen Stand der Erfüllung dieser drei Ebenen gegeben.

### 2.4.3 Aktueller Stand: Erfüllung der Interoperabilitätsebenen

Verschiedene Quellen analysieren den aktuellen Stand hinsichtlich der Erfüllung der verschiedenen Ebenen. Die erste der vier vorgestellten Klassifikationen wurde 1999 veröffentlicht [82]. In der Analyse der Erfüllung der Ebenen beschreibt Sheth, dass in den folgenden Jahren durch weitere Standardisierung die Anforderungen an Transport- und syntaktische Interoperabilität besser erfüllt werden als 1999. Weiterhin wird beschrieben, dass die zentralen Herausforderungen in der Lösung der semantischen Interoperabilität liegen werden [82]. Im gleichen Jahr wird in [86] beschrieben, dass erhebliche Fortschritte in den Bereichen syntaktische und Transport-Interoperabilität erzielt wurden, Lösungen für die semantische Interoperabilität derzeit jedoch nicht verfügbar sind. 2019 wurde eine Taxonomie für Interoperabilität entwickelt, verschiedene Ansätze zur Erreichung von Interoperabilität vorgestellt und offene Forschungspunkte beschrieben [84]. Die unteren Ebenen der Interoperabilität (Transport und Syntax) sind nach [84] im Jahr 2019 gut erfüllt. Lösungen für eine umfassende semantische Interoperabilität sind jedoch noch nicht verfügbar. Es wird daher gefordert, die Forschungsaktivitäten in diesem Bereich auszuweiten. In ISO/IEC 21823-1 wird ebenfalls der aktuelle Stand der Erfüllung der Interoperabilitätsebenen betrachtet. In der dortigen Analyse wird hervorgehoben, dass die Erreichung der semantischen Interoperabilität eine größere Schwierigkeit darstellt als die der Transport- und syntaktischen Interoperabilität [83]. Aufgrund der Vielzahl an heterogenen Vokabularen, die zwischen den Systemen verwendet werden, wird es als schwierig angesehen, Adapter zu entwickeln, die unterschiedliche Vokabulare aufeinander abbilden können.

Zusammenfassend kommen alle Analysen zu dem gleichen Ergebnis: Transport- und syntaktische Interoperabilität sind leichter zu erreichen als semantische Interoperabilität.

In den letzten Jahren wurde viel getan, um die notwendigen Technologien und Konzepte für diese beiden Ebenen weiterzuentwickeln, so dass die Entwicklung hier weit fortgeschritten ist. Im Vergleich dazu ist die semantische Interoperabilität schwieriger zu erreichen. Die Komplexität dieser Ebene führt dazu, dass die Entwicklungen hier noch nicht so weit fortgeschritten sind wie bei den beiden unteren Ebenen. In den kommenden Jahren müssen weitere Forschungsarbeiten durchgeführt und Lösungen entwickelt werden, um semantische Interoperabilität zu erreichen. Im folgenden Abschnitt werden der Begriff der semantischen Interoperabilität und grundlegende Lösungskonzepte vorgestellt.

### 2.4.4 Semantische Interoperabilität

Wie in Kapitel 2.4.1 wird zunächst der Begriff der semantischen Interoperabilität definiert. Im Folgenden sind einige Definitionen aufgeführt.

- **Sandra Heiler [87]**: „Interoperability among components of large-scale, distributed systems is the ability to exchange services and data with one another. [...] Semantic interoperability ensures that these exchanges make sense - that the requester and the provider have a common understanding of the “meanings” of the requested services and data.“
- **ISO/IEC 19941 [88]**: „Interoperability so that the meaning of the data model within the context of a subject area is understood by the participating systems.“
- **IEC Whitepaper Semantic Interoperability [89]**: „Semantic interoperability can be defined as the ability of two or more assets (e. g. agents, machines, systems) to exchange and understand each other’s data correctly.“
- **ChatGPT**: „Semantic interoperability refers to the ability of different computer systems or software applications to exchange and interpret data with a common understanding of the meaning of that data. This means that the systems can communicate effectively and accurately without the need for human intervention to translate or interpret the data.[...]“

Die älteste Definition stammt aus dem Jahr 1995 [87], die jüngste aus dem Jahr 2023 vom Chatbot „ChatGPT“. Obwohl fast 30 Jahre zwischen den verschiedenen Definitionen liegen, sind sich alle vier Definitionen sehr ähnlich. Im Zentrum aller Definitionen steht der Austausch (Exchange) von Daten und die Fähigkeit, die Bedeutung (Meaning) dieser Daten zu verstehen (Understand). Die Daten werden dabei zwischen zwei oder mehreren Systemen ausgetauscht. In der Definition von „ChatGPT“ wird im Gegensatz zu den anderen Definitionen noch der Mensch mit einbezogen: Ziel der semantischen Interoperabilität ist es, dass kein Mensch eingreifen muss, um zwischen den Systemen zu übersetzen. In den

einzelnen Definitionen wird jedoch nur gesagt, dass die ausgetauschten Daten vom jeweils anderen System verstanden werden müssen. Wie dieses Verständnis erreicht werden kann, wird in den Definitionen nicht beschrieben.

Wie ein gemeinsames Verständnis grundsätzlich erreicht werden kann, ist in [89] und [90] beschrieben. Grundsätzlich werden hier zwei gleiche Ansätze beschrieben. Im ersten Ansatz wird semantische Interoperabilität durch die Verwendung eines gemeinsamen Informationsmodells erreicht, in dem die Informationen semantisch einheitlich modelliert sind. Durch die Verwendung eines einheitlichen (standardisierten) semantischen Standards sind die Systeme in der Lage, die Daten des jeweils anderen Systems zu verstehen und zu interpretieren. Dieser Ansatz wird im Folgenden als *homogene Semantik* bezeichnet. Beim zweiten Ansatz können unterschiedliche semantische Beschreibungen der Informationen eines Informationsmodells verwendet werden. Damit Systeme die Informationen verstehen und interpretieren können, müssen diese jedoch aufeinander abgebildet werden. Eine standardisierte Semantik ist hier also nicht notwendig, jedoch müssen die verwendeten Beschreibungen aufeinander abgebildet werden. Dieser Ansatz wird im Folgenden als *heterogene Semantik* bezeichnet.

Interoperable Systeme bilden die Basis für verschiedene neue Lösungen und Anwendungen. In dieser Arbeit sollen solche Systeme implementiert werden, um Erkundungsmechanismen umzusetzen. Dies soll im Rahmen einer Monitoring-Anwendung für Anlagen der TGA erfolgen. Monitoring-Anwendungen sind ein wichtiges Werkzeug, um den Betriebszustand von Gebäuden zu überwachen und einen energieeffizienten und optimierten Betrieb zu gewährleisten. Aufgrund der Ziele der europäischen Klimapolitik wird der Einsatz von TMon-Anwendungen in Gebäuden in den nächsten Jahren weiter zunehmen.

## 2.5 Europäische Klimapolitik

Im Folgenden werden die europäischen Klimaziele und ihre Umsetzung in Form von Gesetzen, insbesondere innerhalb von Gebäuden, beschrieben.

### 2.5.1 Europäische Klimaziele

Das folgende Kapitel gibt einen Überblick über die Entstehung der internationalen Klimaziele und ihren aktuellen Stand innerhalb der Europäischen Union (EU) und Deutschlands. Die Begrenzung des durch menschliche Aktivitäten verursachten Klimawandels ist eine zentrale Herausforderung unserer Zeit. Um eine Begrenzung zu erreichen, müssen alle Branchen und Industrien Energie einsparen. Im Jahr 1992 wurde das erste offizielle internationale Abkommen verabschiedet, welches das Ziel hat, den Klimawandel zu

stoppen. In dem als „Rahmenübereinkommen der Vereinten Nationen über Klimaänderungen“ bezeichneten Abkommen wurden jedoch keine konkreten Grenzwerte festgelegt, an die sich die Staaten halten müssen. Es wird lediglich darauf hingewiesen, dass der Klimawandel ein ernstes Problem darstellt, das die Staaten zum Handeln zwingt [91]. In weiteren Abkommen, bspw. [92], wurden die klimapolitischen Ziele weiter geschärft. Die gegenwärtige Klimapolitik beruht hauptsächlich auf dem Abkommen von Paris [93]. In diesem Abkommen haben 197 Staaten festgehalten, dass die globale Erderwärmung auf möglichst 1,5°C auf jeden Fall deutlich unter 2°C im Vergleich zum vorindustriellen Zeitalter beschränkt werden soll. Der „EU Green Deal“, der auf Basis dieses Abkommens beschlossen wurde, zielt darauf ab, den Übergang zu nachhaltiger Energie und Umwelt zu fördern [94]. In diesem Abkommen wird festgehalten, dass die netto-Emissionen der EU bis 2050 auf null reduziert werden sollen, um so der erste klimaneutrale Kontinent zu werden. Ab 2050 soll dann eine jährliche negative Emissionsbilanz erreicht werden. Um dieses Ziel zu erreichen und gesetzlich zu verankern, wurde in diesem Beschluss angekündigt, dass bis 2020 der Entwurf eines europäischen Klimagesetzes fertiggestellt sein soll. Dieses Gesetz wurde mehrmals überarbeitet und im Juni 2021 endgültig angenommen [95]. In diesem Klimagesetz wurden bis 2050 mehrere Zwischenziele definiert, die erreicht werden sollen. Unter anderem sollen die Nettotreibhausgasemissionen im Jahr 2030 um mindestens 55% gegenüber dem Jahr 1990 gesenkt werden. Um dieses Ziel zu erreichen wurde das Paket „Fit for 55“ beschlossen [96]. Dieses Paket beinhaltet Vorschläge zur Aktualisierung aktueller EU-Rechtsvorschriften im Hinblick auf die Reduktion von Emissionen. Insgesamt werden zwölf Aktionsschwerpunkte definiert, die zur Erreichung des Ziels der Reduktion behandelt werden müssen. Bei der Erreichung der Klimaziele ist der Gebäudesektor ein Schlüsselfaktor. Der Betrieb von Wohn- und Nichtwohngebäuden spielt eine bedeutende Rolle, da er 36% des Endenergieverbrauchs und 28% der Treibhausgasemissionen in Deutschland verursacht [3]. Auf ganz Europa bezogen erhöhen sich diese Werte auf 40% des Energieverbrauchs und 36% der Treibhausgasemissionen [97]. Um den Energieverbrauch und die Emissionen im Gebäudesektor zu reduzieren, ist es notwendig, den Betrieb von Gebäuden energieeffizienter zu gestalten.

### 2.5.2 Maßnahmen zur Steigerung der Energieeffizienz

Neben dem europäischen Klimagesetz als Instrument der Gesetzgebung gewinnt die Finanzierung der Klimawende an Bedeutung. Die EU stellt Mittel zur Verfügung und animiert private Investoren, die notwendige Wende mitzufinanzieren [94]. Ziel ist es, das Finanzwesen der EU nachhaltiger zu gestalten. Zukünftige Investitionen sollen nach nachhaltigen Gesichtspunkten bewertet werden. Konkret sind diese ökologischer, sozialer und wirtschaftlicher Natur (Environmental, Social and Governance (ESG)). Für jeden dieser drei Bereiche wurden Ziele definiert, zu denen zukünftige Investitionen beitragen sollen. Die Ziele im

Bereich Ökologie (Umweltziele) wurden wie folgt definiert: Klimaschutz, Anpassung an den Klimawandel, nachhaltige Nutzung und Schutz von Wasser- und Meeresressourcen, Übergang zu einer Kreislaufwirtschaft, Vermeidung und Verminderung von Umweltverschmutzung, sowie Schutz und Wiederherstellung der Biodiversität und Ökosysteme [98]. Eine Aktivität gilt als nachhaltig, wenn sie zu einem der sechs Umweltziele einen Beitrag leistet und dabei gleichzeitig keines der anderen Ziele negativ beeinträchtigt. Innerhalb dieser „EU Taxonomie“ sind aktuell 13 Sektoren definiert, welche sich wiederum aus verschiedenen Aktivitäten zusammensetzen, die einem oder mehreren der sechs Umweltziele zugeordnet werden [99]. Einer der 13 Sektoren, „Construction and real estate“, bezieht sich auf die Baubranche. Der Sektor setzt sich aus sieben Aktivitäten zusammen, welche jeweils die Umweltziele Klimaschutz und Anpassung an den Klimawandel betreffen. Im Zuge dieser Arbeit sind die Aktivitäten „Acquisition and ownership of buildings“, „Installation, maintenance and repair of energy efficiency equipment“, „Installation, maintenance and repair of instruments and devices for measuring, regulation and controlling energy performance of buildings“ und „Renovation of existing buildings“ relevant. Beispielsweise gilt als nachhaltige Aktivität, dass für Nichtwohngebäude mit einer Nennleistung von 290 kW ein GA-System implementiert ist, welches in der Lage ist, ein Gebäude effizient zu betreiben. Eine weitere nachhaltige Aktivität bezieht sich auf die Installation eines *Energiemanagementsystems (EMS)*, welches zur Überwachung des energieeffizienten Betriebs eines Gebäudes eingesetzt wird. Der rechtliche Rahmen für diese Aktivitäten ist maßgeblich in der „Richtlinie über die Gesamtenergieeffizienz von Gebäuden (EPBD)“ [100] definiert. Diese Richtlinie wurde 2010 veröffentlicht und 2018 überarbeitet. Aktuell liegt ein Vorschlag der Europäischen Kommission von 2021 zur Überarbeitung der Richtlinie vor. Die aktuelle Überarbeitung wurde im Rahmen des „Fit for 55 Paketes“ durchgeführt, um die Ziele des „Green Deal“ für 2030 und 2050 im Gebäudesektor zu adressieren und einen Rahmen zu schaffen, wie diese Ziele erreicht werden können [101]. In dieser Richtlinie ist festgelegt, dass Nichtwohngebäude mit einer Nennleistung größer 290 kW für den Betrieb von Anlagen der TGA bis zum 31.12.2024 über ein GA-System verfügen müssen. Dieser Schwellenwert soll bis zum 31.12.2029 auf 70 kW reduziert werden. Das GA-System muss dabei in der Lage sein, den Energieverbrauch eines Gebäudes zu überwachen und zu analysieren und Benchmarks für die Energieeffizienz eines Gebäudes aufzustellen [101].

Neben der EPBD wurde auch die „Richtlinie zur Energieeffizienz (EED)“ von der EU als weiteres Mittel zur Energieeinsparung eingeführt. In dieser werden Maßnahmen beschrieben, die die Energieeffizienz betreffen und nicht nur auf Gebäude beschränkt sind. Da Gebäude jedoch eine bedeutende Rolle im Gesamtenergiebedarf von Unternehmen und öffentlichen Einrichtungen spielen, werden diese in der Richtlinie ebenfalls berücksichtigt. In der Richtlinie wird empfohlen, dass öffentliche Einrichtungen als Vorbild fungieren sollen. Das bedeutet, dass Städte und Kommunen in ihren Gebäuden Maßnahmen ergreifen sollen, um deren Energieeffizienz zu verbessern. Hierzu gehören neben der Sanierung von

Gebäuden auch die Einführung von Energiemonitoring (EMon)-Systemen und eines EMS. Die Richtlinie bezieht sich nicht nur auf öffentliche Einrichtungen, sondern adressiert auch Energieeinsparungen von Unternehmen. Unternehmen mit einem durchschnittlichen Energieverbrauch von über 85 TJ über alle Energieträger hinweg müssen ein EMS, zum Beispiel nach ISO 50001 [102], implementieren. Unternehmen mit einem durchschnittlichen Energieverbrauch von über 10 TJ müssen ein *Energieaudit* durchführen. Außerdem sollen alle Unternehmen dazu ermutigt werden, ein EMS zu implementieren oder Audits durchzuführen, auch wenn sie gesetzlich nicht dazu verpflichtet sind [103].

Die Überarbeitung der Gesetze der EU zur Steigerung der Energieeffizienz zielt darauf ab, Unternehmen und öffentliche Einrichtungen zu verpflichten und zu motivieren, mehr Energie einzusparen. Gebäude spielen aufgrund ihres hohen Anteils am Endenergieverbrauch der EU hierbei eine wichtige Rolle. Um Energie zu sparen, sollen zunächst mehr Monitoringlösungen installiert werden, um den Betrieb von Gebäuden zu überwachen und Optimierungspotenziale aufzudecken. Hierfür gibt es verschiedene Richtlinien, die im Folgenden betrachtet werden.

### 2.5.3 Überwachung der Energieeffizienz in Gebäuden

Für das TMon von Gebäuden stehen in Deutschland insbesondere die VDI-Richtlinie 6041 [104] und die Empfehlung zum TMon des Arbeitskreis Maschinen- und Elektrotechnik staatlicher und kommunaler Verwaltungen (AMEV) [105] zur Verfügung. Die AMEV-Richtlinie definiert konkrete Prüfgrößen für Gebäude und deren Anlagen auf Basis der VDI-Richtlinie.

Das TMon gemäß der VDI-Richtlinie ist in verschiedene Bereiche unterteilt: Anlagenmonitoring (AMon), EMon und Gebäude- und Behaglichkeitsmonitoring (GBMon). Zusätzlich gibt es noch das Einregelungsmonitoring (ERMon) und das Langzeitmonitoring (LZMon). Das ERMOn erstreckt sich in der Regel über die ersten zwei Jahre der Nutzungsphase eines Gebäudes. Im Anschluss beginnt das LZMon und endet erst, wenn das Gebäude abgerissen wird. Das AMon von Anlagen in Gebäuden dient der Erfassung und Auswertung von Betriebszuständen der technischen Einrichtungen. Dabei werden insbesondere einzelne Komponenten und Anlagenteile betrachtet. Deren Betriebsdaten werden erfasst und analysiert, um einen optimierten Betrieb sowie präventive und prädiktive Wartungen zu ermöglichen. Das EMon beschäftigt sich mit der Erfassung und Auswertung von Daten, die den Energieverbrauch eines Gebäudes und dessen Anlagen betreffen. Im Gegensatz zum AMon werden hierbei nicht nur einzelne Komponenten betrachtet, sondern auch übergeordnete, aggregierte Ebenen eines Gebäudes einbezogen. Die erfassten und analysierten Daten des AMon einzelner Komponenten können somit als Basis für das EMon dienen.

Die dritte Komponente, das GBMon, soll dem Nutzer eines Gebäudes die bestmögliche Behaglichkeit ermöglichen [104].

Basierend auf der VDI-Richtlinie definiert die AMEV-Empfehlung konkrete Prüfgrößen für das TMon. Die Begriffsbestimmungen bezüglich der Unterscheidung des TMon in die Teilbereiche (AMon, TMon und GBMon) werden in der Empfehlung zwar aufgeführt, jedoch bei der Spezifikation der Prüfgrößen nicht näher betrachtet. Die Empfehlung bezieht sich insbesondere auf das TMon in öffentlichen Einrichtungen, die jedoch auch auf nicht öffentliche Gebäude übertragen werden kann. Die definierten Prüfgrößen werden in 18 Bereiche aufgeteilt, zum Beispiel das Gesamtgebäude, Solarthermieanlagen, Heizkreise, Klimaanlage oder Wärmepumpen. Beispiele für Wärmepumpen-Prüfgrößen sind Betriebsstunden, die aktuelle Vorlauftemperatur oder die berechnete Jahresarbeitszahl. Für jede Prüfgröße ist außerdem festgelegt, ob sie obligatorisch oder optional im jeweiligen Prüfumfang enthalten sein muss [105].

Das TMon eines Gebäudes bezieht sich auf die Anlagen der TGA. Zudem sind immer mehr Unternehmen angehalten bzw. motiviert, ein EMS nach ISO 50001 [102] einzuführen (siehe Kapitel 2.5.2). Dabei handelt es sich um ein ganzheitliches System zur Erfassung und Analyse aller energierelevanten Daten eines Unternehmens. Die Erfassung der spezifischen Gebäudeenergiekosten spielt hierbei eine wichtige Rolle. Monitoring-Anwendungen nach VDI 3814 können in ein EMS integriert werden [104]. Sie werten die Energieverbräuche des Gebäudes aus und bilden Kennzahlen hierfür. Innerhalb eines EMS werden *Energieleistungskennzahlen (EnPI)s* gebildet, die zur Bewertung des Energieverbrauchs eines Unternehmens verwendet werden können [102]. Die Kennzahlen des TMon können bei der Erstellung dieser EnPIs hilfreich sein [104].

## 3 Anforderungsanalyse

In dieser Arbeit wird eine Methode entwickelt, die semantische Interoperabilität ermöglicht. Darauf aufbauend sollen automatisiert ausführbare Erkundungsmechanismen erstellt werden. Der Anwendungsfall hierbei ist eine Monitoring-Anwendungen, in welche die Informationen der Systeme automatisiert eingebunden werden. Das folgende Kapitel beschreibt die Anforderungen, die hierfür erfüllt sein müssen. Sicherheitsaspekte in Bezug auf Datenschutz und Sicherheit der Informationen, werden nicht betrachtet.

### 3.1 Anforderungen an ein automatisiertes Monitoring

Zunächst wird betrachtet, welchen Umfang das Monitoring umfassen soll, bzw. in welchen Kontext es eingebunden wird. In [104, 105] wird der Anwendungsbereich von Monitoring-Anwendungen auf das einzelne Gebäude beschränkt. Gleichzeitig können die im EMon erhobenen Daten in ein übergeordnetes EMS nach ISO 50001 eingebunden werden [104]. Insbesondere bei der Erfüllung der Richtlinien EPBD und EED (vgl. Kapitel 2.5.2) ist dies relevant. Während sich die EPBD-Richtlinie auf einzelne Gebäude bezieht und Anforderungen an ein Monitoring stellt, bezieht sich die EED-Richtlinie auf den Energieverbrauch eines Unternehmens und EMS-Systeme. Die nahtlose Integration einzelner Monitoring-Anwendungen von Gebäuden in übergeordnete EMS-Systeme ist somit eine wichtige Anforderung. Nach [102] umfasst ein EMS ein ganzes Unternehmen, welches über mehrere Standorte (Liegenschaften) verfügen kann. Eine Liegenschaft wiederum kann sich aus mehreren Gebäuden (mit Monitoring-Anwendungen) zusammensetzen. In diesem Kontext wird das EMS eines Unternehmens als Gesamtsystem betrachtet, welches aus mehreren Teilsystemen (Liegenschaften und Gebäuden) besteht. Es lassen sich folgende Anforderungen an das System ableiten.

1. Systeme, Teilsysteme und Komponenten müssen untereinander kompatibel und miteinander kombinierbar sein.

Damit die automatisierte Einbindung von Komponenten eines Gebäudes in eine Monitoring-Anwendung realisiert werden kann, müssen diese grundsätzlich kompatibel sein. Kompatibilität bezeichnet hierbei die Eignung der Komponenten und Systeme für eine gemeinsame Funktionserfüllung zusammen verwendet zu werden [81]. Um dies zu erfüllen, müssen sie

ebenfalls miteinander kombinierbar sein. Hierzu müssen Mechanismen existieren, welche Komponenten zu Teilsystemen und Teilsysteme zu Systemen zusammensetzt.

2. Die Systeme müssen während ihres Lebenszyklus um zusätzliche Komponenten und Teilsysteme erweiterbar und gleichzeitig in andere Systeme integrierbar sein.

Über den Lebenszyklus eines Systems werden unterschiedliche Anforderungen an Erweiterungen des Systems gestellt. Der Lebenszyklus technischer Anlagen ist kürzer als der des gesamten Gebäudes. Aus diesem Grund werden während des Betriebs eines Gebäudes alte Anlagen ausgetauscht und neue technische Anlagen installiert. Die neuen Anlagen müssen wiederum in das TMon des Gebäudes eingebunden werden. Das TMon muss daher flexibel um neue Komponenten erweiterbar sein. Gleichzeitig müssen die einzelnen Systeme und Teilsysteme in andere Systeme integrierbar sein. Wird für ein Unternehmen ein neues EMS aufgesetzt, müssen existierende Monitoring-Lösungen bestehender Gebäude automatisiert in dieses integriert werden können.

3. Die Teilnehmer müssen dieselbe Netzinfrastruktur nutzen, indem sie an ein Netz angeschlossen sind und das gleiche Kommunikationsprotokoll verwenden oder ein Gateway nutzen, das zwischen den Protokollen übersetzt.

Grundvoraussetzung für die Interaktion von Teilnehmern innerhalb eines Systems ist deren Netzwerkkonnektivität. Werden verschiedene Kommunikationsprotokolle innerhalb des Netzwerks verwendet, muss ein Gateway eingesetzt werden, welches zwischen diesen übersetzt. Diese Anforderung lässt sich der Ebene der Transport-Interoperabilität aus Kapitel 2.4.2 zuordnen.

4. Die Teilnehmer müssen eindeutig identifizierbar sein.

Damit Teilnehmer innerhalb eines Systems zielgerichtet und wiederholt angesprochen, sowie in das System eingebunden werden können, müssen diese eindeutig identifizierbar sein. Durch die Anforderung 2 „Erweiterbarkeit“ wird darüber hinaus die Anforderung gestellt, dass ein Teilnehmer nicht nur innerhalb eines bestehenden Systems, sondern weltweit eindeutig identifizierbar sein muss. Da ein System dynamisch erweiterbar sein soll, muss eine eindeutige Identifizierbarkeit auch im Falle einer solchen Erweiterung des Systems gewährleistet sein.

5. Die Informationen jedes Teilnehmers müssen für andere Teilnehmer verfügbar sein, damit diese die Informationen erkunden können.

Für eine automatisierte Interaktion müssen die Fähigkeiten der einzelnen Teilnehmer untereinander bekannt sein. Daher müssen die Informationen über Fähigkeiten und Daten

eines Teilnehmers den anderen Teilnehmern bereitgestellt werden und für diese erkundbar sein.

6. Es müssen Modelle vorhanden sein, die die Informationen und Fähigkeiten der Teilnehmer repräsentieren.

Damit die Informationen für andere Teilnehmer erkundbar sind, müssen Informationsmodelle existieren. Diese bilden die Daten und Fähigkeiten der verschiedenen Teilnehmer in maschinenlesbarer Form ab.

7. Um für andere Teilnehmer verfügbar zu sein, müssen die Informationen durch einen Repräsentanten in der virtuellen Welt repräsentiert werden.

Damit auf diese Modelle zugegriffen werden kann, werden sie in einen virtuellen Repräsentanten der einzelnen Teilnehmer integriert. Dieser stellt Zugriffspunkte bereit, welche den Zugriff auf die Modelle, und die Erkundung der darin liegenden Informationen ermöglichen.

Die Anforderungen 4, 5, 6 und 7 lassen sich der Ebene der syntaktischen Interoperabilität (vgl. Kapitel 2.4.2) zuordnen.

8. Die erkundeten Informationen müssen für andere Teilnehmer interpretierbar sein, damit sie weiterverarbeitet werden können.

Damit die erkundeten Informationen automatisiert verarbeitet und in Anwendungen integriert werden können, müssen sie interpretierbar sein. Nur wenn die Informationen von anderen Teilnehmern verstanden werden, können sie automatisiert verarbeitet werden.

9. Semantische Modelle, die die Informationen der Teilnehmer beschreiben, müssen vorhanden sein.

Die Anforderung der Interpretierbarkeit bedingt, dass die Informationen in Form semantischer Modelle vorliegen müssen. Um Interoperabilität zwischen Teilnehmern zu gewährleisten, müssen alle Teilnehmer entweder die gleichen Modelle als Basis verwenden, oder Mechanismen zur Übersetzung bzw. Verknüpfung heterogener semantischer Modelle müssen erstellt werden.

10. Mechanismen zur automatisierten Abbildung heterogener semantischer Modelle müssen vorhanden sein.

Übergeordnetes Ziel des automatisierten Monitorings ist es, menschliches Eingreifen bei der Konfiguration zu verhindern. Können sich unterschiedliche Unternehmen und Branchen nicht auf einheitliche Standards einigen und keine Mechanismen zur automatisierten Abbildung entwickelt werden, ist ein händisches Eingreifen zur Abbildung erforderlich. Dies hat zur Folge, dass die Informationen der verschiedenen Teilnehmer nicht automatisiert in Monitoring-Anwendungen eingebunden werden können. Aus diesem Grund müssen Mechanismen existieren, die heterogene Semantik verarbeiten können, und auf ihr eigenes Vokabular abbilden. Der Aspekt der semantischen Interoperabilität ist mit den Anforderungen 8, 9 und 10 adressiert.

11. Erkundungsmechanismen zur Integration der interpretierten Informationen in Anwendungen müssen implementiert werden.

Die Anforderungen 1 bis 10 beschreiben allgemeine Anforderungen, die für automatisierte Interaktionen und Erkundungsmechanismen erfüllt sein müssen. Die Anforderungen müssen erfüllt werden, damit die Informationen der Anlagen in eine Monitoring-Anwendung integriert werden können. Um dies automatisiert zu gewährleisten, sind implementierte Erkundungsmechanismen erforderlich, die verwendet werden können, um nach bestimmten Informationen zu suchen und sie in die Anwendung einzubinden.

## 3.2 Diskussion Anforderungsanalyse

Im vorherigen Abschnitt wurden elf Anforderungen beschrieben, die für ein automatisiertes Monitoring erfüllt sein müssen. Die ersten zehn Anforderungen sind generischer Natur und können auch auf andere Anwendungsfälle als das Monitoring übertragen werden, beispielsweise für die automatisierte Interaktion verschiedener Entitäten oder das automatisierte Engineering. Die elfte Anforderung ist auf den Anwendungsfall dieser Arbeit bezogen, da die Informationen der Entitäten automatisch ausgelesen und in die Anwendung eingebunden werden sollen. In Abbildung 3.1 sind die verschiedenen Anforderungen beispielhaft dargestellt.

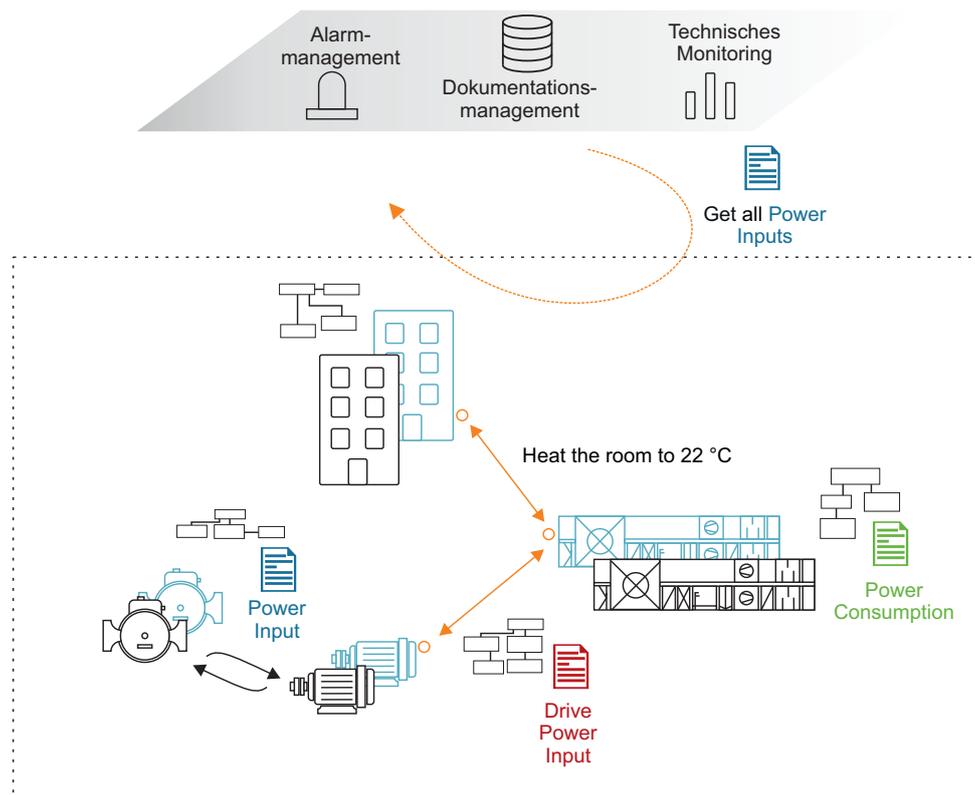


Abbildung 3.1: Visuelle Darstellung der spezifizierten Anforderungen

Die Abbildung zeigt als technische Komponente eine Pumpe, welche in das System einer Raumluftechnik (RLT)-Anlage integriert ist, welche wiederum Teil eines Gebäudes ist. Eine Monitoring-Anwendung greift hierbei auf die Informationen der Komponenten und Systeme zu, wobei die Information *Power Input* von der Anwendung angefordert wird. Um interoperabel miteinander interagieren zu können, müssen die spezifizierten Anforderungen aus Kapitel 3.1 erfüllt sein: Die Pumpe und die RLT-Anlage müssen miteinander kompatibel sein, sodass sie ihre Informationen nahtlos miteinander austauschen können. Wird die Pumpe gegen eine neue Pumpe ausgetauscht, muss die neue Pumpe ebenfalls mit der RLT-Anlage kompatibel sein. Um von anderen Teilnehmern angesprochen werden zu können, müssen die jeweiligen Entitäten eindeutig identifizierbar sein. Die Informationen müssen für andere Teilnehmer abrufbar sein. Hierzu müssen Informationsmodelle existieren, welche die Informationen anderen Teilnehmern bereitstellen. Semantische Modelle müssen ebenfalls vorhanden sein, damit die Informationen verständlich und interpretierbar sind. Werden von den Entitäten unterschiedliche semantische Modelle verwendet (wie in der Abbildung dargestellt), müssen Übersetzungsmechanismen existieren, damit eine Interaktion dennoch automatisiert realisiert werden kann. Sind diese Anforderungen alle erfüllt, kann die Monitoring-Anwendung auf die Informationen der Entitäten zugreifen und sie einbinden.

In Kapitel 2.4 wurden verschiedene Ansätze vorgestellt, welche Interoperabilität auf unterschiedliche Ebenen aufteilen. Das folgende Kapitel gibt einen Überblick über den aktuellen Stand der Technik hinsichtlich der Erfüllungsgrade der Ebenen der Interoperabilität. Auf

Basis des aktuellen Stands der Technik, wird in Kapitel 4.5 analysiert, welche der spezifizierten Anforderungen des Kapitels 3.1 erfüllt sind, und welche offenen Herausforderungen noch existieren. Auf Basis dieser Herausforderungen wird das Ziel dieser Arbeit definiert.

## 4 Stand der Technik

In diesen Kapiteln wird der aktuelle Stand der Technik der Bereiche digitale Zwillinge, I4.0 und automatisierte Klassifikation von Daten der GA beschrieben. Am Ende des Kapitels wird der Erfüllungsgrad der definierten elf Anforderungen aus Kapitel 3.1 im Hinblick auf den aktuellen Stand der Technik dargestellt und die Forschungslücke herausgearbeitet, die in dieser Arbeit adressiert wird.

### 4.1 Digitale Zwillinge

Zentraler Bestandteil der elf Anforderungen ist die Forderung, dass die Informationen eines Assets für andere verfügbar und interpretierbar sein müssen. Dazu müssen sie modelliert und für andere Teilnehmer bereitgestellt werden. In einer zunehmend digitalisierten und globalisierten Welt müssen diese Informationen virtuell zugänglich sein, um sicherzustellen, dass die Informationen der Assets nicht nur intern und lokal, sondern auch unternehmensweit und global abrufbar sind. Anforderung 7 besagt, dass ein virtueller Repräsentant eines Assets vorhanden sein muss, der diese Informationen zur Verfügung stellt. Die virtuelle Repräsentanz eines Assets wird derzeit mit verschiedenen Begriffen beschrieben, wie bspw. *virtuelle Entität* [106], *digitaler Schatten* [107] oder *digitales Modell* [107]. In dieser Arbeit wird der Begriff digitaler Zwilling verwendet. Der Begriff des digitalen Zwillings hat sich in den letzten Jahren etabliert und wird in wissenschaftlichen Veröffentlichungen [108, 109] und internationalen Normungskomitees [110] am häufigsten verwendet.

Die erste Erwähnung des Begriffs digitaler Zwilling wurde nach [111] 2003 in einem Kurs an der Universität von Michigan getätigt, während die erste offizielle Definition des Begriffs auf ein Projekt der National Aeronautics and Space Administration (NASA) von 2010 zurückgeht [112]. Seitdem wurde in verschiedenen Veröffentlichungen, Arbeitskreisen und der Normung der Begriff des digitalen Zwillings aufgegriffen und weiter spezifiziert, wobei hier unterschiedliche Ansätze verfolgt wurden. In [108, 109] werden verschiedene akademische Veröffentlichungen analysiert und aus diesen Definitionen der Begriff des digitalen Zwillings abgeleitet. Im Folgenden wird die Definition gemäß [108] wiedergegeben:

„A set of adaptive models that emulate the behaviour of a physical system in a virtual system getting real time data to update itself along its life cycle. The digital twin replicates the physical system to predict failures and opportunities for changing, to prescribe realtime actions for optimizing and/or mitigating unexpected events observing and evaluating the operating profile system.“

Ein digitaler Zwilling bildet das Verhalten des zugehörigen physischen Assets in der virtuellen Welt ab und ist somit das virtuelle Abbild bzw. der virtuelle Repräsentant des Assets. Ein digitaler Zwilling muss über den gesamten Lebenszyklus des Assets gepflegt und mit realen Daten versorgt werden. Der zweite Teil der Definition bezieht sich auf die Anwendungsbereiche eines digitalen Zwillings, z.B. um Ausfälle vorherzusagen oder den Betrieb zu optimieren. Zu Beginn der Definition wird von mehreren Modellen gesprochen, aus denen sich ein digitaler Zwilling zusammensetzt. Dies bedeutet, dass für verschiedene Anwendungsfälle oder Lebenszyklusphasen eines Assets unterschiedliche Modelle verwendet werden.

In den letzten Jahren wurden verschiedene Konzepte hinsichtlich der Standardisierung der Informationsmodelle digitaler Zwillinge entwickelt. Beispiele hierfür sind die AAS [11], die *Web of Things (WoT) Thing Description* [113], *Next Generation Services Interface-Linked Data (NGSI-LD)* [114] oder die *Digital Twins Definition Language (DTDL)* [115]. In verschiedenen Veröffentlichungen werden diese Konzepte verglichen [116–118]. Beim Vergleich kommt [118] zu dem Schluss, dass die Konzepte der AAS und der Thing Description am besten geeignet sind, um die dort spezifizierten Anforderungen zu erfüllen. Der Vergleich in [116] ergibt, dass die AAS, Thing Description und NGSI-LD die umfassendsten Möglichkeiten besitzen, um die dort beschriebenen Anforderungen zu erfüllen. [117] vergleicht verschiedene Konzepte hinsichtlich ihrer Eignung für die Integration von Automationsstationen in Automatisierungssysteme. Keiner der Standards kann die in [117] spezifizierten Anwendungsfälle vollständig abdecken. Um dieses Problem zu lösen, wird gezeigt, wie AAS und Thing Description kombiniert werden können. Die Ergebnisse der drei Veröffentlichungen zeigen, dass die AAS und die Thing Description für unterschiedliche Anwendungen von digitalen Zwillingen nutzbar sind. Zudem werden beide Konzepte kontinuierlich weiterentwickelt, um gegenwärtige Mängel in Zukunft zu beheben. Im Rahmen dieser Arbeit wird eine Methode zur Erreichung semantischer Interoperabilität von digitalen Zwillingen entwickelt. Die Grundprinzipien der Methode lassen sich unabhängig vom verwendeten digitalen Zwilling-Konzept (z.B. AAS oder Thing Description) einsetzen. Bei der Umsetzung der Methode als Service müssen jedoch die Struktur und Möglichkeiten der jeweiligen Informationsmodelle der Zwillinge berücksichtigt werden. In dieser Arbeit wurde hierzu das Konzept der AAS angewendet, da im Zuge dieser Arbeit neben der Methode zur semantischen Interoperabilität auch weitere Arbeiten im Kontext der AAS durchgeführt wurden.

## 4.2 Interoperabilität im Kontext Industrie 4.0

Das folgende Kapitel stellt einen Überblick über den aktuellen Stand der Technik hinsichtlich Interoperabilität im Kontext der I4.0 dar. Zunächst werden die Grundlagen des Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0) kurz erläutert, da dies die Basis für einen Teil dieser Arbeit bildet. Danach folgt eine Beschreibung des Metamodells der AAS sowie des I4.0-Service-Modells. Darauf aufbauend wird erläutert, wie derzeit im Kontext von I4.0 semantische Interoperabilität erreicht werden soll.

### 4.2.1 Referenzarchitekturmodell Industrie 4.0

Das RAMI 4.0 wurde entwickelt, um unterschiedliche Aspekte von I4.0 in einem Modell zusammenzuführen. Ziel war es, eine gemeinsame Basis, hinsichtlich des Verständnisses von I4.0, zu entwickeln. Mit Hilfe des Modells können vorhandene Normen und Standards verortet, Überschneidungen identifiziert und gelöst, Use-Case-Inhalte lokalisiert und übergeordnete Regeln definiert werden. Das hierfür entwickelte dreidimensionale Modell ist in Abbildung 4.1 dargestellt.

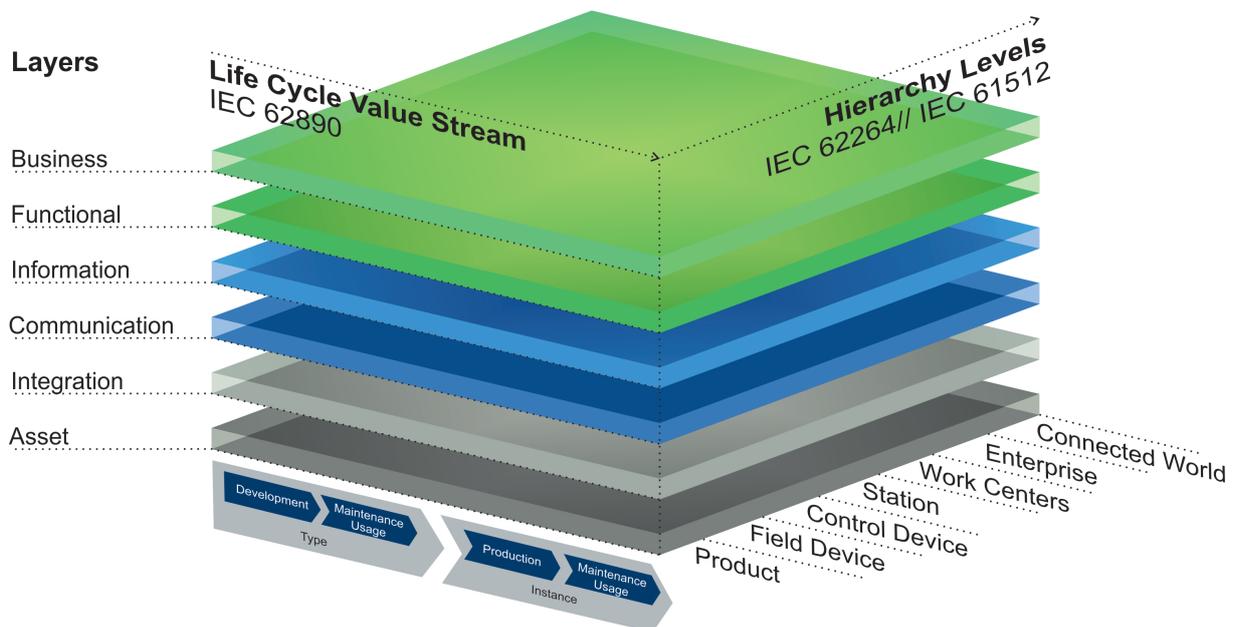


Abbildung 4.1: RAMI 4.0 nach [119]

Auf der senkrechten Achse werden sechs verschiedene Layer abgebildet, durch die unterschiedliche Sichtweisen wie das Kommunikationsverhalten oder die funktionale Beschreibung von Produkten möglich sind. In der waagerechten Achse wird der Lebenszyklus von Produkten, in Anlehnung an die IEC 62890 [120], dargestellt. Grundlage dieser Norm ist die Unterscheidung zwischen dem Typ und der Instanz eines Assets. Die Typ-Phase erstreckt sich von der ersten Idee zu einem Produkt bis hin zur Freigabe der Serienproduktion. Die

Instanz-Phase erstreckt sich von der Produktion, basierend auf dem allgemeinen Typ, über den Gebrauch, bis hin zur Entsorgung des Produktes. Die dritte Achse repräsentiert die Verantwortlichkeiten und Funktionalitäten, welche innerhalb einer Fabrik auftreten. Über einfache Feldgeräte und kompakte Einheiten, bis hin zur ganzen Fabrik, können hier Verantwortlichkeiten untereinander dargestellt werden [121]. Für eine tiefere Betrachtung der Grundlagen des Modells wird auf [119] verwiesen.

#### 4.2.2 Struktur der Verwaltungsschale

Im Bereich der I4.0 hat sich als Basis für Interoperabilität das Konzept der AAS etabliert [10]. Die AAS ist der digitale Repräsentant eines Assets in der digitalen Welt [122]. Ein Asset kann hierbei jede Entität sein, die einen Wert für eine Organisation hat [123]. Die AAS wird auch als die Umsetzung des digitalen Zwillings der I4.0 bezeichnet [10]. Das Informationsmodell der AAS definiert die Struktur, wie Informationen verschiedener Assets strukturiert sein müssen [11]. Die Komposition aus AAS und Asset wird als *I4.0-Komponente* bezeichnet. Ein Überblick über das Metamodell der AAS ist in Abbildung 4.2 gegeben. Bei der Notation der verschiedenen Unified Modeling Language (UML)-Abbildungen dieser Arbeit wurde sich an der Notation gemäß [11] orientiert.

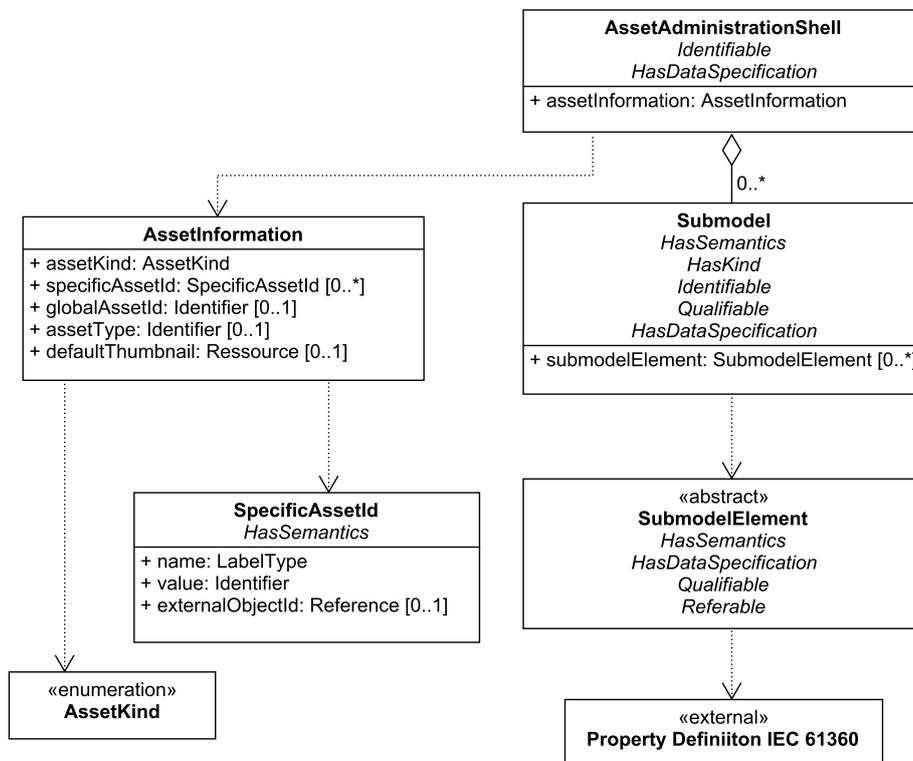


Abbildung 4.2: Überblick über das Metamodell der AAS nach [11]

Da das Metamodell der AAS sehr umfassend ist, werden im folgenden Abschnitt nur

die für diese Arbeit relevanten Aspekte beschrieben. Eine ausführliche Beschreibung des gesamten Modells ist in [11] gegeben. Die AAS erbt von der Klasse *Identifiable*. Hierdurch wird festgelegt, dass jede AAS über eine weltweit eindeutige *Identifikationsnummer (ID)* verfügen muss. Die in der Klasse *Identifiable* verpflichtende ID soll hierbei ein *Internationalized Resource Identifier (IRI)* sein. Die Informationen über das von der AAS repräsentierte Asset sind in der Klasse *Asset-Information* enthalten. Zentraler Baustein der AAS sind *Teilmodelle (Submodels)*, welche Eigenschaften und Funktionalitäten der Assets und deren Inhalte abbilden [11]. Teilmodelle enthalten *Submodel-Elements (SE)s*, welche die tatsächlichen Informationen der Assets abbilden. Die Klasse SE ist die abstrakte Klasse aller Elemente, die ein Teilmodell enthalten kann. Abbildung 4.3 zeigt die abstrakte Klasse SE mit relevanten Subklassen für diese Arbeit. Die vollständige Liste der Subklassen ist in [11] beschrieben.

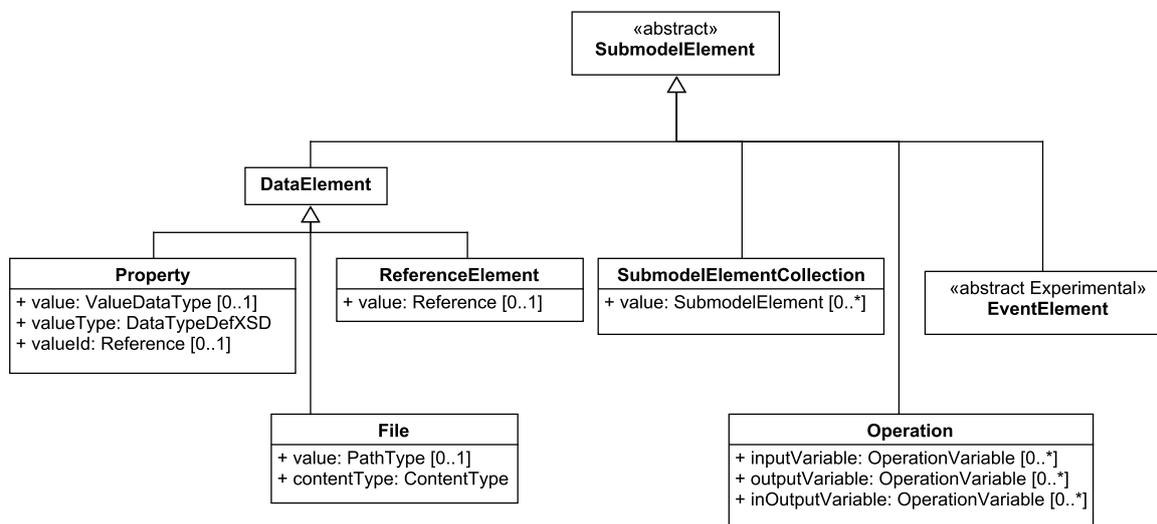


Abbildung 4.3: Abstrakte SE-Klasse mit Subklassen nach [11]

Die am häufigsten genutzte Klasse ist *Property*, welche Merkmale darstellt, um bspw. die Grenzen einer Komponente zu beschreiben. Datenblätter in Form von PDFs, oder andere Dokumente, können mittels der Klasse *File* den Teilmodellen hinzugefügt werden. Referenzen zwischen Klassen innerhalb einer AAS und zwischen AAS werden mittels der Klasse *Reference-Element* hergestellt. Im weiteren Verlauf dieser Arbeit wird dieses Element insbesondere dann angewendet, wenn Werte eines Teilmodells an anderer Stelle der AAS benötigt werden, um zum Beispiel Berechnungen durchzuführen. Diese Berechnungen können mithilfe der Klasse *Operation* durchgeführt werden. Die Klasse *Submodel-Element-Collection (SMC)* wird verwendet, um weitere SEs thematisch zu bündeln, sodass zusammenhängende SEs innerhalb eines Teilmodells zusammengefasst werden können. Die Klasse *Event* löst Ereignisse aus, jedoch ist sie aktuell noch nicht vollständig spezifiziert.

Jedes SE erbt von der Klasse *Has-Semantics*. Diese Klasse legt fest, dass jedes Element über eine *Semantic-ID* verfügen muss, welche eine Referenz zur semantischen Definition des Elements enthält [11]. Mithilfe dieser ID kann entweder direkt auf einen externen Standard wie *ECLASS* [124] verwiesen werden oder auf eine *Concept-Description (CD)*. In dieser sind Attribute enthalten, welche das jeweilige Element semantisch definieren. Dabei kann die CD auch eine Kopie eines externen Standards sein. CDs sollten grundlegend einem *Data-Specification-Template* folgen, welches definiert, welche Attribute die CD enthalten kann. Für verschiedene SEs, wie Properties oder SMCs sollen in Zukunft unterschiedliche Templates definiert werden, da sie je nach SE-Typ unterschiedliche Attribute aufweisen. In [11] wird das Template *Data-Specification-IEC-61360* definiert, welches für die Definition von Properties empfohlen wird. Die Inhalte des IEC-Templates sind in Abbildung 4.4 dargestellt.

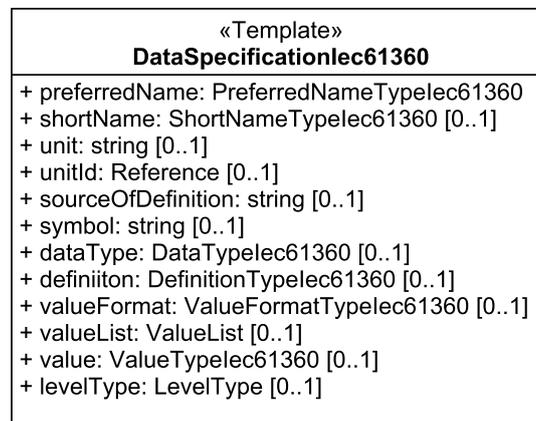


Abbildung 4.4: Data Specification Template IEC 61360 nach [11]

In der Klasse sind 12 Attribute definiert, welche ein Property zusätzlich zu den Informationen aus dem Metamodell der AAS weiter beschreiben. Bei der Spezifikation des Templates wurde die IEC 61360-1 [125] verwendet. Bis auf das Attribut *Preferred-Name* ist keines verpflichtend. Die Attribute *Short-Name* und *Definition* können verwendet werden, um ein Property so zu spezifizieren, dass die Bedeutung für einen Menschen interpretierbar ist. Mittels der Attribute *Unit*, *Data-Type* und *Symbol* können Metadaten angegeben werden, welche das Property näher definieren.

Das Metamodell der AAS definiert die Struktur, wie die Informationen eines Assets in der digitalen Welt modelliert und strukturiert sein müssen. Durch den Einsatz eines einheitlichen Modells kann syntaktische Interoperabilität erreicht werden. Neben der syntaktischen Interoperabilität ist für eine automatisierte Interaktion von Assets auch semantische Interoperabilität notwendig. Der aktuelle I4.0-Ansatz wird im folgenden Kapitel beschrieben.

### 4.2.3 Semantische Interoperabilität von Verwaltungsschalen

Neben der spezifizierten Struktur durch das Informationsmodell wird für die Interaktion von I4.0-Komponenten eine eindeutige Semantik benötigt [10, 122]. Diese wird erreicht, wenn die Teilmodelle in standardisierter Form vorliegen und deren Semantik ebenfalls einheitlich standardisiert ist. Verschiedene Initiativen standardisieren Teilmodelle für Komponenten der Industrie, bspw. [126, 127], wodurch sogenannte *Submodel-Templates* entstehen [14]. Die Standardisierung der Teilmodelle wird in Deutschland von der *Industrial Digital Twin Association (IDTA)* koordiniert. Die Integration der Teilmodelle in Vokabulare wie den *ECLASS-Standard* [124], oder das *IEC Common Data Dictionary (IEC CDD)* [128], ermöglicht die eindeutige Identifikation der Teilmodelle sowie deren SEs und bereitet so den Weg für eine eindeutige Semantik. Syntaktisch wird ein *Key-Value-Paar-Konzept* verwendet, das zwischen dem Key als Identifikation für das entsprechende Modellelement und der Zuordnung der Bedeutung unterscheidet. Das Modell sieht explizit vor, dass ein Modellelement mehr als einen Identifikator hat und dass die Identifikatoren von unterschiedlichen Typen sein können. So wird z. B. für den Zugriff auf das Modellelement in *OPC UA* oder *Hypertext Transfer Protocol (HTTP)* ein frei definierbarer String verwendet, und für den Verweis auf den Dictionary-Eintrag ein standardisierter Identifier, wie z.B. eine *International Registration Data Identifier (IRDI)* oder eine *Universal Resource Locator (URL)*. In dem Dictionary, das ebenfalls eine standardisierte Struktur aufweist (IEC 61360 [125]), ist dann mittels Attributen das Element definiert. Diese Attribute sind teilweise maschineninterpretierbar.

Der derzeitige I4.0-Ansatz zur semantischen Interoperabilität verfolgt einen homogenen (vgl. Kapitel 2.4.4) Ansatz. Innerhalb eines Wertschöpfungsnetzwerks müssen sich die verschiedenen Teilnehmer auf einen gemeinsamen semantischen Standard einigen, um automatisierte Interaktionen zu ermöglichen. Ein Wertschöpfungsnetzwerk ist hierbei ein Netzwerk aus Wertschöpfungsketten, welche über Unternehmensgrenzen hinweg entstehen können [129]. Der Ansatz der homogenen Semantik zum Erreichen semantischer Interoperabilität wird auch in [11] beschrieben. Es wird unter anderem festgehalten, dass Interoperabilität nicht gewährleistet werden kann, wenn die CD einer AAS keine Kopie eines externen semantischen Standards darstellt. Wenn eine proprietäre, herstellereigene Spezifikation eines Merkmals innerhalb einer CD verwendet wird, kann keine Interoperabilität zwischen AAS garantiert werden. Der Einsatz homogener Semantik wird in [11] als *Exact-Matching* bezeichnet. Darüber hinaus wird ein zweiter Ansatz des Matchings (*Intelligent-Matching*) beschrieben und in drei Unterkategorien aufgeteilt. Bei der ersten Unterkategorie des Matchings kann die Version ignoriert werden (*Ignore-Versioning*). Wenn zwei Semantic-IDs auf ECLASS basieren, jedoch unterschiedliche Versionen verwendet werden (z.B. 11 und 12), kann das Ende der ID, das die Version beschreibt, ignoriert werden. Die zweite Unterkategorie umfasst vordefinierte semantische Zuordnungen (*Semantic-Mapping*). Dadurch

können Mappings zwischen Elementen erstellt werden, die innerhalb eines semantischen Standards (z.B. ECLASS) oder zwischen Standards (z.B. zwischen ECLASS und IEC CDD) die gleiche Bedeutung haben. Dadurch können Äquivalenztabelle erstellt werden, die diese Verbindungen enthalten. Bei der dritten Matching-Art sollte Domänenwissen (*Domain-Knowledge*) verwendet werden. Hierbei muss dieses jedoch in maschinenlesbarer Form vorliegen, um eine Zuordnung von Elementen zueinander zu ermöglichen.

#### 4.2.4 Industrie 4.0 Service Model

Der Zugriff auf die Informationen einer AAS wird durch Schnittstellen (*Interfaces*) realisiert. Um einen automatisierten Zugriff auf die Interfaces verschiedener AAS zu ermöglichen, werden diese Interfaces standardisiert. Nach [12] werden durch das I4.0-Service-Model vier Level beschrieben, die bei der Standardisierung von Interfaces durchlaufen werden. Zunächst werden auf einem *Technologie-neutralen Level* die Interfaces und dazugehörigen Operationen in einer allgemeinen textuellen Form beschrieben. Das zweite Level (*Technologie-spezifisches Level*) beschreibt wie diese allgemeinen Beschreibungen in verschiedenen Technologien (z.B. *HTTP/REST*, *OPC UA*, *MQTT*) angewendet werden können. Das dritte Level wird als *Implementierungslevel* bezeichnet und umfasst die Implementierung der Interfaces mittels einer konkreten Sprache wie *Python* oder *Java*. Das letzte Level beschreibt das *Runtime Level*, also die konkrete Umsetzung der Interfaces in einer I4.0-Umgebung. In [12] werden die Level 1 und 2 verschiedener Interfaces der AAS beschrieben, bspw. *Get-Submodel*. Dieses Interface kann aufgerufen werden, um ein bestimmtes Teilmodell einer AAS auszulesen. Die allgemeinen Interfaces können in verschiedenen Services bereitgestellt werden. Grundlegend wird hierbei zwischen *Infrastructure-Services* und *Application-relevant-Software-Services* unterschieden [130]. Infrastructure-Services bezeichnen allgemeine Services, die in konkreten Application-relevant-Software-Services wiederverwendet werden können [130]. Die standardisierten Interfaces und darauf aufbauenden Services werden durch das *Industrie 4.0 Operating System – BaSyx 4.0* bereitgestellt.

#### 4.2.5 BaSyx 4.0 – Industrie 4.0 Operating System

Die Open-Source Referenzimplementierung *BaSyx* ist ein vom Fraunhofer Institut entwickeltes Software Development Kit (SDK), welches Softwarekomponenten zur Realisierung einer I4.0-Umgebung bereitstellt [131]. Eine I4.0-Umgebung setzt sich aus drei Kernelementen zusammen [132].

- *AAS-Server*: Mit einem AAS-Server werden AAS inkl. der Teilmodelle in einer HTTP-Laufzeitumgebung ausgeführt und stellen Informationen über die beschriebenen Interfaces bereit.

- *Registry*: Eine Registry dient in einer I4.0-Umgebung als zentrale Verwaltung für AAS. Informationen über Verfügbarkeit und Existenz vorhandener Teilmodelle von AAS werden in der Registry hinterlegt. I4.0-Applikationen können die Registry zur Erkundung des I4.0-Systems nutzen.
- *Data-Bridge*: Die Data-Bridge ermöglicht den Informationsaustausch zwischen physischen Assets (Unterstützung unterschiedlicher Kommunikationsprotokolle bspw. OPC UA, MQTT, HTTP) und einem AAS-Server.

Damit die Data-Bridge innerhalb von Gebäuden eingesetzt werden kann, müssen Kommunikationsprotokolle der GA verarbeitet werden können. Im folgenden Abschnitt sind Kommunikationsprotokolle der GA beschrieben.

### 4.3 Kommunikationsprotokolle der Gebäudeautomation

Innerhalb der GA werden verschiedene Kommunikationsprotokolle eingesetzt, um Automatisierungsfunktionen umzusetzen [133]. Hierzu zählen bspw. *KNX* [134], das *Local Operating Network (LON)*-Protokoll [135], oder das *Building Automation and Control Network (BACnet)*-Protokoll [136]. In dieser Arbeit wurde das BACnet-Protokoll als Basis verwendet. BACnet ist ein standardisiertes Internet Protokoll (IP)-fähiges Netzwerkprotokoll und wird für den Informationsaustausch zwischen verschiedenen Systemteilnehmern der GA eingesetzt [136]. Im Folgenden werden die relevanten Aspekte des Protokolls für diese Arbeit beschrieben. Ein BACnet-Gerät (*BACnet-Device*) beschreibt einen Systemteilnehmer durch verpflichtende und optionale BACnet-Objekte (*BACnet-Objects*). Jedes Gerät hat ein verpflichtendes *Device-Object*, wodurch es für andere Systemteilnehmer auffindbar ist. Hierdurch werden in einem GA-Netzwerk Erkundungsmechanismen ermöglicht. Beispiele für BACnet-Objekte sind analoge- (*Analog-Objects*) und binäre Objekte (*Binary-Objects*), Kalender-Objekte (*Calendar-Objects*), oder Objekte zur Aufzeichnung von Trends (*Trend-Objects*). Ein BACnet-Objekt setzt sich wiederum aus verpflichtenden und optionalen *BACnet-Properties* zusammen. BACnet-Properties beschreiben die Informationen eines BACnet-Objektes wie z.B.:

- *Object-Name* als Kennzeichnung eines Datenpunktes. Dieser muss innerhalb eines BACnet-Gerätes einzigartig sein.
- *Object-Description* als menschenlesbare Beschreibung eines Datenpunktes.
- *Object-Present-Value* als Zustandswert eines Datenpunktes.
- *Object-Unit* um einem Zustandswert eine physikalische Einheit zuzuordnen.

Um Datenpunkte in den Kontext einer Anlagenstruktur einzuordnen werden von Anwendern häufig spezifische Regularien über die Zusammensetzung des BACnet-Object-Name gefordert. Diese Anlagenkennzeichnungsschlüssel (AKS) unterscheiden sich in Abhängigkeit der Anforderungen des Projektträgers und der Umsetzung der ausführenden Unternehmen. Im nächsten Abschnitt sind existierende Ansätze zur Verarbeitung heterogener AKS beschrieben.

#### 4.4 Klassifizierung heterogener Gebäudeautomationsdaten

Es existieren verschiedene Methoden zur Klassifizierung heterogener Datenpunkte von Gebäuden. Dieses Forschungsfeld wird auch als *Gebäudemetadatenklassifizierung* bezeichnet und kann grob in zwei Ansätze unterteilt werden: Zeitreihen [137–144] und Semantik [145–154]. Da diese Arbeit dem semantikbasierten Ansatz zugeordnet ist, werden die unterschiedlichen semantischen Ansätze in der folgenden Tabelle 4.1 skizziert.

Tabelle 4.1: Überblick über die Ansätze zur automatisierten Klassifizierung von Datenpunkten auf Basis der Semantik

Referenz	Methode	Modelltyp	Anzahl Gebäude	Ebene der Klassifizierung	Anzahl Label
[145]	Active-Learning mit Clustering	Hierarchical-Clustering	4	Datenpunkt Typ	-
[146]	Active-Learning mit Clustering	Syntactic-Clustering	3	Sensor Typ	-
[147]	Active-Learning mit Clustering	Gaussian-Mixture-Model-Clustering	3	Sensor Typ	21
[148]	Active-Learning	Conditional-Random-Fields mit Multi-Layer-Perceptron und Multi-Label-Classifier	5	Datenpunkt Typ	284
[154]	Active-Learning	Conditional-Random-Fields	3	Sensor Typ	-
[150]	Active-Learning mit Clustering	Selective-Sampling und Clustering	5	Sensor Typ	18
[149]	Transfer-Learning	LSTM mit Embeddings	3	Sensor Typ	-
[152]	Clustering	DBSCAN-Clustering und Fuzzy-String-Matching	4	Datenpunkt Typ	-
[153]	Transfer-Learning	Bidirectional-LSTM mit Embeddings	6	Entität Typ	-
[151]	Transfer-Learning	Transformer-Model mit Classifier	152	Datenpunkt Typ	218

Aktuelle Ansätze lassen sich basierend auf der verwendeten Methodik und Modelltyp in drei Gruppen einteilen. Die erste Methodik ist die Verbindung von aktivem Lernen (*Active-Learning*) und *Clustering* [145–147, 150]. Für das aktive Lernen [155] ist es erforderlich, dass fortlaufende Annotatoren relevante Datenpunkte für neue Gebäude kennzeichnen, um den Algorithmus an das jeweilige Gebäude anzupassen. Danach werden die Datenpunkte von verschiedenen Clustering-Ansätzen zu kohärenten Clustern gruppiert. Die Methoden erzielen hohe Genauigkeiten mit einem Durchschnitt von über 90% innerhalb ihrer Datensätze. Die zweite Gruppe verwendet eine Kombination aus aktivem Lernen und *Conditional-Random-Fields*, und erreicht ebenfalls eine Genauigkeit von über 90% [148, 154]. Diese Ansätze reduzieren den manuellen Engineering-Aufwand zur Erstellung von TMon-Anwendungen erheblich, für neue Gebäude ist jedoch weiterhin die Kennzeichnung einiger Datenpunkte erforderlich. [149, 151, 153] verwenden Transfer-Learning [156], welches auf Sprachmodellen basiert. Die Kombination von erlernten WEs und LSTM-Netzwerken wird in [149, 153] verwendet, während der Ansatz in [151] auf der Transformer-Architektur basiert. Letztere trainieren ein neues Sprachmodell auf die AKS verschiedener Gebäude, um die Syntax und Semantik der AKS zu erfassen. Aufgrund der Transfer-Fähigkeit trainierter NLP-Modelle ist keine erneute Annotation neuer Gebäude notwendig, da die Modelle über mehrere Gebäude hinweg angewendet werden können. In [153] wurden Genauigkeiten von über 90% erzielt und in [149] lagen die F1-Werte zwischen 63% und 90%. In [151] konnte eine Präzision (Precision) von 77% und eine Genauigkeit von 71% erreicht werden. Hierbei ist anzumerken, dass in [151] die Ergebnisse an 30 Gebäuden ausgewertet wurden, während [153] und [149] ihre Ergebnisse nur an zwei bzw. einem Gebäude ausgewertet haben.

### **4.5 Diskussion - Aktueller Stand der Interoperabilität im Bereich Industrie 4.0**

In diesem Kapitel werden die Erkenntnisse der Kapitel 2, 3 und 4 zusammengefasst, analysiert und die Forschungslücke herausgearbeitet. In Kapitel 2 wurden verschiedene Grundlagen vorgestellt, auf welchen in dieser Arbeit aufgebaut wird, darunter auch Grundlagen zur Interaktion und Interoperabilität von Systemen. In Kapitel 3 wurden elf Anforderungen spezifiziert, die erfüllt werden müssen, um allgemein automatisierte Interaktionen von Systemen zu ermöglichen und darauf aufbauend ein automatisiertes Monitoring zu realisieren. Kapitel 4 gibt einen Überblick über den aktuellen Stand der Technik im Bereich I4.0 hinsichtlich der Interoperabilität von Systemen. Im Folgenden wird aufgezeigt, inwiefern aktuelle I4.0-Konzepte die Anforderungen des dritten Kapitels erfüllen können.

Wenn sich alle Teilnehmer innerhalb eines Wertschöpfungsnetzwerks auf die Verwendung des Metamodells der AAS und eine einheitliche Semantik einigen, werden die spezifizierten Anforderungen erfüllt. Durch die Verwendung des Metamodells der AAS, standardisierter Teilmodelle und Interfaces für den Informationsabruf können die Anforderungen 1 bis 9 erfüllt werden. Die Anforderung 10 zur Verarbeitung heterogener Semantik entfällt, da eine homogene Semantik verwendet wird. Um hierarchische Strukturen von AAS untereinander abzubilden, kann das Teilmodell *Hierarchical-Structures* [157] genutzt werden (Anforderung 1). AAS können zudem während ihres Lebenszyklus um weitere AAS erweitert werden, beispielsweise wenn eine neue technische Komponente in einem Gebäude installiert wird (Anforderung 2). Die Teilmodelle *Asset-Interface-Description (AID)* [158] und *Asset-Interfaces-Mapping-Configuration (AIMC)* [159] haben die Aufgabe, die Informationen der technischen Komponenten in AAS abzubilden, um einen einheitlichen Zugriff auf diese zu ermöglichen (Anforderung 3). Der entwickelte Identifikationsmechanismus für AAS ermöglicht eine weltweit eindeutige Identifikation der AAS und erfüllt somit Anforderung 4. Durch die standardisierten Interfaces wie *Get-Submodel* sind die Informationen von AAS erkundbar (Anforderung 5). Standardisierte Teilmodelle des IDTA repräsentieren unterschiedliche Informationen und Funktionalitäten von Assets (Anforderung 6). Durch die Verwendung von Vokabularen wie ECLASS und einheitlichen Semantic-IDs wird die Interpretierbarkeit der Informationen gemäß Anforderung 8 und 9 ermöglicht. Wenn AAS innerhalb einer I4.0-Umgebung wie BaSyx 4.0 bereitgestellt werden, ist Anforderung 7 erfüllt. Wie beschrieben, setzt dies voraus, dass alle Beteiligten das Konzept der AAS verwenden, und einheitliche Teilmodelle und Semantik verwenden. Die aktuellen Entwicklungen zeigen jedoch, dass diese Voraussetzungen nicht vollständig erfüllt sind. Aktuelle Herausforderungen sind in der Abbildung 4.5 dargestellt.

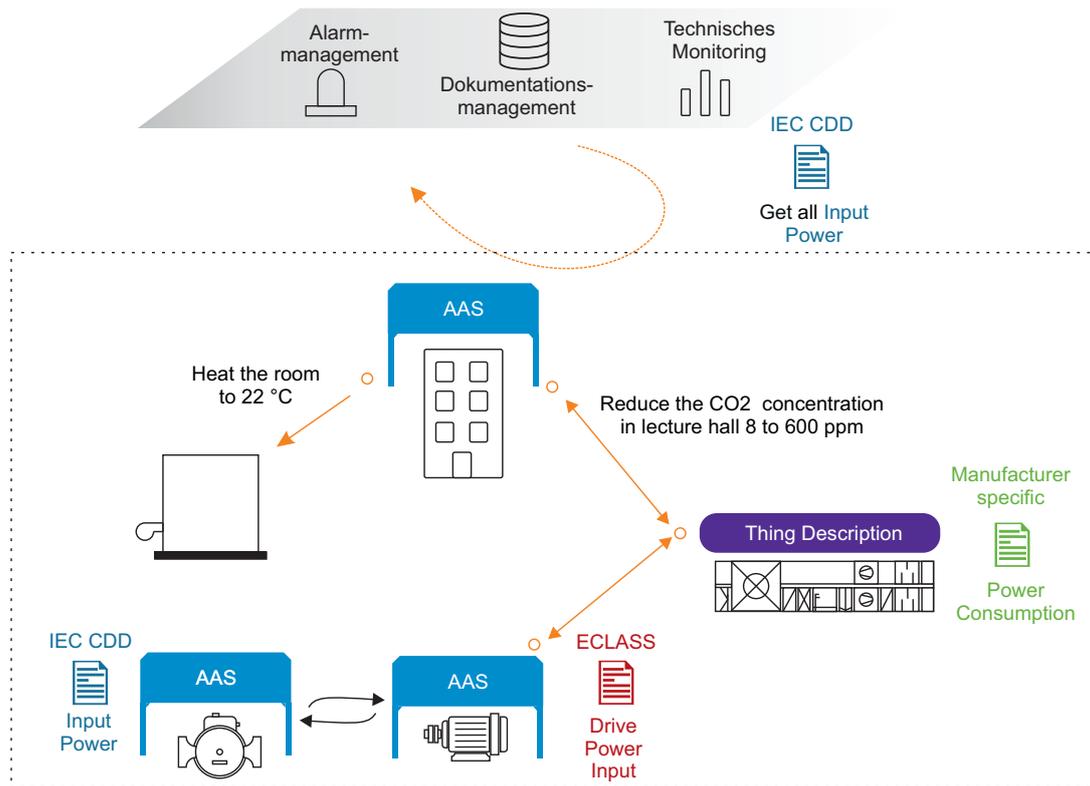


Abbildung 4.5: Aktuelle Herausforderungen im Bereich der I4.0 hinsichtlich des Erreichens von Interoperabilität

Verschiedene technische Komponenten sind innerhalb eines Gebäudes verbaut. Dabei werden einige durch digitale Zwillinge repräsentiert, andere Komponenten müssen jedoch noch mit digitalen Zwillingen ausgestattet werden. Diese Komponenten können aktuell nicht in eine I4.0-Umgebung eingebunden werden. Als Basis der digitalen Zwillinge werden unterschiedliche Metamodelle (AAS und Thing Description) verwendet. Diese müssen aufeinander abgebildet werden, um syntaktische Interoperabilität erreichen zu können. Darüber hinaus wird eine automatisierte Interaktion durch heterogene Semantik verhindert. Herstellerspezifische Beschreibungen, sowie unterschiedliche semantische Standards müssen aufeinander abgebildet werden, um semantische Interoperabilität zu erreichen. Diese ist ebenfalls erforderlich, um einen automatisierten Zugriff von Anwendungen wie dem Alarm- oder Dokumentationsmanagement zu ermöglichen. Die bestehenden Herausforderungen in der Praxis lassen sich in den folgenden fünf Punkten zusammenfassen und werden in den darauf folgenden fünf Abschnitten erläutert.

1. Abbildung unterschiedlicher digitaler Zwilling-Konzepte aufeinander.
2. Konzept zur abgestimmten Modellierung von Asset Informationen.
3. Abbildung heterogener semantischer Beschreibungen aufeinander.
4. Ausstattung bestehender technischer Systeme mit digitalen Zwillingen.

## 5. Realisierung autonomer Interaktionen digitaler Zwillinge.

In Kapitel 4.1 wurde erläutert, dass derzeit unterschiedliche Konzepte für digitale Zwillinge existieren, wie zum Beispiel AAS und Thing Description. Diese sind jedoch nicht kompatibel. Wenn innerhalb eines Netzwerks von verschiedenen Teilnehmern unterschiedliche Konzepte für digitale Zwillinge verwendet werden, ist eine automatisierte Interaktion zunächst nicht möglich. Damit dies realisiert werden kann, müssen Abbildungsvorschriften existieren. Der Vorteil bei der Abbildung zweier solcher Konzepte aufeinander besteht darin, dass sie nur einmal durchgeführt werden muss. Wenn es zwei Metamodelle gibt, wie das der AAS und Thing Description, muss einmal eine Abbildungsvorschrift erstellt werden, die in der Praxis angewendet werden kann. Offizielle Abbildungsvorschriften zwischen verschiedenen digitalen Zwillingskonzepten existieren derzeit noch nicht. Es gibt jedoch wissenschaftliche Veröffentlichungen, die sich diesem Thema widmen. In [160] und [117] wird beschrieben, wie die Thing Description auf ein Teilmodell der AAS abgebildet werden kann. In [118] wird ein Ansatz vorgestellt wie das DTDL-Metamodell auf das Modell der AAS abgebildet werden kann. Die Arbeiten zeigen, dass die Abbildung von verschiedenen Metamodellen aufeinander ein aktuelles Thema wissenschaftlicher Untersuchungen darstellt und in Zukunft weiter vorangetrieben wird.

Die Standardisierung von Teilmodellen für AAS wird derzeit vom IDTA organisiert (vgl. Kapitel 4.2.3). Zu diesem Zweck wurde ein Prozess entwickelt, der beschreibt, wie Anträge für neue Teilmodelle beim IDTA gestellt werden können und wie die Erstellung von Teilmodellen abläuft [161]. Dies ermöglicht eine schnelle Generierung und Veröffentlichung neuer Teilmodelle. Es gibt jedoch keinen Rahmenprozess für die einheitliche Ableitung von Teilmodellen. Anträge für die Erstellung eines Teilmodells können jederzeit beim IDTA gestellt werden. Nach Prüfung, ob sie thematisch mit existierenden Teilmodellen übereinstimmen, können diese erstellt werden, sofern keine Überlappung besteht. Es existiert jedoch kein strukturierter Prozess für die systematische Ableitung von Teilmodellen. Eine Einführung eines solchen Prozesses könnte Überlappungen vermeiden und zu einer besseren Strukturierung beitragen.

Darüber hinaus ist es unwahrscheinlich, dass sich alle Teilnehmer eines Wertschöpfungsnetzwerks auf gemeinsame Teilmodelle einigen können und diese verwenden werden. Stattdessen ist es realistischer anzunehmen, dass eine Vielzahl von Teilmodellen existiert, welche die gleichen Informationen von Assets beschreiben, jedoch nicht aufeinander abgestimmt sind. Insbesondere in einer globalisierten Welt, in der verschiedene Teilnehmer miteinander interagieren, wird dieses Szenario als realistisch angesehen. Es werden sowohl hersteller-spezifische Modelle als auch Modelle aus verschiedenen Industrien oder Ländern existieren und genutzt werden. Diese Tatsache zeigt sich bereits in der Handhabung von Teilmodellen innerhalb der IDTA. Im Online-Repository der IDTA werden die Teilmodelle veröffentlicht

und können dort mit verschiedenen Labeln klassifiziert werden. Einige Beispiele für solche Label sind: „Submodel template is the only one existing for a single domain“, „Submodel template is preferred and well accepted for a domain“ oder „Submodel template is maintained by a single company“. Diese Klassifizierungen sind hilfreich, um zu erkennen wie akzeptiert ein Teilmodell ist und ob es andere Teilmodelle in diesem Bereich gibt. Dies zeigt, dass aktuell schon verschiedene Teilmodelle (von einzelnen Unternehmen oder international) existieren, welche ähnliche Bereiche abdecken. Diese Entwicklung wird in den kommenden Jahren vermutlich weiter zunehmen. Werden verschiedene Teilmodelle definiert, erscheint es darüber hinaus unwahrscheinlich, dass dennoch die gleiche Semantik verwendet wird. Es existieren verschiedene Vokabulare, bspw. ECLASS oder IEC CDD, welche als Referenz zur semantischen Beschreibung von Assets verwendet werden können. Diese sind nicht aufeinander abgestimmt. Zusätzlich gibt es innerhalb der Standards überlappende semantische Beschreibungen (vgl. Tabelle 4.2).

Tabelle 4.2: Verschiedene Semantic-IDs des ECLASS Standards für die maximale Umgebungstemperatur

Semantic-ID	Name	Definition
0173-1#02-AAB906#006	max. Umgebungstemperatur während des Betriebs	größter Grenzwert der während des Aktivseins bestimmten Temperatur der das Betriebsmittel umgebenden Atmosphäre
0173-1#02-AAF525#003	max. Umgebungstemperatur während Betriebsphase	unter festgelegten Bedingungen bestimmte Temperatur der das Betriebsmittel umgebenden Atmosphäre, Zeitdauer, in deren Verlauf etwas stattfindet: Zeitraum des Aktivseins, größter Grenzwert einer physikalischen Größe, deren Wert nicht überschritten werden kann oder darf
0173-1#02-AAN313#004	obere Umgebungstemperaturgrenze	höchste Umgebungstemperatur, der ein Gerät ohne dauerhafte Einschränkung des Betriebsverhaltens standhalten kann oder muss
0173-1#02-AAW302#002	max. Umgebungstemperatur (bei Betrieb)	maximal zulässige Temperatur, der das Betriebsmittel umgebenden Atmosphäre während Betrieb

Die ECLASS-Merkmale der Tabelle beschreiben alle die maximale Umgebungstemperatur, der ein Asset während der Betriebsphase ausgesetzt werden darf. Die Semantic-IDs zeigen, dass sie in unterschiedlichen Produktgruppen definiert wurden, und deshalb nicht aufeinander abgestimmt sind. Das Beispiel zeigt, dass auch innerhalb des ECLASS-Standards heterogene semantische Beschreibungen existieren, welche jedoch den gleichen semantischen Inhalt haben. In Kapitel 4.2.3 wurden die Matching-Methoden dargestellt, welche aktuell für heterogene Semantik von AAS existieren. Mit diesen Methoden kann die hier aufgezeigte Problematik heterogener Semantik jedoch nicht aufgelöst werden. Die Verarbeitung heterogener Semantik, um semantische Interoperabilität von AAS zu erreichen, ist aktuell

Gegenstand der Forschung. In [162] wurde eine Modelltransformationssprache entwickelt, die es erlaubt Teilmodelle von AAS aufeinander abzubilden. Die Teilmodelle weisen dabei dieselben semantischen Inhalte auf, liegen jedoch in unterschiedlichen Formen vor, z.B. als herstellerspezifisches und als IDTA-Teilmodell. Voraussetzung für eine erfolgreiche Transformation ist, dass die beiden Teilmodelle bekannt sind und eine initiale Abbildung der Teilmodelle mittels der Sprache aufeinander stattfindet. Im späteren Betrieb kann diese Abbildung genutzt werden, um ein Teilmodell in ein anderes zu transformieren. Eine automatisierte Abbildung unbekannter, heterogener Vokabulare ist mit diesem Ansatz jedoch nicht möglich.

Eine weitere Herausforderung besteht darin, bereits bestehende technische Anlagen mit einem digitalen Zwilling auszurüsten. Existierende technische Anlagen der Industrie und von Gebäuden haben eine Lebensdauer von mehreren Jahrzehnten. Daher ist es erforderlich, ein Konzept zu entwickeln, das die Ausstattung dieser Anlagen mit digitalen Zwillingen ermöglicht, um auch sie in Wertschöpfungsnetzwerke integrieren zu können. Wenn nur neu zu erstellende Anlagen und Komponenten berücksichtigt würden, würde es noch Jahre dauern, um vollständig entwickelte I4.0-Lösungen zu schaffen, die einen Mehrwert bieten können. Die Betriebsdaten dieser existierenden Anlagen folgen in der Regel keinen semantischen Standards, sondern basieren auf hersteller- oder anwenderspezifischen Beschreibungen. Um diese nun als Informationsquelle für einen digitalen Zwilling der Anlagen zu erschließen, müssen diese klassifiziert und auf einen Standard abgebildet werden. In Kapitel 4.4 wurde der aktuelle Stand der Technik in Bezug auf die automatisierte Klassifizierung von Daten der GA dargelegt. Keiner dieser Ansätze erfüllt die Anforderung einer vollständig automatisierten Klassifizierung der Daten, die ohne manuellen Aufwand in Monitoring-Anwendungen verwendet werden können. Es mangelt an einem Konzept für die automatisierte Generierung digitaler Zwillinge für vorhandene Anlagen, sowie für die Klassifizierung ihrer Betriebsdaten zur automatisierten Weiterverarbeitung.

Eine weitere Herausforderung besteht in der Umsetzung vollständig autonomer, aufeinanderfolgender Interaktionen. Sofern die anderen Herausforderungen erfüllt sind, werden sowohl die syntaktische Interoperabilität als auch die semantische Interoperabilität gewährleistet. Dies bildet die Basis, um automatisierte Interaktionen zwischen digitalen Zwillingen zu ermöglichen. Um diese Interaktionen zu realisieren, muss das zu erwartende Verhalten der Zwillinge bekannt sein und der Interaktionspartner muss darauf reagieren können. Diese Anforderung bezieht sich auf die letzte Interoperabilitätsebene, die *verhaltensbezogene Interoperabilität*. Ein rein semantisches Verständnis der Funktionalitäten und Informationen der jeweiligen Partner reicht hierbei nicht aus. Das Ergebnis einer Interaktion muss für den anderen Partner interpretierbar sein. Dieser Bereich wird aktuell durch die Entwicklung einer I4.0-Sprache adressiert [122]. Um Interaktionen automatisiert durchführen zu können, werden basierend auf dieser Sprache Interaktionsprotokolle ent-

worfen, bspw. für Ausschreibungsverfahren [163], die festlegen, welche Aktionen innerhalb des Rahmens der Protokolle möglich sind. Die Definition der jeweiligen Protokolle für verschiedenste Anwendungsfälle erfordert jedoch einen hohen Aufwand, sodass in Zukunft weitere Möglichkeiten zur Realisierung der Interaktionen untersucht werden müssen.

Die Analyse des gegenwärtigen Stands der Technik im Bereich I4.0 bezüglich Interoperabilität zeigt, dass noch Herausforderungen zu bewältigen sind. Dies steht im Einklang mit der allgemeinen Diskussion aus Kapitel 2.4.3, in der beschrieben wurde, dass besonders im Bereich der semantischen und verhaltensbezogenen Interoperabilität noch offene Herausforderungen bestehen. In dieser Arbeit werden Konzepte entwickelt, welche speziell die Herausforderungen 2, 3 und 4 adressieren, um semantische Interoperabilität von digitalen Zwillingen zu erreichen. Diese Konzepte bilden die Grundlage für automatisierte Interaktionen von digitalen Zwillingen. Dabei werden sowohl die Bereiche der homogenen als auch heterogenen Semantik betrachtet. Im Bereich der homogenen Semantik wird ein Modell zur Ableitung von Teilmodellen entwickelt, das auf verschiedene Assets anwendbar ist (Kapitel 5). Durch dieses Modell können Teilmodelle entlang allgemeiner Lebenszyklusphasen abgeleitet werden, um eine Überlappung von Teilmodellen zu verhindern. Auf Basis dieses Konzepts werden Teilmodelle für Pumpen entwickelt. Im Bereich der heterogenen Semantik wird eine Methode entwickelt, die es ermöglicht, unterschiedliche Vokabulare und deren Elemente automatisiert aufeinander abzubilden (Kapitel 6). Zu diesem Zweck werden aktuelle Verfahren aus dem Bereich des NLP eingesetzt. Der entwickelte Ansatz erlaubt es digitalen Zwillingen, unbekannte Semantik zu verarbeiten und auf ihr eigenes zugrunde liegendes Vokabular abzubilden. Der Ansatz ist unabhängig vom Metamodell des Zwillings einsetzbar, da nur die Semantik und Metadaten auf Merkmalsebene verwendet werden. Die Methode wird in konkreten I4.0-Interfaces implementiert, sodass sie im Kontext von AAS verwendet werden kann. Darüber hinaus wird ein Konzept entwickelt, das es ermöglicht bestehende Anlagen in I4.0-Umgebungen zu integrieren. Hierzu werden aktuelle I4.0-Technologien (wie existierende Teilmodelle und der Einsatz von BaSyx) genutzt. Darauf aufbauend wird eine Methode entwickelt, um die heterogenen Betriebsdaten der Systeme zu verarbeiten und auf ein einheitliches Vokabular abzubilden (Kapitel 7). Ausgehend von nunmehr semantisch einheitlichen digitalen Zwillingen werden Strukturinformationen der technischen Systeme generiert, um sie für Erkundungsmechanismen zugänglich zu machen. Die automatisierte Zugriffsmöglichkeit und damit die Interaktion mit den digitalen Zwillingen wird im Anwendungsfall des TMon von Gebäuden veranschaulicht (Kapitel 8).

## 5 Semantische Interoperabilität - Homogene Semantik

Im Mittelpunkt dieser Arbeit steht die Entwicklung eines Konzepts zur Erlangung semantischer Interoperabilität zwischen AAS. Dabei werden sowohl die Fälle homogener Semantik als auch heterogener Semantik betrachtet. Beiden Ansätzen ist gemein, dass Informationen der AAS mittels Teilmodellen bereitgestellt werden. Der homogene Fall basiert auf dem Grundsatz, dass von verschiedenen Teilnehmern eines Wertschöpfungsnetzwerks standardisierte Teilmodelle verwendet werden. Gemeinsam mit den VDMA Fachverbänden „Pumpen und Systeme“ und „Kompressoren, Druckluft- und Vakuumtechnik“ wurden Teilmodelle für Pumpen entwickelt [164]. Diese wurden zunächst technologieunabhängig spezifiziert. In weiteren Schritten wurden die SEs der Teilmodelle in den ECLASS-Standard integriert. Darüber hinaus wurde eine *OPC UA Companion Specification* erstellt, welche die entwickelten Teilmodelle auf OPC UA abbildet [165]. Die Entwicklung der Teilmodelle wurde 2019 abgeschlossen und war somit eine der ersten abgeschlossenen Arbeiten zum Thema Teilmodelle im Kontext I4.0. Um zu verhindern, dass SEs und Teilmodelle, welche komponentenübergreifende Aspekte beschreiben, wie bspw. die Wartung von Komponenten, in späteren Standardisierungsarbeiten erneut definiert werden, wurde ein Vorgehensmodell zur Ableitung von Teilmodellen spezifiziert. Dieses ist auf verschiedene Assets anwendbar und kann als einheitlicher Mechanismus zur Ableitung von Teilmodellen verwendet werden. Auf Basis dieses Vorgehensmodells wurden offizielle Teilmodelle für Pumpen entwickelt. Darüber hinaus wurden innerhalb dieser Arbeit weitere Teilmodelle im Kontext von Gebäuden spezifiziert, um den Anwendungsfall der Arbeit abzudecken. Die folgenden Unterkapitel beschreiben zunächst das allgemeine Vorgehensmodell und im Anschluss kurz die entwickelten Teilmodelle für Pumpen. Die weiteren erstellten Teilmodelle dieser Arbeit, welche sich auf Gebäude beziehen, sind in Kapitel 7 beschrieben.

### 5.1 Entwurf von Industrie 4.0 Teilmodellen – Vorgehensmodell

Die beschriebene Vorgehensweise ist auf verschiedene Assets anwendbar. Das Modell basiert auf einer prozess- und lebenszyklusorientierten Modellierung von Assetfunktionen. Des Weiteren wird eine Klassifikation vorgeschlagen, mit welcher Assets in Gruppen zusammengefasst und hierarchisch strukturiert werden können. Ziel ist es, dass sowohl physikalische Assets wie Schrauben oder Ventilatoren, als auch Assets wie Software abgebildet werden können. Die gewählte Orientierung an grundlegenden Funktionen bildet

die Basis, um alle während des Lebenszyklus anfallenden Informationen durchgängig zu verknüpfen. In das Modell wurden beispielsweise mit der Funktion *Systemanforderungen spezifizieren* Aspekte übertragen, die über das Wertschöpfungsnetzwerk des Herstellers hinausgehen. Auf diese Weise können Anforderungen des Käufers in die Entwicklung des Typs einfließen.

### 5.1.1 Lebenszyklusphasen als Grundlage von Teilmodellen

In einer der drei Achsen des in Kapitel 4.2.1 beschriebenen RAMI 4.0 ist der Lebenszyklus dargestellt. Hierbei wird grundsätzlich in die Typ- und Instanz-Phase eines Assets unterschieden. Aufbauend auf dieser allgemeinen Unterscheidung soll, innerhalb des entwickelten Modells, der Lebenszyklus eines Assets an Hand grundlegender Funktionen beschrieben werden. In der ISO 15288 [166] werden Prozesse definiert, welche den Lebenszyklus technischer Systeme beschreiben. ISO 12207 [167] wiederum spezifiziert Prozesse mit dem Fokus auf Software, die sich jedoch kaum von denen der ISO 15288 unterscheiden. Die Prozesse bilden die Grundlage für die Spezifikation der Teilmodelle. Wie bereits beschrieben, soll eine prozessorientierte Modellierung von Assetfunktionen als Basis der Teilmodelle dienen. Aus diesem Grund wurden die Prozesse auf grundlegende Assetfunktionen abgebildet (vgl. Abbildung 5.1).

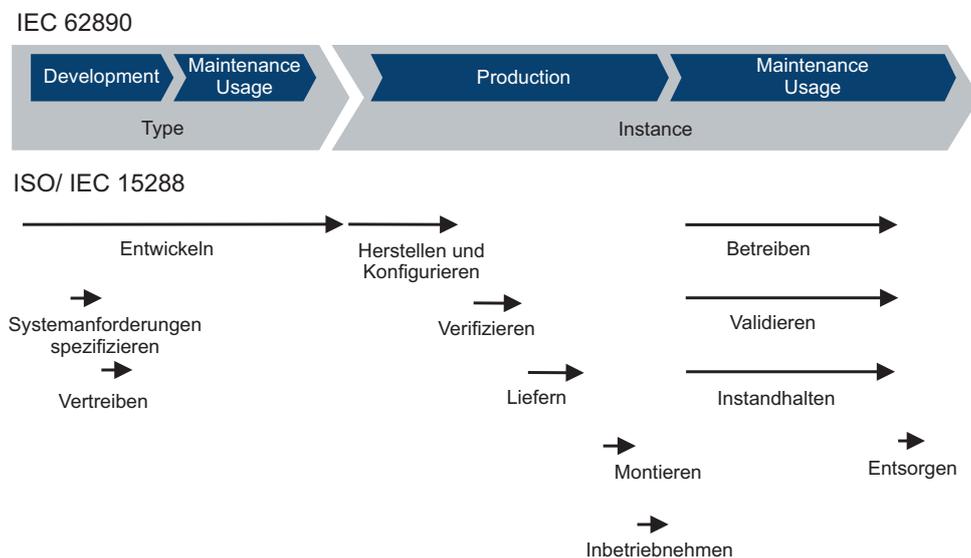


Abbildung 5.1: Zugeordnete Lebenszyklusprozesse der ISO 15288 zu dem Lebenszyklusmodell aus RAMI 4.0

Die grundlegenden Funktionen werden mit einer kurzen Definition nach [166] erläutert. Anhand von SEs werden Beispiele für den möglichen Inhalt der Teilmodelle gegeben.

- **Entwickeln:** Es sollen ausreichend detaillierte Informationen über das Produkt bereitgestellt werden, SEs über den Produkttyp werden bereitgestellt.

Beispiele für SEs: Maximal zulässige Dauerdrehzahl, Ventilator-Wirkungsgrad, Maximaler Ventilatorleistungsbedarf, Eintrittsquerschnitt

- Systemanforderungen spezifizieren: Die Anforderungen des Käufers an das System werden spezifiziert, es wird festgelegt, welche Anforderungen ein System erfüllen muss.

Beispiele für SEs: Maximaler Eintrittsdruck, Auslegungsförderstrom

- Vertreiben: Ein Produkt oder eine Dienstleistung soll vertrieben werden.

Beispiele für SEs: Name des Verkäufers, Postleitzahl der Käuferanschrift, Nettopreis des Artikels

- Herstellen und konfigurieren: Ein Produkt wird hergestellt, welches den Anforderungen des Käufers entspricht, auf Basis dieser Anforderungen kann eine Vorkonfiguration des Produktes stattfinden.

Beispiele für SEs: Seriennummer, Herstelldatum

- Verifizieren: Es soll sichergestellt werden, dass das Produkt den spezifizierten Anforderungen entspricht.

Beispiele für SEs: Prüfdatum, Garantierte Daten (Volumenstrom etc.)

- Liefern: Dem Käufer wird ein Produkt oder eine Dienstleistung bereitgestellt.

Beispiele für SEs: Lieferdatum, Stadt der Lieferanschrift

- Montieren: Das Produkt wird montiert, verschiedene Produkte werden zu einem System zusammengesetzt.

Beispiele für SEs: Zulässiges Anzugsmoment, Minimale Schraubenlänge

- Inbetriebnehmen: Das Produkt/ System wird in geordneter und geplanter Weise in den Betriebsstatus überführt.

Beispiele für SEs: Regelungsart, Absenkbetrieb einstellen

- Betreiben: Das System wird zur Bereitstellung seiner Dienste verwendet.

Beispiele für SEs: Volumenstrom, Leistungsaufnahme des Antriebs

- Validieren: Überwachung des Systems, um nachzuweisen, dass das System seine Ziele und Anforderungen erfüllt.

Beispiele für SEs: Verschiedene Indikatoren aus Betriebsdaten im Vergleich zu Auslegungs- bzw. Herstellerdaten

- Instandhalten: Die Fähigkeit des Systems, einen Dienst bereitzustellen, soll aufrechterhalten werden.

Beispiele für SEs: Verfügbarkeit, Zuverlässigkeit, durchschnittliche Betriebszeit zwischen Ausfällen, Wartungszyklen

- Entsorgen: Verwendung des Systems oder Systemelements soll beendet werden und die ersetzten Elemente entsorgt werden.

Beispiele für SEs: Verwendete Werkstoffe

In dem vorgestellten Lebenszyklusmodell kann jedes physikalische Asset und Software verortet werden. Die beschriebenen Funktionen bilden die Grundlage der Teilmodelle, welche allgemein gehalten sind, um auf jedes Asset angewendet werden zu können. Für die tatsächliche Ausprägung der Teilmodelle können diese Funktionen in weitere Unterfunktionen unterteilt werden, um die speziellen Anforderungen eines Assets bzw. einer Branche zu erfüllen. In Abbildung 5.2 ist beispielhaft dargestellt, wie Teilmodelle von den grundlegenden Funktionen *Betreiben* und *Instandhalten* abgeleitet werden können.

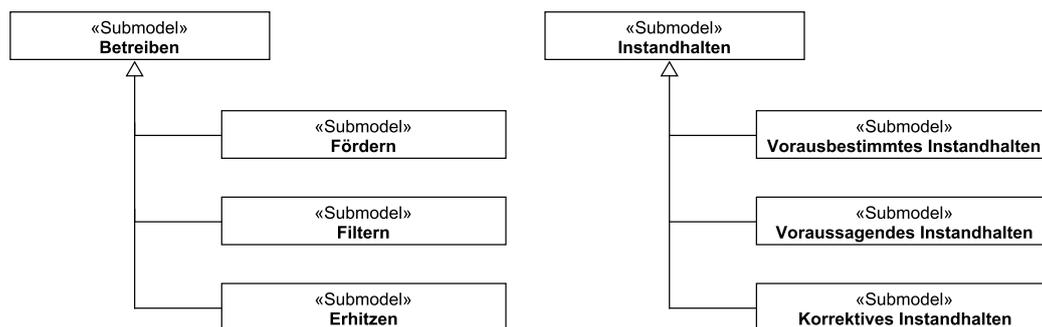


Abbildung 5.2: Beispiele für abgeleitete Teilmodelle von den grundlegenden Assetfunktionen

Grundlage für die von der Funktion *Betreiben* abgeleiteten Teilmodelle können Normen wie die DIN 8580 [168], in welcher Fertigungsverfahren klassifiziert werden, oder Funktionen der Verfahrenstechnik sein. Für Teilmodelle der *Instandhaltung* kann wiederum die DIN EN 13306 [169] als Basis verwendet werden, in der grundlegende Begriffe der *Instandhaltung* definiert werden.

### 5.1.2 Klassifikation von Assets

Während sich der erste Teil des Modells auf die lebenszyklusorientierte Modellierung von Assetfunktionen bezieht, beinhaltet der zweite Teil des Modells die Klassifikation von Assets. Die einzelnen Teilmodelle setzen sich aus SEs zusammen. Ziel ist es, dass SEs,

welche das gleiche beschreiben, möglichst nur einmal standardisiert werden. So ist bspw. die *Maximal zulässige Umgebungstemperatur* ein SE, welches Verwendung bei unterschiedlichen Assets findet. Des Weiteren gibt es SEs, die für eine Gruppe von Assets (wie bspw. Pumpen) relevant sind und SEs, welche ausschließlich für einen bestimmten Typen einer solchen Gruppe (bspw. Kreiselpumpen) definiert werden.

Zur Klassifikation von Assets bietet sich der *International Standard Industrial Classification of All Economic Activities (ISIC)* an, der zur Gliederung von Wirtschaftsbereichen und Industriezweigen verwendet wird [170]. Dieser Standard ordnet jede Branche einer von 21 Hauptgruppen zu, welche wiederum in Untergruppen aufgeteilt sind. Durch die Untergruppen entsteht eine vierstufige Hierarchie, welche beispielhaft in Abbildung 5.3 dargestellt ist.

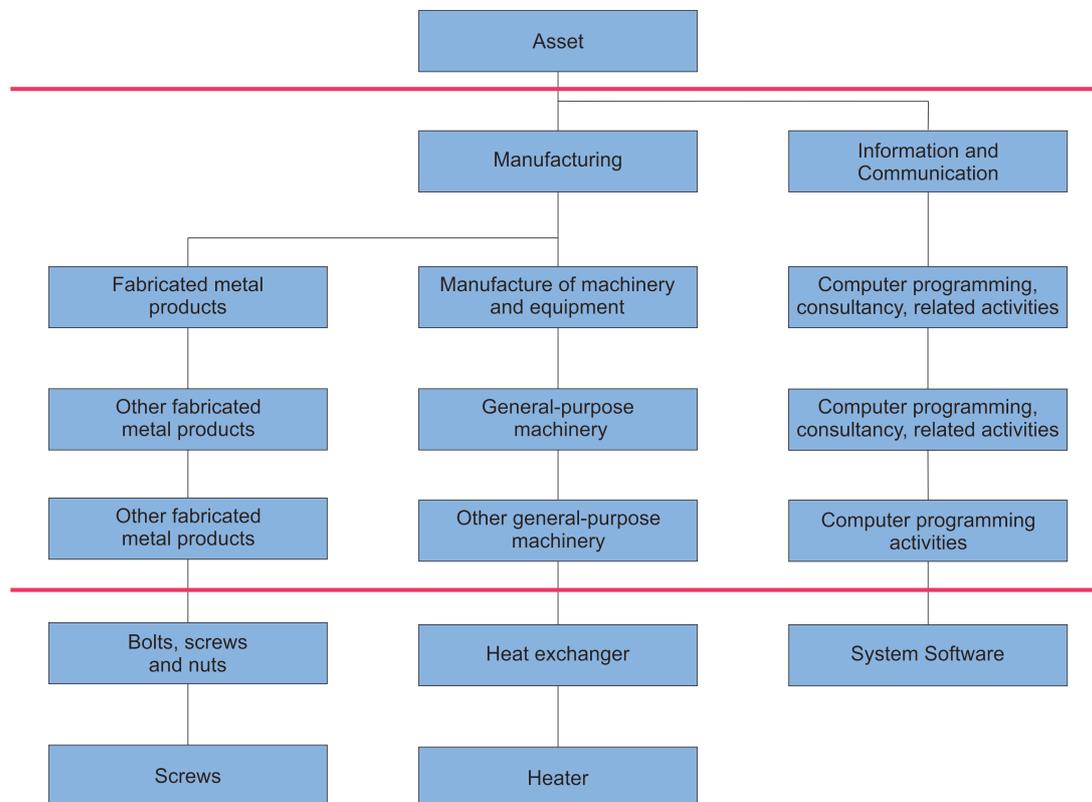


Abbildung 5.3: Beispiel einer Klassifikation von Assets

Die Klassifikation nach ISIC wurde nach oben sowie unten erweitert (dargestellt durch die roten Linien). Oberhalb der Hauptgruppen wurde das allgemeine Asset eingeführt. Unterhalb der letzten Unterkategorie wurden weitere Klassen eingeführt, um die Hierarchisierung der Assets genauer ausdrücken zu können. Unter dem allgemeinen Asset beginnt die Klassifikation der Güter. Die zwei abgebildeten Hauptgruppen sind das *Verarbeitende Gewerbe* (Manufacturing) und *Information und Kommunikation* (Information and Communication). Ausgehend vom verarbeitenden Gewerbe sind die Unterkategorien *Maschinenbau* (Manufacture of machinery and equipment) und *Metallerzeugnisse* (Fabricated metal products)

dargestellt. Ausgehend von der Kategorie Maschinenbau wird die Unterkategorie *Nicht wirtschaftszweigspezifische Maschinen* (General-purpose machinery) definiert. Die letzte Stufe der ISIC führt die Kategorie *Andere nicht wirtschaftszweigspezifische Maschinen* (Other general-purpose machinery) ein. Die untersten Stufen der ISIC reichen jedoch nicht aus, um eine hinreichend genaue Klassifizierung der Assets zu gewährleisten. Die Kategorien müssen hierzu weiter unterteilt werden. Dies soll am Beispiel der Kategorie *Andere nicht wirtschaftszweigspezifische Maschinen* beschrieben werden. Diese Kategorie umfasst Komponenten wie Ventilatoren oder Wärmetauscher (Heat exchanger). Wärmetauscher können des Weiteren in Kühler oder Erhitzer eingeteilt werden, in der Abbildung ist ein Erhitzer (Heater) dargestellt. In Abbildung 5.3 sind außerdem die beiden Assets Schrauben (Screws) und Software (System Software) als Beispiele für weitere Assets der anderen Zweige abgebildet.

### 5.1.3 Kombination der grundlegenden Assetfunktionen und Assetklassifikation

Im folgenden Abschnitt werden die beiden Teile des Vorgehensmodells zusammengeführt. Durch die Kombination der grundlegenden Prozesse und der Klassifizierung entstehen die Voraussetzungen für die Wiederverwendbarkeit von Teilmodellen und SEs (vgl. Abbildung 5.4).

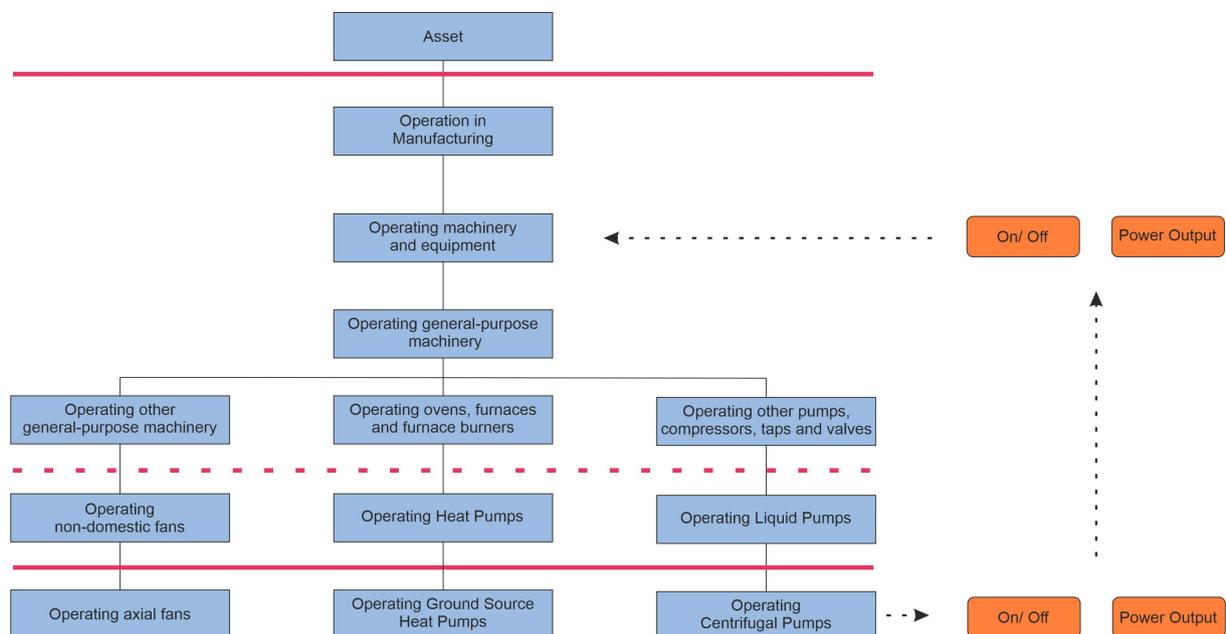


Abbildung 5.4: Wiederverwendbarkeit von SEs

Beispielhaft sind auf der untersten Ebene die Prozesse *Ventilatoren-, Erdwärmepumpen- und Kreiselpumpen betreiben* abgebildet. Bei der Definition von SEs für Teilmodelle wird innerhalb dieses Vorgehensmodells das *Bottom-up-Prinzip* verwendet. Bei jedem zu definierenden SE wird analysiert ob es nur auf der untersten Stufe, hier bspw. Kreiselpumpen,

verwendet werden kann oder ebenfalls auf einer höheren Stufe. Dies wird beispielhaft anhand der Spezifikation der Teilmodelle für Pumpen erläutert. Die Eigenschaften für Kreiselpumpen, Verdrängerpumpen und verschiedene Vakuumpumpen wurden definiert und anschließend analysiert, welche Eigenschaften übergreifend für alle Flüssigkeits- und Vakuumpumpen gelten. Im nächsten Schritt musste definiert werden, welche Eigenschaften übergreifend für die Gruppe *Nicht wirtschaftszweigspezifische Maschinen* gelten. Die Abbildung enthält die SEs *On/Off* und *Power Output* als Beispiele. Diese beiden sind keine pumpenspezifischen Eigenschaften, sondern gelten für alle elektrisch betriebenen Anlagen. Sie sind gute Beispiele für Eigenschaften, die einer höheren Klassifizierungsebene zugeordnet werden können und daher für mehrere Anlagen gültig sind.

## 5.2 Teilmodelle für Pumpen

Auf Basis des entwickelten Vorgehensmodells wurden Teilmodelle für Pumpen modelliert.

### 5.2.1 Vorgehensweise Entwicklung von Teilmodellen

Die Teilmodelle wurden auf Basis von drei Anwendungsfällen entwickelt: Konfiguration, Betrieb und Instandhaltung. Diese wurden im Lebenszyklusmodell verortet, um die entsprechenden Teilmodelle abzuleiten (vgl. Abbildung 5.5.)

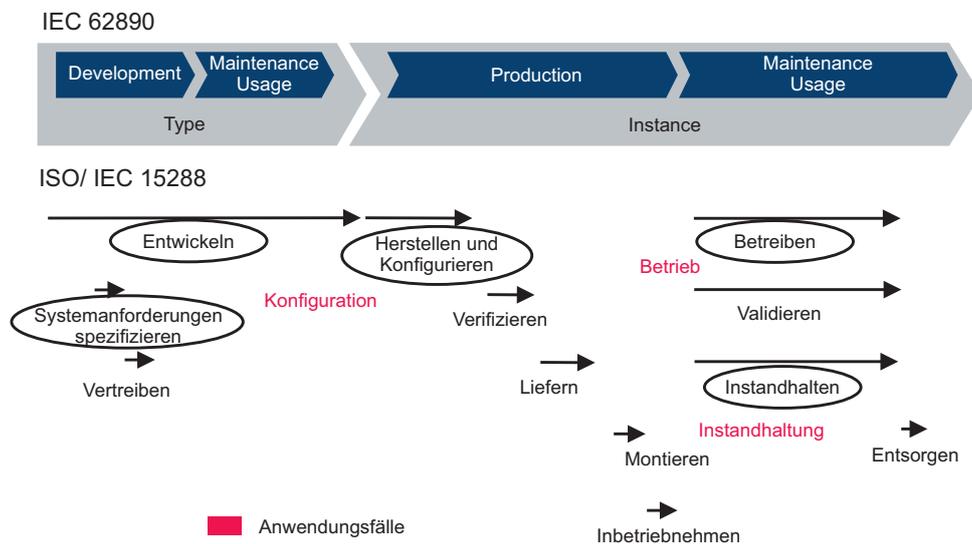


Abbildung 5.5: Verortung der Anwendungsfälle im Lebenszyklusmodell

Die Abbildung zeigt, dass der Anwendungsfall Konfiguration den grundlegenden Funktionen *Entwickeln*, *Systemanforderungen spezifizieren* und *Herstellen und Konfigurieren* zugeordnet wird. Die Anwendungsfälle Betrieb und Instandhaltung werden den Funktionen *Betreiben* bzw. *Instandhalten* zugeordnet. Die grundlegende Funktion *Betreiben* wird

jedoch in weitere Unterfunktionen unterteilt, da sie nicht ausreicht, um den Betrieb in seiner Gänze zu beschreiben. Die Funktionen sind in Abbildung 5.6 dargestellt.

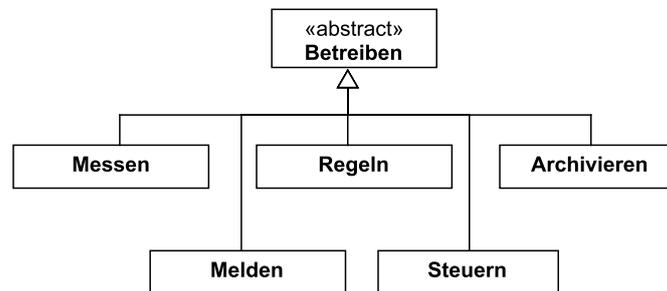


Abbildung 5.6: Abgeleitete Funktionen von Betreiben

Bei den Funktionen Messen, Melden, Regeln, Steuern und Archivieren handelt es sich um generische Funktionen, die für andere Assets ebenfalls anwendbar sind. Je nach Asset können dann ausgewählte Funktionen als Basis für Teilmodelle dienen. Aus den grundlegenden Funktionen werden die folgenden Teilmodelle für Pumpen abgeleitet:

- Design: Es sollen ausreichend detaillierte Daten und Informationen über das Produkt bereitgestellt werden, Properties über den Produkttyp werden bereitgestellt
- System-Requirements: Die Anforderungen des Käufers an das System werden spezifiziert, es wird festgelegt welche Anforderungen ein System erfüllen muss
- Implementation: Ein Produkt wird hergestellt, welches den Anforderungen des Käufers entspricht, auf Basis dieser Anforderungen kann eine Vorkonfiguration des Produktes stattfinden
- Measurements: Messungen der physikalischen Größen des Systems werden durchgeführt
- Control: Das System wird geregelt
- Actuation: Die Ansteuerung und Betätigung des Systems wird durchgeführt
- Signals: Meldungen des Systems werden generiert
- Events: Ereignisse können ausgehend von dem System generiert werden und dienen der Überwachung von diesem
- Archiving: Eigenschaften referenzierter Objekte können nach vorgegebenen Kriterien überwacht und in einem internen Speicher abgelegt werden

- Condition-Based-Maintenance: Instandhaltungsstrategie, bei der die Instandsetzung oder der Austausch von Teilen oder Komponenten aufgrund einer Ausfallvorhersage auf der Basis des Zustands der Anlage erfolgt
- Preventive-Maintenance: Instandhaltungsstrategie, bei der die Instandsetzung oder der Austausch von Teilen oder Komponenten nach einem festgelegten Zeitplan erfolgt
- Breakdown-Maintenance: Instandhaltungsstrategie, bei der die Instandsetzung oder der Austausch von Teilen oder Komponenten nach Ausfall der Anlage erfolgt
- Supervision: Das System wird hinsichtlich Fehlerzuständen im Betrieb überwacht

Die Inhalte und Beschreibungen der einzelnen Teilmodelle sind in den folgenden Kapiteln näher beschrieben.

### 5.2.2 Teilmodelle Design, System-Requirements und Implementation des Anwendungsfalls Konfiguration

Das Teilmodell *Design* enthält die Herstellerangaben zur Pumpe. Hierbei handelt es sich um Daten, welche die Pumpe und ihre Einschränkungen hinsichtlich der Arbeitsbedingungen beschreiben, unter denen sie verwendet werden kann. Beispielsweise werden die maximal zulässige Umgebungstemperatur oder der maximal zulässige Gehäusebetriebsdruck definiert. Diese Eigenschaften geben Grenzen an, die für den Betrieb der Pumpe gelten. Weitere Eigenschaften, wie beispielsweise die optimale Förderhöhe, beschreiben die Arbeitsbedingungen, unter denen die Pumpe optimal funktioniert, sowohl unter Normalbedingungen als auch speziellen Bedingungen. Diese SEs sind größtenteils konstant und verändern sich während des gesamten Lebenszyklus der Pumpe nicht. Ein Beispiel für die angebotenen Steuerungsmodi ist in Abbildung 5.7 dargestellt.

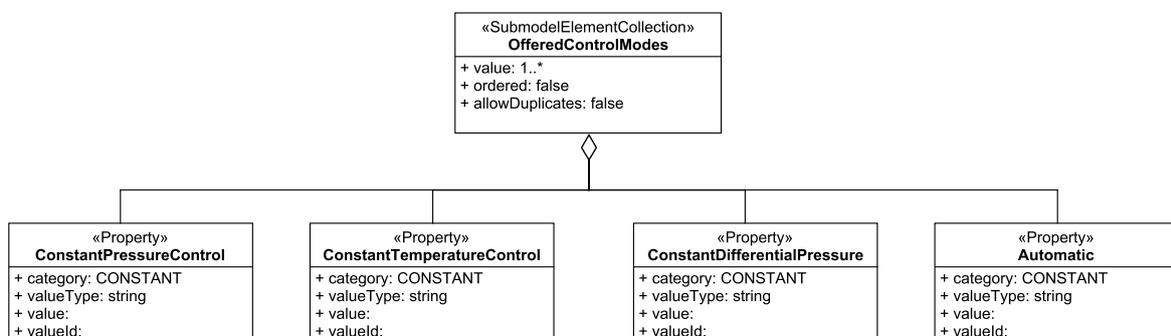


Abbildung 5.7: Mögliche Regelungsarten für Pumpen

In der Abbildung sind vier der möglichen elf Modi dargestellt.

Das Teilmodell *System-Requirements* enthält die Informationen des Käufers oder Planers. Diese Eigenschaften beschreiben den Anwendungsfall, in dem die Pumpe verwendet werden soll, sowie die vorherrschenden Arbeitsbedingungen für diesen Anwendungsfall. Hierbei werden Elemente wie der maximale Durchfluss oder die gewünschte Steuerungsmethode berücksichtigt.

Das Teilmodell *Implementation* beinhaltet die nötigen Funktionen und Eigenschaften für die Bereitstellung einer vorkonfigurierten Pumpe. Dieses Teilmodell wird auf Basis der Anforderungen des Teilmodells *System-Requirements* und den technischen Daten einer Pumpe des Teilmodells *Design* generiert.

Für die Spezifikation der drei Teilmodelle wurden grundlegende Normen aus dem Bereich der Pumpentechnik verwendet [171–175]. Wo möglich, wurden Namen und Definition der spezifizierten SEs aus diesen Normen entnommen.

### 5.2.3 Teilmodelle Condition-Based-, Preventive- und Breakdown-Maintenance des Anwendungsfalls Instandhaltung

Der Anwendungsfall der Instandhaltung wurde in drei Instandhaltungsstrategien unterteilt: *Condition-Based-Maintenance*, *Preventive-Maintenance* und *Breakdown-Maintenance*. Es gibt verschiedene Standards, welche die Grundlagen für die Instandhaltung beschreiben. Für den Wortschatz der Instandhaltung existieren auf ISO-Ebene die Norm ISO 13372 [176] und auf EN-Ebene die Norm EN 13306 [169]. Die Normen beschreiben im Grundsatz dieselben Instandhaltungsstrategien, verwenden jedoch unterschiedliche Bezeichnungen. Da internationale Standards verwendet werden sollten, um eine möglichst breite internationale Anerkennung der Begriffe zu erhalten, folgen die Teilmodelle den Bezeichnungen der ISO-Norm und nicht der EN-Norm. Die Eigenschaften der Teilmodelle wurden spezifisch für die jeweilige Instandhaltungsstrategie ausgeprägt. Es gibt jedoch auch SEs, die für alle Strategien von Bedeutung sind. Aus diesem Grund wurde die abstrakte Klasse *Maintenance* eingeführt, von der die drei Teilmodelle erben. Die Elemente der abstrakten Klasse gelten für alle drei Teilmodelle und umfassen Eigenschaften wie den Zustand der Komponente, die mittlere Betriebszeit zwischen Ausfällen oder die Ausfallrate.

Das Teilmodell *Condition-Based-Maintenance* (zustandsbasierte Wartungsstrategie), definiert Eigenschaften zur Beschreibung der erwarteten Lebensdauer der Komponenten. Das Teilmodell ist unabhängig von Pumpen und kann auf jedes andere Asset übertragen werden. In diesem Modell werden SEs wie die Zuverlässigkeit oder die Verfügbarkeit definiert, jedoch nicht wie diese Werte intern berechnet werden.

Das Teilmodell *Preventive-Maintenance* (präventive Wartungsstrategie) beschreibt die SEs der Wartungsstrategie für eine in regelmäßigen Abständen stattfindende Wartung. Die enthaltenen Eigenschaften gelten ebenfalls für verschiedene Komponenten und sind nicht spezifisch für Pumpen. Der Bereich der präventiven Wartung ist umfangreich und in verschiedenen Normen ausführlich beschrieben, bspw. [177]. Ein allgemein gültiges Teilmodell sollte zentral und Asset übergreifend definiert werden. Das Teilmodell *Preventive-Maintenance* beschreibt deshalb nur grundlegende SEs zur Beschreibung der vorbeugenden Wartung und wird durch ein umfassenderes Teilmodell ersetzt, sobald dieses von zentraler Stelle definiert wurde.

Das Teilmodell *Breakdown-Maintenance* beschreibt Eigenschaften, die den Ausfall von Pumpen definieren. Im Gegensatz zu den anderen beiden Teilmodellen beschränkt sich dieses ausschließlich auf Pumpen. Zum einen werden Eigenschaften wie die Anzahl der Pumpenausfälle und der Schweregrad beschrieben. Zum anderen werden SMCs definiert, die auf den spezifischen Ausfall der Pumpe eingehen. Basis hierfür ist das VDMA Einheitsblatt 24223, welches verschiedene Bausteine definiert, die unterschiedliche Ursachen für Pumpenausfälle beinhalten [178]. Diese wurden in das Teilmodell aufgenommen und innerhalb von SMCs kombiniert.

#### 5.2.4 Teilmodelle des Anwendungsfalls Betrieb

Der Anwendungsfall Betrieb bildet größtenteils die Inhalte des VDMA Einheitsblattes 24223 ab [178]. Das Blatt dient der Standardisierung grundlegender Funktionen und Informationen von Vakuum- und Flüssigkeitspumpen im Betrieb, unabhängig von kommunikationsspezifischen Eigenschaften [178]. Einige Inhalte aus [178], wie *Events*, wurden an internationale Standards angepasst. Der Anwendungsfall Betrieb wird durch die Teilmodelle *Measurements*, *Events*, *Archiving*, *Control*, *Actuation* und *Signals* abgedeckt, welche in den folgenden Abschnitten beschrieben werden.

Die abstrakte Klasse *Measurements* beinhaltet sämtliche, für eine Messung erforderlichen Eigenschaften (vgl. Abbildung 5.8). Alle konkreten Messungen erben von dieser Klasse und folgen ihrer Struktur.

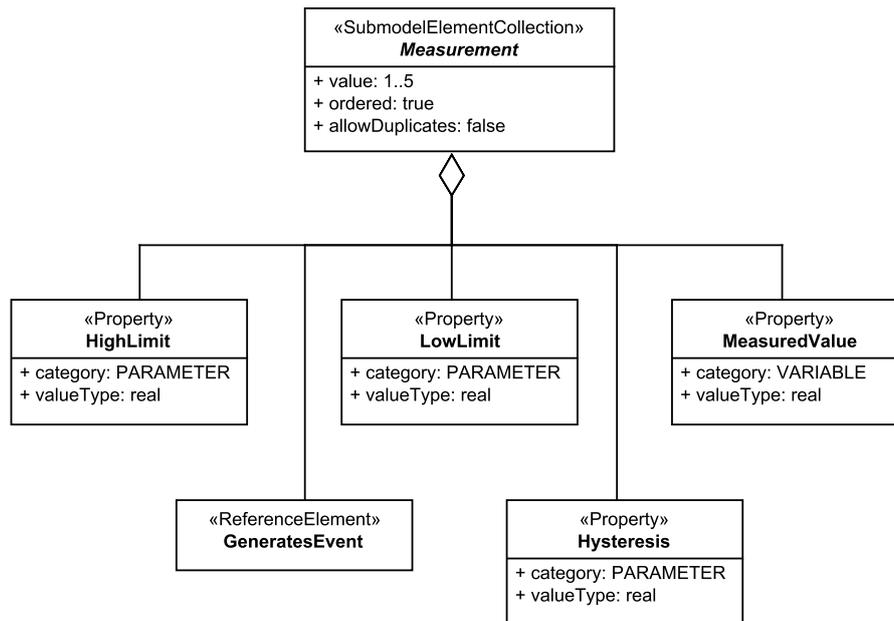


Abbildung 5.8: Abstrakte Klasse Measurements

Eine Messung setzt sich insgesamt aus bis zu fünf SEs zusammen, von denen ausschließlich der *Measured-Value* verpflichtend ist. Zusätzlich können Obergrenzen und Untergrenzen definiert werden, bei deren Überschreitung oder Unterschreitung ein Ereignis ausgelöst wird. Zur Auslösung des Ereignisses ist das Referenzelement *Generates-Event* erforderlich. Es wird unterschieden zwischen Eigenschaften, die direkt gemessen werden (*Measured*) und solchen, die aus Messungen abgeleitet werden (*Derived*). Die Messwerte umfassen sämtliche derzeit verfügbaren und abgeleiteten Werte für Pumpen. Insgesamt wurden 63 Messungen definiert, wie bspw. die Durchflussmenge, die Förderhöhe der Pumpe oder die Anzahl der Starts. Das Teilmodell *Measurements* setzt sich aus allen möglichen Messungen zusammen, die eine bestimmte Pumpe durchführen kann.

In [11] wird beschrieben, dass *Events* mögliche SEs von Teilmodellen sind, die Spezifikation sich jedoch noch in einer experimentellen Phase befindet. Aus diesem Grund wurden für die Teilmodelle für Pumpen die OPC UA Definitionen von Events verwendet. OPC UA wurde ausgewählt, da in einem zweiten Schritt die Teilmodelle auf OPC UA abgebildet werden (nicht in dieser Arbeit beschrieben). Der Aufbau des *Base-Event-Type* von OPC UA wird ausführlich in [179] und [180] beschrieben. Insgesamt wurden acht Events definiert, die von verschiedenen SEs der Teilmodelle ausgelöst werden können.

Über das Teilmodell *Archiving* können die Werte eines Objekts gespeichert werden. Die zu archivierenden Werte werden gemeinsam mit einem Zeitstempel in einer Liste gespeichert, die in einem internen Speicher abgelegt ist. Das Teilmodell ermöglicht die Beobachtung und Analyse der Trends von Objekten. Die Definition des Teilmodells wurde gemäß der

Norm ISO 16484-5 erstellt [136]. Die Elemente sind in Abbildung 5.9 dargestellt.

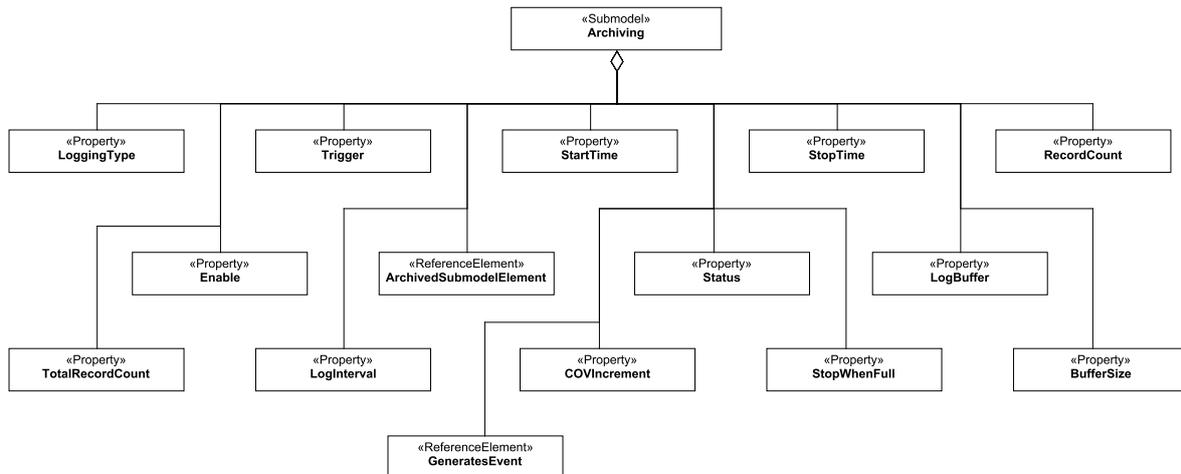


Abbildung 5.9: Teilmodell Archiving

Das Property *Logging-Type* gibt an, auf welche Weise die Werte des zu archivierenden Objekts erfasst werden, was anhand der objektspezifischen Eigenschaften ermittelt wird. Ein Verweis auf das aufzuzeichnende Property erfolgt mittels des Referenzelements *Archived-Submodel-Element*. Durch die Properties *Enable*, *Start-Time* und *Stop-Time* kann die Aufzeichnung von Werten gestartet und gestoppt werden und es kann zurückgemeldet werden, ob aktuell Werte aufgezeichnet werden. Die weiteren Properties enthalten Details zum Speicher, zur möglichen und tatsächlichen Anzahl von Einträgen sowie zur Handhabung bei vollem Speicher. Darüber hinaus ist es möglich, mit dem Referenzelement *Generates-Event* ein Ereignis auszulösen.

Das Teilmodell *Control* wurde entwickelt, um eine präzise Regelung der Pumpe zu ermöglichen. Hierzu wurden die Variablen eines Regelkreises gemäß DIN IEC 60050-351 [181] an die Struktur der AAS angepasst und in das Teilmodell *Control* überführt. Sämtliche für eine Regelung benötigten Variablen, wie Regel- und Rückführgröße, wurden in das Modell aufgenommen. Zudem wurden Eigenschaften definiert, welche die aktuelle Steuerungsart der Pumpe anzeigen. Im Falle eines Fehlers kann über ein Referenzelement ein Ereignis ausgelöst werden.

Die abstrakte Klasse *Actuation* beinhaltet Eigenschaften, welche die Ausführung einer Aktion eines Objekts beschreiben. Die auf Basis dessen spezifizierten Teilmodelle *Pump-Actuation*, *Bypass-Valve-Actuation* und *Throttle-Valve-Actuation* erben diese Eigenschaften. Zum Teilmodell *Pump-Actuation* wurden weitere spezifische Eigenschaften hinzugefügt, die über die allgemeinen Actuation-Eigenschaften hinausgehen.

Das Teilmodell *Signals* wurde eingeführt, um eine Rückmeldung über den aktuellen Status von Objekten zu liefern. Hierzu wurde die abstrakte Klasse *Discrete-Input* eingeführt,

dargestellt in Abbildung 5.10, von der alle abgeleiteten Discrete-Inputs erben.

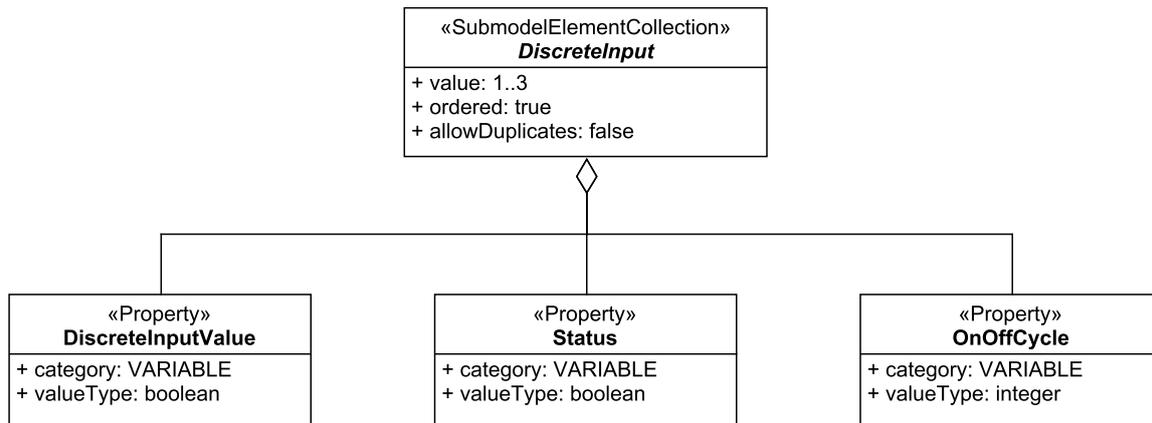


Abbildung 5.10: SMC Discrete Input

Die Sammlung Discrete-Input setzt sich aus bis zu drei Properties zusammen. Jede Sammlung muss über das Property *Discrete-Input-Value* verfügen, welches den aktuellen Wert des Eingangs enthält. Optional sind außerdem die Properties *Status* und *On-Off-Cycle*. Insgesamt wurden 28 diskrete Eingänge definiert, wie beispielsweise *Pump-Activation* oder *Pump-Speed-Max*.

### 5.2.5 Integration der Teilmodellinhalte in ECLASS

Zum Zeitpunkt der Erstellung der Teilmodelle für Pumpen im Jahr 2019 war der IDTA noch nicht gegründet. Teilmodelle, die zu diesem Zeitpunkt erstellt wurden, konnten noch nicht als offizielle I4.0-Teilmodelle anerkannt werden. Um mit den Teilmodellen für Pumpen dennoch Anforderungen der AAS zu erfüllen, wurden sie in den ECLASS-Standard integriert. Auf diese Weise haben die Teilmodelle für Pumpen und die darin spezifizierten SEs eine eindeutige IRDI erhalten. Diese kann als Semantic-ID referenziert werden. Bei der Spezifikation der Teilmodelle wurde die ISIC-Klassifikation verwendet. Die SEs wurden nicht alle einem Pumpentyp zugeordnet, sondern an die hierarchische Klassifikation der ISIC angepasst. SEs sind also bspw. der Klasse Pumpe (gültig für alle Pumpen), Flüssigkeitspumpe (gültig für Flüssigkeitspumpen) und Kreiselpumpe (gültig ausschließlich für Kreiselpumpen) zugeordnet. Diese Zuordnung wurde auch auf den ECLASS-Standard übertragen. ECLASS ist in den beiden Varianten *Basic* und *Advanced* verfügbar. Während mittels *Basic* flache Listen einer spezifischen Produktklasse zugeordnet werden, können in der *Advanced*-Variante *Aspekte* und *Blöcke* verwendet werden. Ein Aspekt beschreibt, im Gegensatz zu einem Block, keine produktspezifischen Eigenschaften und ist somit keiner Klassifikationsklasse direkt zugeordnet. In einem Aspekt können universelle SEs organisiert werden, welche beliebig vielen Klassifikationsklassen zugeordnet

werden können. Die SEs für Pumpen wurden in Aspekten zusammengefasst, und dann entsprechend den relevanten ECLASS-Produktklassen hinzugefügt. Der Produktklasse Umlaufbeschleunigerpumpe (0173-1#01-AAZ371#020) wurden bspw. die Aspekte *Pumpe Design* (0173-1#01-AGZ178#002), *Flüssigkeitspumpe Design* (0173-1#01-AGZ179#002) und *Kreiselpumpe Design* (0173-1#01-AGZ180#002) zugeordnet. Während im ersten Aspekt SEs wie die maximale Umgebungstemperatur enthalten sind, enthält der Aspekt zum Design von Kreiselpumpen spezifische SEs wie die spezifische Saugzahl. Der allgemeine Aspekt *Pumpe Design* wird auch anderen ECLASS-Produktklassen zugeordnet, bspw. der Kreiselpumpe (0173-1#01-AAZ379#020), der Aspekt Kreiselpumpen Design hingegen nur der Kreiselpumpen innerhalb des ECLASS-Standards.

### 5.3 Fazit homogene Semantik

In diesem Kapitel wurde ein Vorgehensmodell zur Ableitung von Teilmodellen entworfen, und auf Basis dessen Teilmodelle für Pumpen abgeleitet. Das Vorgehensmodell basiert auf einer prozess- und lebenszyklusorientierten Modellierung von Assetfunktionen und ermöglicht die einheitliche Ableitung von Teilmodellen. Das Modell ist auf alle Assets anwendbar und kann zur Definition von Teilmodellen verwendet werden. Konkret wurde das Modell auf die Entwicklung von Teilmodellen für Pumpen angewandt. Die Teilmodelle und deren SEs wurden im Anschluss in den ECLASS-Standard integriert, sodass sie eine eindeutige Identifikation erhalten. Mit dem Vorgehensmodell wurde die in Kapitel 4.5 aufgeführte Herausforderung „Konzept zur abgestimmten Modellierung von Asset Informationen“ adressiert. Teilmodelle können nun einheitlich abgeleitet werden, wodurch die in den Teilmodellen spezifizierten Informationen aufeinander abgestimmt sind. Die Anforderungen 8 und 9 aus Kapitel 3 werden durch die Ableitung standardisierter Teilmodelle erfüllt. Durch die Integration in den ECLASS-Standard sind die einzelnen SEs der Teilmodelle eindeutig identifizierbar. Durch den Einsatz von Teilmodellen existieren semantische Modelle (Anforderung 9), und die Informationen sind für andere interpretierbar (Anforderung 8).

## 6 Semantische Interoperabilität - Heterogene Semantik

In diesem Kapitel wird das entwickelte Konzept zur Verarbeitung heterogener Semantik mittels NLP beschrieben. Zunächst wird in Kapitel 6.1 die grundsätzliche Idee erläutert. Um heterogene Semantik zu verarbeiten werden verschiedene Ansätze entwickelt (Kapitel 6.1.2-6.1.5). Diese werden in den jeweiligen Kapiteln kurz vorgestellt und im Anschluss auf dem erstellten Evaluationsdatensatz getestet und gegenübergestellt (Kapitel 6.1.6). Die entwickelte Methode wird als ein I4.0-Interface ausgeprägt, sodass es in I4.0-Umgebungen implementiert werden kann (Kapitel 6.2.1 und 6.2.2). Innerhalb einer prototypischen Implementierung wird das I4.0-Interface umgesetzt (Kapitel 6.2.3).

### 6.1 Natural Language Processing als Basis für semantische Interoperabilität von Verwaltungsschalen

In Kapitel 4.5 wurde der aktuelle Stand der Technik im Bereich I4.0 hinsichtlich der Erfüllung der Interoperabilitätsebenen und der spezifizierten Anforderungen dieser Arbeit beschrieben. Die Herausforderung ein einheitliches Modell zur Ableitung von Teilmodellen zu entwerfen, um die Mehrfachdefinition dieser zu verhindern, wurde mit dem vorherigen Kapitel 5 adressiert. Gleichzeitig wurde im Diskussionskapitel beschrieben, dass eine Eini-gung auf globale, einheitliche Teilmodelle als unwahrscheinlich angesehen wird und aktuell nicht umsetzbar ist. Daher müssen heterogene Beschreibungen von nicht abgestimmten Teilmodellen aufeinander abgebildet werden. Teil dieser Arbeit ist die Entwicklung einer Methode, welche eine solche Abbildung ermöglicht, um automatisierte Interaktionen von AAS realisieren zu können. Übergeordnetes Ziel ist es dabei, dass die Verarbeitung hete-rogener Semantik ohne manuellen Aufwand durchgeführt werden kann. Eine Abbildung in Form von händisch angelegten Äquivalenztabelle oder Transformationssprachen ist hierfür demnach ungeeignet. Bei beiden Möglichkeiten müssen die verwendeten Vokabulare vorab bekannt sein, sodass eine initiale, händische Abbildung durchgeführt werden kann.

Der in dieser Arbeit entwickelte Ansatz basiert auf der Idee, dass AAS befähigt werden, unbekannte Semantik zu verarbeiten, ohne dass diese vorab händisch auf das eigene Vokabular abgebildet wird. Die Basis sind Methoden aus dem Bereich des NLP und der entwickelte Ansatz wird im Folgenden als Semantic-Matching (SM) bezeichnet [182]. Die grundsätzliche Idee ist in der folgenden Abbildung 6.1 dargestellt.

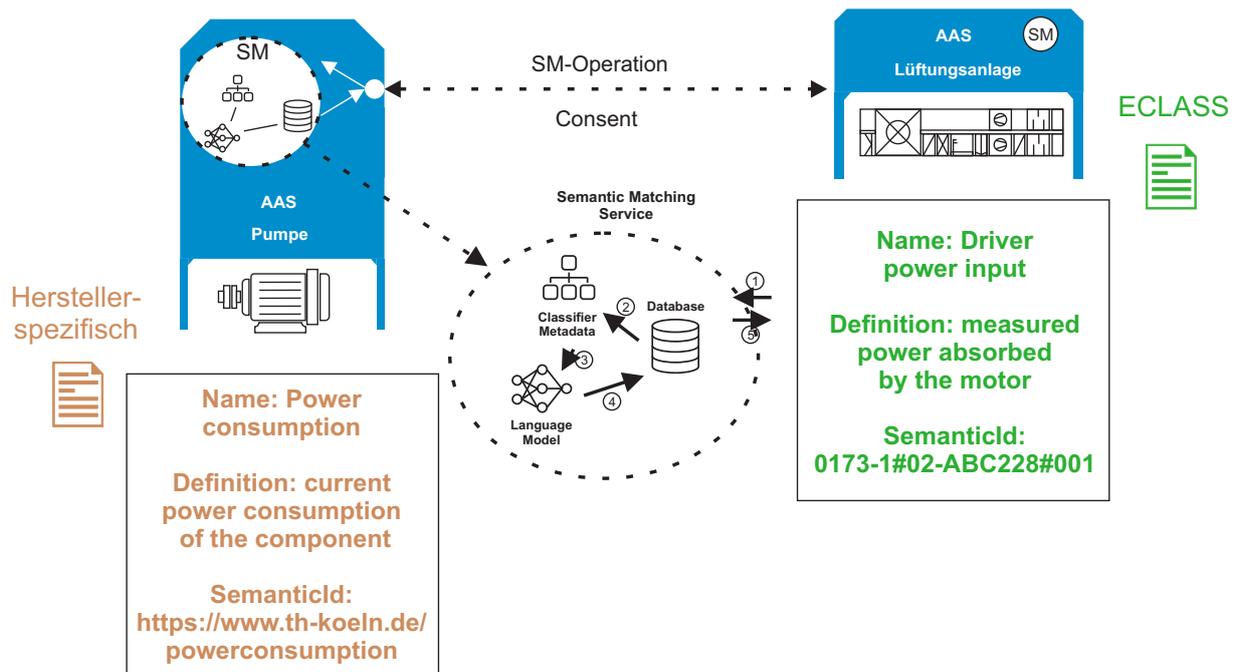


Abbildung 6.1: Semantische Interoperabilität von AAS mittels SM

Die zwei AAS einer Pumpe und einer RLT-Anlage interagieren miteinander. Hierbei basieren sie auf heterogener Semantik, also kennen das Vokabular der jeweils anderen AAS nicht. Die AAS der RLT-Anlage verwendet den ECLASS-Standard als Vokabular, während die Pumpe ein herstellereinspezifisches Vokabular verwendet. Im gezeigten Beispiel fragt die RLT-Anlage bei der Pumpe nach der *Leistungsaufnahme des Antriebs*. Damit semantische Interoperabilität zwischen den beiden hergestellt werden kann, wird ein Matching der beiden Vokabulare benötigt. Die Abbildung der unbekannt Semantik auf die eigene wird durch den *SM-Service* realisiert. Hierfür werden in dieser Arbeit verschiedene NLP-basierte Ansätze entwickelt. Zum Finetuning und zur Evaluierung dieser Ansätze muss zunächst ein Datensatz erstellt werden, welcher im folgenden Kapitel beschrieben wird.

### 6.1.1 Datensatz mit heterogenen Beschreibungen

Um zu überprüfen wie gut sich NLP-Methoden für die Verarbeitung heterogener Semantik einsetzen lassen, muss dies auf einem Datensatz evaluiert werden. Hierzu wurde im Rahmen dieser Arbeit ein Datensatz basierend auf dem ECLASS-Standard [124] erstellt. Der Datensatz besteht aus einem *Corpus*- und einem *Query-Teil*. Der Corpus enthält eine Sammlung von 672 exemplarischen ECLASS-Merkmalen von Kreiselpumpen- und Verdrängerpumpen, sowie RLT-Anlagen. Er repräsentiert SEs eines standardisierten Zielvokabulars. Der Query-Teil des Datensatzes besteht aus Paraphrasen der Corpus-Elemente, d.h. die Queries besitzen die gleiche semantische Bedeutung aber unterschiedliche Namen und Definitionen und repräsentieren somit ein heterogenes Vokabular anderer Komponenten. Es ist die Aufgabe

der LMs, den Corpus-Elementen automatisiert die korrekten Paraphrasen zuzuordnen. Neben den Attributen *Name* und *Definition* enthält der Datensatz Informationen zu den Metadaten *Datentyp* und *Einheit* der SEs. Für jedes SE wurden händisch bis zu elf Paraphrasen (zwischen zwei und elf), bestehend aus Name und Definition, erstellt (Auszug in Tabelle 6.1).

Tabelle 6.1: Beispiele für Paraphrasen der Merkmale der Teilmodelle

Merkmal des Corpus-Teils		Paraphrase des Query-Teils	
Name	Definition	Name	Definition
Name of product	Product designation under which the device is marketed	Product title	Title under which the product is known
Manufacturer	Name of the manufacturer of a device	Producer	Company or organization that transforms raw material into a product
Maximum allowable ambient temperature	Highest allowable ambient temperature for which the equipment (or any part to which the term refers) is suitable	Highest permissible ambient temperature	Maximum temperature that may be present in the immediate exterior environment of the operation resource. This is the maximum environmental temperature not ever to be exceeded in which the operating resource can be operated without permanent changes in its characteristics
Optimum rate of flow	Rate of flow at the point of best efficiency	Best volume flow	Optimum volume flow of a component

Insgesamt wurden 1.711 Paraphrasen der 672 Merkmale erstellt. Sowohl der Corpus- [183] als auch der Query-Datensatz [184] sind veröffentlicht und frei zugänglich.

### 6.1.2 Sentence Embeddings zur Verarbeitung heterogener Semantik

In Kapitel 2.3.5 wurde der Bereich PI als Teilgebiet von NLP beschrieben. Ein Beispiel für PI ist der QQP-Datensatz, bei dem vorhergesagt werden soll, ob Fragen eine Paraphrase einer existierenden Frage eines Korpus sind [67]. Verschiedene Modell-Architekturen haben gute Ergebnisse für diesen Datensatz erzielt, unter anderem das Modell *Sentence-BERT* [185]. Bei diesem Modell werden die kontextualisierten WEs des *BERT*-Modells [54] zu einem Sentence-Embedding (SEm) mit einer fixen Dimension zusammengefasst. Hierzu können unterschiedliche *Pooling-Methoden* verwendet werden. Mittels der Kosinus-Ähnlichkeit kann im Anschluss überprüft werden ob sich die SEms zweier Sätze ähneln. Dies ermöglicht

den Vergleich, ob eine Frage im QQP-Datensatz ein Duplikat im Korpus aufweist oder nicht [185]. Der Ansatz der SEms wird als erste Möglichkeit zur Verarbeitung heterogener Semantik verwendet. Der grundsätzliche Trainingsablauf, zur Erstellung von SEms innerhalb dieser Arbeit, ist in der folgenden Abbildung 6.2 dargestellt.

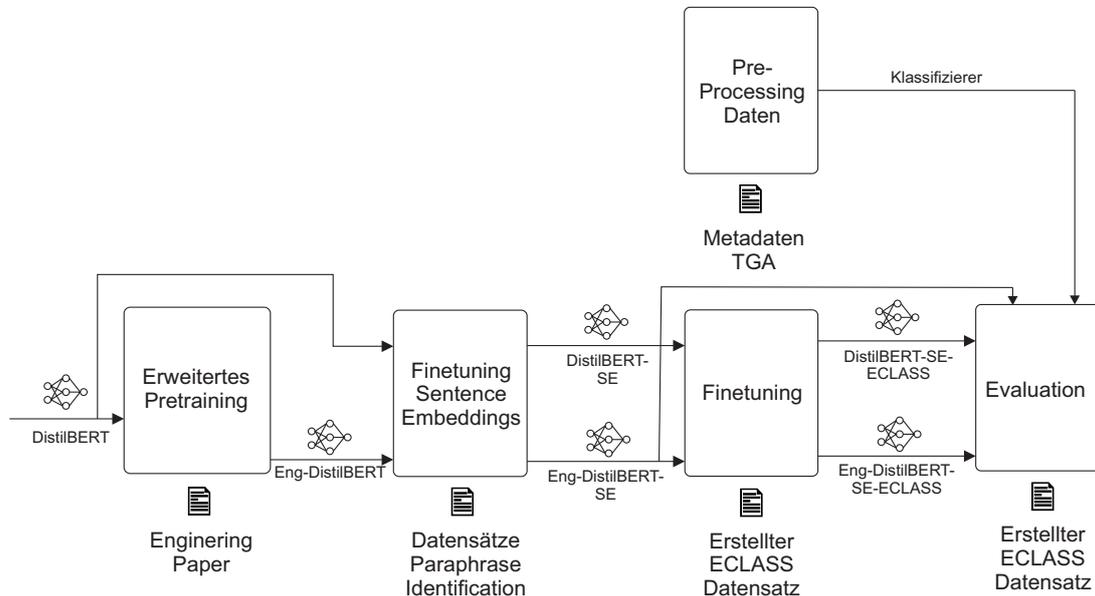


Abbildung 6.2: Ablauf des Trainings von SEms

Als Basis wird das PLM *DistilBERT* [56] verwendet. Dieses wird in einem ersten Schritt auf domänenspezifischen Texten weiter trainiert. Im Anschluss werden das Basismodell, *DistilBERT*, und das erweiterte Modell, *Eng-DistilBERT*, auf allgemeinen Datensätzen aus dem Bereich PI verfeinert. Der vorletzte Schritt beinhaltet das Finetuning auf einem Teil des erstellten ECLASS-Datensatzes. Im letzten Schritt werden dann die verschiedenen Modelle auf dem bisher nicht verwendeten Ausschnitt des ECLASS-Datensatzes evaluiert, um zu analysieren ob sich die Erweiterungen, wie das erweiterte Pretraining und Finetuning, positiv auf die Qualität der Vorhersagen des Modells auswirken. Parallel zu dem Training der Modelle wurden Metadaten analysiert, welche zur Unterstützung der Klassifizierung verwendet werden können. Die einzelnen Schritte sind in den folgenden Kapiteln näher beschrieben.

### Erweitertes Pretraining des Language Models

Das als Basismodell verwendete *DistilBERT* ist eine kleine Version des *BERT*-Modells (66 Millionen Parameter vs. 110 Millionen Parameter) und benötigt weniger Rechenleistung zum Training der Modelle. Das *DistilBERT-Modell* wurde auf englischer Literatur trainiert. Hierzu wurden der Books-Corpus [186] und das englische Wikipedia verwendet. Die Anwendungsfälle und *Benchmarks*, die üblicherweise genutzt werden, um LMs zu

bewerten, sind allgemeiner Natur. Aus diesem Grund werden die Modelle auf einer möglichst umfangreichen Literatur trainiert, um eine Abdeckung allgemeiner Anwendungsfälle zu gewährleisten. Untersuchungen zeigen jedoch, dass es für spezifische Anwendungsfälle lohnenswert ist, die vortrainierten PLMs auf weiterer domänenspezifischer Literatur zu trainieren [187]. Die Ergebnisse in [187] zeigen, dass eine zweite Phase des Pretrainings auf Texten der Domäne des Anwendungsfalls zu besseren Ergebnissen führt (*Domain-adaptive-Pretraining*). Bspw. wurde das vortrainierte PLM *RoBERTa* [55] auf Veröffentlichungen aus dem Bereich *Computer-Science* weiter trainiert, was zu einer Erhöhung des F1-Scores um 3,5 Prozentpunkte auf der SCIERC [188] Benchmark führte. Für ein erweitertes Pretraining innerhalb dieser Arbeit wurden Quellen gesucht, in denen technisches Verständnis in textueller Form enthalten ist. Als Quelle wurde der *S2ORC-Datensatz* [189] verwendet. Der Datensatz umfasst 81,1 Millionen englischsprachige, akademische Veröffentlichungen aus mehreren Fachbereichen wie Medizin, Chemie, Mathematik oder Physik. Die 228.000 Veröffentlichungen zum Thema Ingenieurwissenschaften (Engineering) wurden aus diesem Korpus verwendet. Das Training des Modells wurde mit der Transformer Bibliothek von *Hugging Face* umgesetzt [190]. Das dort zur Verfügung gestellte *DistilBERT*-Modell wurde auf den Veröffentlichungen, mit den Parametern aus Tabelle 6.2, weiter trainiert.

Tabelle 6.2: Parameter des erweiterten Pretraining

Hyperparameter - Name	Hyperparameter - Wert
Anzahl Tokens	2.195.229.696
Anzahl Tokens pro Sequence	512
Batch-Size	24
Gradient-Accumulation	64
Effektive Batch-Size	1.536
Anzahl Epochen	6

Insgesamt bestehen die 228.000 Veröffentlichungen aus 2,2 Milliarden Tokens. Für die *Tokenisierung* wurde das *Word-Piece-Modell* verwendet [191]. Die Tokens wurden in das Modell in Sequenzen eingelesen. Untersuchungen in [55] zeigen, dass die besten Ergebnisse erzielt werden, wenn nicht nur ein einzelner Satz eingelesen wird, sondern satzübergreifende Sequenzen. Daher wurden Sequenzen von 512 Tokens eingelesen. Ebenfalls in [55] wird beschrieben, dass bessere Ergebnisse erzielt werden, wenn eine größere Batch-Size verwendet wird. Durch die Limitierung der zur Verfügung stehenden Hardware (zwei *Graphics Processing Units (GPU)s*), war die Batch-Size zunächst auf 24 begrenzt. Durch den Einsatz des Gradient-Accumulation-Verfahrens, mit einer Größe von 64, konnte die effektive Batch-Size jedoch auf 1.536 erhöht werden. Das Modell wurde für sechs Epochen trainiert. Dieses Modell, bezeichnet als *Eng-DistilBERT*, wurde als Basismodell verwendet, um darauf aufbauend SEMs zu erstellen.

## Finetuning Sentence Embeddings

Für das Training der SEms wurde die Bibliothek *Sentence Transformer* [192] und die Datensätze *MultiNLI* [193], *STS-Benchmark* [194] und *QQP* [67] verwendet. Die drei Datensätze werden häufig eingesetzt, um im Bereich PI neue Modelle zu trainieren. Da für den Anwendungsfall der semantischen Interoperabilität von AAS bisher keine Datensätze für ein Finetuning zur Verfügung standen, wurden diese allgemeinen Datensätze verwendet. Hierdurch erlernt das Modell die Aufgabe des PI und kann diese auf den konkreten Anwendungsfall adaptieren. Die Modelle *Eng-DistilBERT* und *DistilBERT* wurden jeweils mit den drei Datensätzen verfeinert, sodass die Modelle *DistilBERT-SE* und *Eng-DistilBERT-SE* erstellt wurden.

Aus dem erstellten ECLASS-Datensatz wurden 1.040 Paraphrasen ausgewählt, um ein weiteres Finetuning durchzuführen. Diese Modelle werden als *DistilBERT-SE-ECLASS* und *Eng-DistilBERT-SE-ECLASS* bezeichnet. Der Schritt dient der Evaluation, welche Verbesserungen in der Genauigkeit erzielt werden können, wenn die Modelle auf einen domänenspezifischen Datensatz verfeinert werden. Die auf dem ECLASS-Datensatz verfeinerten Modelle wurden bei der Evaluation den Modellen ohne ein spezifisches Finetuning gegenübergestellt, um die Verbesserung der Genauigkeit bewerten zu können. Abbildung 6.3 zeigt den Ablauf der Evaluation für den Ansatz des SM, basierend auf SEM und PI.

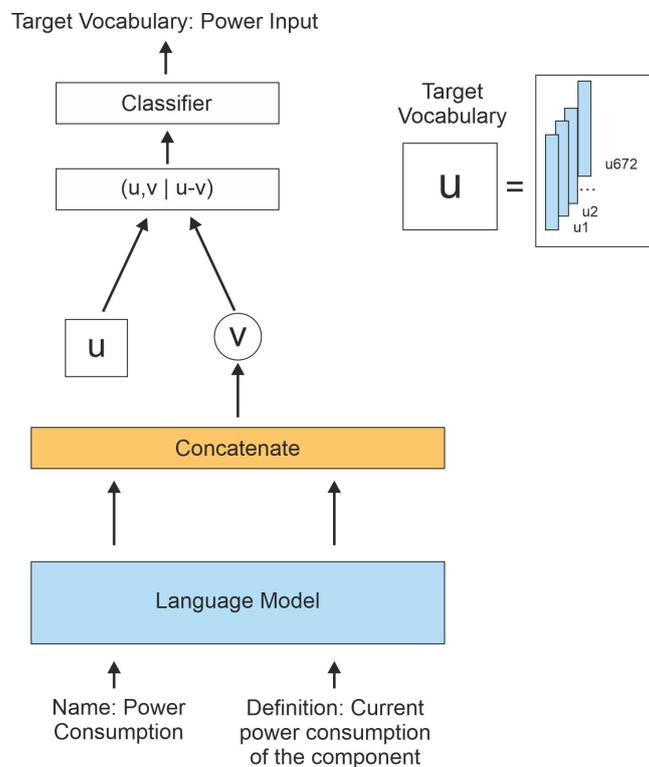


Abbildung 6.3: Ablauf Evaluation SEms

Der Name und die Definition eines unbekanntes SE (hier eine Query aus dem ECLASS-

Evaluationsdatensatz) werden an das Modell übergeben. Das zu evaluierende LM verarbeitet Namen und Definition zunächst getrennt voneinander und bildet zwei unabhängige SEms. Diese werden in einem nächsten Schritt miteinander zu dem Vektor  $v$  verkettet (*Concatenate*). Das Zielvokabular (*Target-Vocabulary*),  $u$ , repräsentiert den ECLASS-Datensatz. Für alle 672 Elemente des Datensatzes sind vorab SEms gebildet worden. Für jedes der Elemente wird im abschließenden Schritt die Kosinus-Ähnlichkeit mit dem Vektor  $v$  des unbekanntem SE berechnet. Das Element des Zielvokabulars mit der höchsten Ähnlichkeit wird als passende Paraphrase ausgewählt und als Match ausgegeben.

### Vergleich mit weiteren Sentence Embedding Modellen

Seit Beginn der Untersuchungen zu dieser Arbeit hat sich der Bereich des NLP im Allgemeinen und der Bereich der SEms im Besonderen stetig weiterentwickelt. Um eine Vergleichbarkeit mit verfügbaren State of the Art Modellen herzustellen, wurden zusätzlich Modelle evaluiert, die derzeit die besten Ergebnisse im Bereich der SEms und *Semantic-Search* erzielen [195]. Ein Screenshot der aktuell besten Modelle nach [195] ist in Anhang B enthalten. Als Vergleichsmodelle wurden die beiden Modelle *all-mpnet-base-v2* und *multiqa-mpnet-base-dot-v1* ausgewählt. Für diese Modelle wurde das PLM *MPNet* als Basis verwendet [196]. *MPnet* ist wie das *DistilBERT*-Modell ein Encoder-basiertes Modell. Innerhalb dieser Arbeit wurde das Basismodell *MPNet* zusätzlich auf den gleichen drei Datensätzen (NLI, STS, QQP) wie die *DistilBERT*-Modelle verfeinert und wird im folgenden als *MPNet-base-SE* bezeichnet.

Der auf SEM basierende Ansatz des SM wurde um den Einsatz von KGs erweitert, und ist im folgenden Abschnitt beschrieben.

#### 6.1.3 Erweiterung mit Entity Embeddings

Die Integration von KGs in die SM-Methode wird durch zwei unterschiedliche Ansätze realisiert. Die hybride Methode besteht aus Repräsentationen eines LM in Verbindung mit statischen EEs. Die kontextualisierte Methode umfasst die Repräsentationen eines kontextualisierten Knowledge-LM. Die Umsetzung dieser Methoden erfordert die Aufbereitung von Trainingsdaten und das Training verschiedener Modelle. Die Trainingsdaten der beiden Methoden bestehen zum einen aus ingenieurwissenschaftlichen Veröffentlichungen (vgl. Kapitel 6.1.2) und den KGs BRICK [197] und Project Haystack [198], die das Wissen der gebäudetechnischen Domäne widerspiegeln.

## Statische Embeddings

Die Erstellung der Trainingsdaten der statischen EEs erfolgte auf Basis der KGs. Die Trainingsdaten mussten hierfür als N-Triple-Serialisierung dargestellt werden: Jede Zeile der KGs enthält ein Subjekt (Entität), ein Prädikat (Relation) und ein Objekt (Entität) in Form von URLs. Die Präfixe innerhalb der URLs wurden entfernt, sodass die KG-Tripel nur noch aus Begriffen natürlicher Sprache bestanden. Diese Begriffe wurden separiert nach Entität und Relation extrahiert. Da die Trainingsdaten aus zwei verschiedenen KGs bestehen, die bedeutungsgleiche Entitäten durch unterschiedliche Begriffe definieren, erfolgte eine manuelle Gruppierung der extrahierten Entitäten und Relationen nach Synonymen. Diese Gruppen wurden manuell um weitere Synonyme ergänzt. Die einzelnen Synonymgruppen wurden mit IDs versehen, sodass die Elemente der Tripel durch diese IDs ersetzt werden konnten. Die Trainingsdaten bestehen aus einer Liste von ID-Tripeln und einer Liste der Synonymgruppen inklusive IDs. Insgesamt umfassen sie 3.630 verschiedene Entitäten, 50 Relationen und 20.860 Tripel.

Das Training der statischen EEs erfolgte mit Hilfe des Modells *Eng-CoLAKE* (beschrieben im nächsten Abschnitt). Anhand der ID-Tripel wurden die Zusammenhänge innerhalb des KG durch das Modell in EEs umgewandelt. Das Ergebnis des Trainings sind statische EEs, für jede in den Tripel vorkommende Entität. In der Anwendung werden die statischen EEs mit den gebildeten SEms eines beliebigen LM verknüpft. Abschließend erfolgt ein Ähnlichkeitsvergleich der verketteten Embeddings mittels der Kosinus-Ähnlichkeit (vgl. Abbildung 6.4).

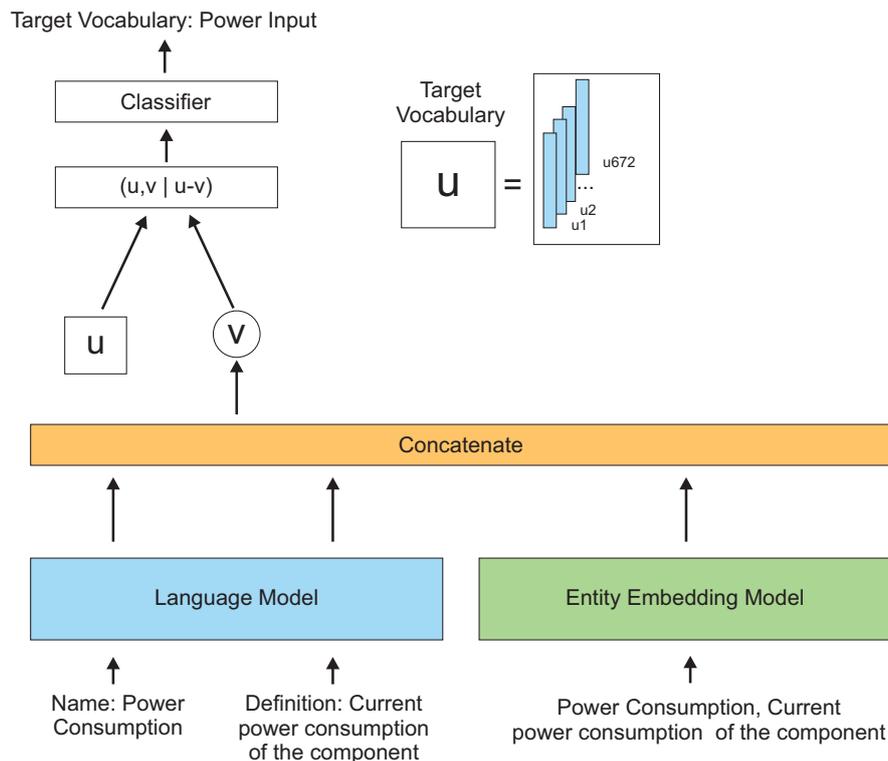


Abbildung 6.4: Ablauf Evaluation Kombination SEMs mit statischen EEs

Die statischen EEs des *Eng-CoLAKE* Modells wurden mit den trainierten LMs dieser Arbeit kombiniert (vgl. Tabelle 6.3).

Tabelle 6.3: Übersicht über kombinierte LMs mit EEs

Name des LM	Name kombiniertes LM mit EEs
DistilBERT-SE	DistilBERT-SE-Eng-CoLAKE
Eng-DistilBERT-SE	Eng-DistilBERT-SE-Eng-CoLAKE
DistilBERT-SE-ECLASS	DistilBERT-SE-ECLASS-Eng-CoLAKE
Eng-DistilBERT-SE-ECLASS	Eng-DistilBERT-SE-ECLASS-Eng-CoLAKE

### Kontextualisierte Embeddings

Für das Training der kontextualisierten Embeddings wurden einerseits die ingenieurwissenschaftlichen Texte und andererseits die ID-Tripel der KGs verwendet. Aus der Kombination dieser beiden Elemente wird ein Knowledge-Language-Graph gebildet, um die heterogenen Strukturen von Text und KG verbinden zu können. Dieser setzt sich aus einem Satz und Teilgraphen des KG zusammen. Die Verknüpfung von Satz und Teilgraph erfolgt über die im Satz vorkommenden Elemente. Dazu ist es erforderlich, dass die Entitäten des Satzes mittels regulärer Ausdrücke erkannt werden. Die Entitäten des Satzes werden

als Entitätsknoten markiert, an denen die Teilgraphen anknüpfen, sodass daraus ein Knowledge-Language-Graph entsteht.

Das Training der kontextualisierten Embeddings wurde in Anlehnung an [78] durchgeführt. Als Basismodell wurde das LM *RoBERTa* verwendet [55]. Die Modellarchitektur wurde so angepasst, dass zusätzlich die Aufgaben des *Masked-Entity-Modeling* und *Masked-Relation-Modeling* im Pretraining umgesetzt werden konnten. Die EEs wurden initialisiert, indem der Mittelwert der WEs der Entitäten ermittelt wurde. Zur Anpassung an die gebäudetechnische Domäne erfolgte ein erweitertes Pretraining des Modells und der EEs, mit Hilfe des kombinierten Knowledge-Language Graphen. Aus dem vortrainierten Modell *Eng-CoLAKE* können nun, wie im vorherigen Abschnitt erwähnt, statische EEs extrahiert werden. Das Finetuning und die Evaluation des Modells wurden gemäß der Beschreibung in Abschnitt 6.1.2 durchgeführt. Als Ergebnis des Finetuning sind die Modelle *Eng-CoLAKE-SE* und *Eng-CoLAKE-SE-ECLASS* entstanden. Zusätzlich zu reinen SEs und der Kombination von SEs mit EEs, wurde als weiterer Ansatz der Einsatz von *Large Language Models (LLM)s* untersucht. Dieser ist im folgenden Abschnitt beschrieben.

#### 6.1.4 Large Language Models zur Verarbeitung heterogener Semantik

Studien zeigen, dass die Leistung von Transformer-Modellen durch eine Erhöhung der Anzahl der Modellparameter verbessert werden kann [199]. In diesem Zusammenhang werden Modelle mit einer besonders großen Anzahl von Parametern als LLMs bezeichnet und sind in der Lage, verschiedene NLP-Aufgaben ohne zusätzliches Finetuning zu erfüllen, was als *Zero-Shot* bezeichnet wird [200].

In dieser Arbeit wird das bestehende LLM *T0\_3B* verwendet, welches zum Zeitpunkt der Evaluation auf verschiedenen allgemeinen Benchmarks die besten Ergebnisse lieferte [200]. Das Modell ist ein Encoder-Decoder-Transformer, der auf dem *T5*-Modell [201] basiert. Im Gegensatz zum SM mit Sentence-Transformern wird bei der Verwendung von LLMs ein unbekanntes SE zusammen mit einem SE des Zielvokabulars in das Modell eingespeist. Dazu werden die Namen und Definitionen der SEs in eine Satzstruktur, den *Prompt* [202], integriert. Die Wahl des Prompts hat einen bedeutenden Einfluss auf die Leistung des Modells [200]. Der in dieser Arbeit verwendete Prompt lautet:

- *SE1 Name; SE1 Definition.*
- *SE2 Name; SE2 Definition.*
- *Are these paraphrases duplicates?*

Im Gegensatz zu einem Sentence-Transformer-Modell erzeugt das LLM keinen Vektor (SEm), der die Bedeutung repräsentiert, sondern einen Wert zwischen Null und Eins, der die Ähnlichkeit der SEs charakterisiert [203]. Dieser Wert kann dann mit den Ähnlichkeitswerten der anderen SEs des Zielvokabulars verglichen werden. Auf dieser Grundlage kann das SE des Zielvokabulars ausgewählt werden, welches die höchste Ähnlichkeit mit dem unbekanntem SE aufweist. Der Ablauf der Evaluation ist in folgender Abbildung 6.5 dargestellt.

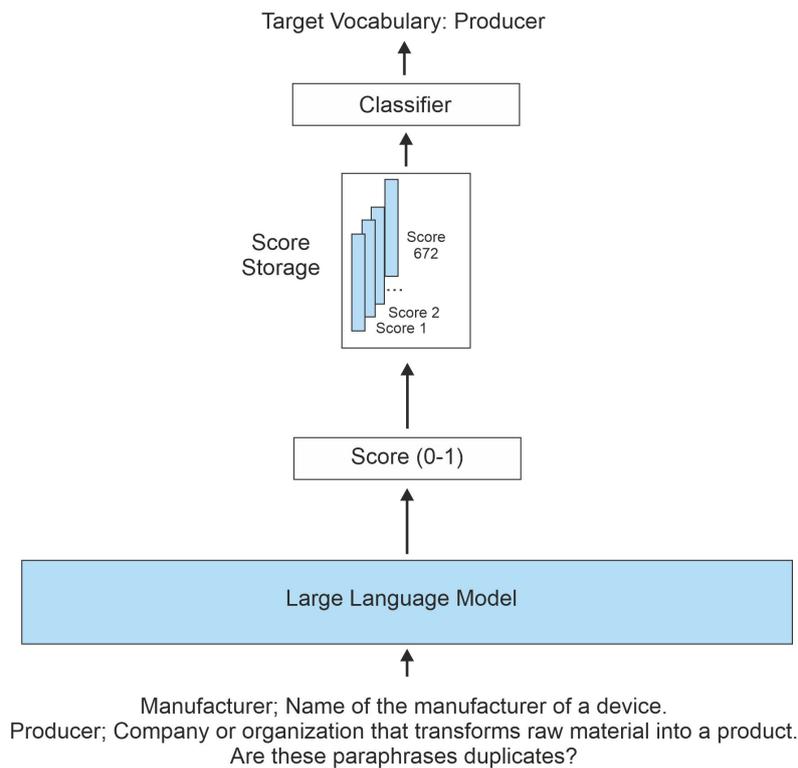


Abbildung 6.5: Ablauf Evaluation LLMs

Name und Definition des unbekanntem SE und jedes Elements des Zielvokabulars werden durch das LLM verarbeitet. Zusammen mit der Frage, ob sie Duplikate sind, werden die Angaben als Prompt in das Modell eingespeist und ein Wert zwischen 0 und 1 wird als Ergebnis ausgegeben. Alle Werte werden in einem Speicher abgelegt und im Anschluss wird das SE des Zielvokabulars mit dem höchsten Wert als entsprechende Paraphrase ausgewählt.

### 6.1.5 Metadaten als zusätzliche Klassifizierung

Im Mittelpunkt dieser Arbeit steht die semantische Interoperabilität zwischen AAS. Auch in Anwendungsfällen mit anderen digitalen Zwillingen kann die entwickelte SM-Methode eingesetzt werden. Das Metamodell der AAS spezifiziert jedoch zusätzliche Metadaten, die

zur Unterstützung der Klassifizierung verwendet werden können. Erste Analysen bei der Auswertung der SM-Ergebnisse haben ergeben, dass Fehler vor allem dadurch auftreten, dass verschiedene SEs oft sehr ähnlich bezeichnet und definiert sind [204]. So gibt es zum Beispiel mehrere, ähnlich definierte, Druckgrenzwerte für eine Pumpe. Zur präziseren Abgrenzung der SEs untereinander kann vorab eine metadatenbasierte Klassifizierung durchgeführt werden. Druckgrenzwerte, die sich z. B. lediglich durch den Zeitpunkt ihrer Definition unterscheiden (Definition bei der Herstellung vs. Planung), können durch eine entsprechende Metainformation voneinander abgegrenzt werden und kommen so nicht mehr als gegenseitige Paraphrase in Frage.

In einem ersten Schritt mussten die Metadatenkategorien gewählt werden, nach denen ein klassifizierender Algorithmus die SEs voneinander abgrenzt. Da auf der AAS und ihren internen Strukturen, in Form von Teilmodellen und SEs aufgebaut wird, wurden die in [11] definierten Attribute hinsichtlich ihrer Eignung als potenzielle Metadaten für das SM analysiert. Sie sind in folgender Tabelle 6.4 dargestellt.

Tabelle 6.4: Mögliche Metadaten der AAS

Metadatum - Name	Metadatum - Beschreibung
Submodel Element Type	Angabe, welche SE Art verwendet wird bspw. Property, File oder Operation
Datatype	Angabe, welcher Datentyp dem SE zugeordnet wurde
Unit	Angabe, welche Einheit dem SE zugeordnet wurde
Category	Angabe, ob ein SE der Kategorie Constant, Parameter oder Variable zugeordnet wurde

Die erste Metadatenkategorie *Submodel-Element-Type* unterscheidet die SEs in die unterschiedlichen Subklassen bspw. Property, File oder SMC. Aus der Klasse *Has-Data-Specification* werden der Datentyp (*Data-Type*) und die Einheit (*Unit*) eines SE als Metadatenkategorien entnommen. Das Metadatum *Category* erbt von der abstrakten Klasse *Referable* und unterscheidet die SEs in die Kategorien *Constant*, *Parameter* und *Variable*. In ersten durchgeführten Versuchen wurden alle vier Metadaten verwendet [204]. Um zu untersuchen, wie sich der Einsatz der Metadaten in der Praxis eignet, wurden existierende Instanzen von AAS untersucht (bspw. [205]). Bis auf das Metadatum *Submodel-Element-Type* sind alle Metadaten optional und werden daher nicht immer angegeben. Dies muss bei der Erstellung des Klassifizierers beachtet werden. Darüber hinaus gibt es keine verbindlichen Angaben, welcher *Category* ein SE zugeordnet werden soll, also ob *Parameter*, *Variable* oder *Constant*. Da dies letztendlich dem Anwender bei der Erstellung der AAS überlassen bleibt, können uneinheitlich beschriebene AAS entstehen. Daher wurde das Metadatum *Category* wieder aus dem Modell entfernt. Ebenso wurde das Metadatum

Submodel-Element-Type entfernt. Dieses ist zwar als einziges verpflichtend, die Untersuchungen in [204] haben jedoch gezeigt, dass es keinen Einfluss auf die Güte des Modells hat. Die unterschiedlichen Typen sind in ihrer Beschreibung und Namen so unterschiedlich, dass es keines Metadatum zur Unterscheidung bedarf. Letztendlich wurden nur die Metadaten Einheit und Datentyp verwendet. Diese wurden zunächst auf höhere Ebenen abgebildet, um sicherzustellen, dass bei der Definition des Datentyps oder der Einheit später keine Fehler bei der Klassifizierung auftreten, die auf gegebene Freiheitsgrade zurückzuführen sind. Die Vorlauftemperatur kann bspw. in der Einheit Kelvin, Grad Celsius, oder Fahrenheit angegeben werden. Würde jeder dieser Einheiten ein eigenes Metadatum zugeordnet, würde der Klassifizierer Daten unterschiedlich klassifizieren, wenn sie als Einheit Kelvin oder Grad Celsius aufweisen, obwohl sie bspw. jeweils eine Vorlauftemperatur repräsentieren. Um die Einheiten übergeordneten Kategorien zuzuordnen, wurde die DIN EN 61360 verwendet, da dort eine Einteilung der physikalischen Merkmale in die Hauptgebiete der Physik, zum Beispiel *Mechanik* oder *Raum und Zeit*, erfolgt [125]. Die Gruppierung wird teilweise übernommen und teilweise detaillierter aufgebrochen. So wird die Gruppe Raum und Zeit in *Länge*, *Volumen*, *Fläche*, *Geschwindigkeit*, und *Zeit* unterteilt. Tabelle 6.5 gibt einen Überblick über die ausgewählten Einheiten und die entsprechenden übergeordneten Kategorien.

Tabelle 6.5: Übergeordnete Kategorien Einheit

Übergeordnete Kategorie	Einheit
FREQUENCY	<i>Hertz, Hz, s<sup>-1</sup>, <math>\frac{1}{s}</math></i>
PRESSURE	<i>Pascal, Pa, Bar, <math>\frac{N}{m^2}</math></i>
SCALARS PERCENT	<i>Prozent, %</i>
POWER	<i>Watt, W, Kilowatt, kW, <math>\frac{J}{s}</math></i>
ELECTRICAL	<i>Volt, V, Ampere A, Kilowattstunden, kWh</i>
ACOUSTICS	<i>Dezibel, dB, dB(A)</i>
TEMPERATURE	<i>Kelvin, K, GradCelsius, °C</i>
MECHANICS	<i>Newton, N, Newtonmeter, Nm, <math>\frac{m}{m}</math>, <math>\frac{m^3}{m^2s}</math></i>
FLOW	<i>Massenstrom, <math>\frac{kg}{s}</math>, <math>\frac{kg}{h}</math>, Volumenstrom, <math>\frac{m^3}{s}</math>, <math>\frac{m^3}{h}</math>, <math>\frac{l}{s}</math>, <math>\frac{l}{h}</math></i>
LENGTH	<i>Mikrometer, <math>\mu m</math>, Millimeter, mm, Zentimeter, cm, Dezimeter, dm, Meter, m</i>
SPEED	<i>MeterproSekunde, <math>\frac{m}{s}</math>, <math>\frac{km}{h}</math></i>
TIME	<i>Sekunden, s, Minuten, m, Stunden, h, Tage, d</i>
VOLUME	<i>Kubikmeter, m<sup>3</sup>, Liter, l</i>
AREA	<i>Quadratmeter, m<sup>2</sup></i>
HEAT TRANSFER	<i>Wärmedurchgangskoeffizient, <math>w(m^2 \cdot k)</math>, Wärmedurchlasswiderstand, <math>m^2 \cdot \frac{k}{w}</math>, Wärmeleitfähigkeit, <math>\frac{w}{m \cdot k}</math></i>
NAN	<i>NaN</i>

Die übergeordneten Kategorien der Datentypen werden ebenfalls aus den vorhandenen

Begriffen der DIN EN 61360 abgeleitet. Folgende Tabelle 6.6 gibt einen Überblick über mögliche Datentypen der SEs von AAS und die übergeordneten Kategorien.

Tabelle 6.6: Übergeordnete Kategorien Datentyp

Übergeordnete Kategorie	Datentyp
BOOLEAN	BOOLEAN, BINARY
STRING	STRING, TRANSLATABLE_STRING, NON_TRANSLATABLE_STRING
DATE	DATE, DATE_TIME, TIME
URI	URI, IRDI, ICID_STRING, ISO_29002_IRDI
NUMBER	INT, INT_MEASURE, INT_CURRENCY, RATIONAL, RATIONAL_MEASURE, REAL, REAL_MEASURE, REAL_CURRENCY
ENUMERATION	ENUM_CODE, ENUM_INT, ENUM_REAL, ENUM_RATIONAL, ENUM_BOOLEAN, ENUM_STRING, ENUM_REFERENCE
AGGREGATION	BAG(b1,b2), LIST(b1,b2), UNIQUE_LIST(b1,b2), SET(b1,b2), CONSTRAINED_SET(b1,b2,cmn,cmx), SET[0,?], SET[1,?], ARRAY(b1,b2), OPTIONAL_ARRAY(b1, b2), UNIQUE_ARRAY(b1, b2), UNIQUE_OPTIONAL_ARRAY(b1, b2)
LARGE OBJECT TYPE	LOB(location, media_type, media_subtype, content_transfer_encoding)
NAN	NaN

Sowohl bei den Einheiten als auch den Datentypen wurde die Kategorie *NaN* eingeführt, die verwendet wird wenn ein Datentyp oder eine Einheit dem Klassifizierer nicht bekannt ist.

Die unterstützende Klassifizierung mittels Metadaten wird bei allen Ansätzen verwendet. Der Ablauf der Evaluation ist exemplarisch für den Ansatz der SEms in Kombination mit Metadaten in Abbildung 6.6 dargestellt.

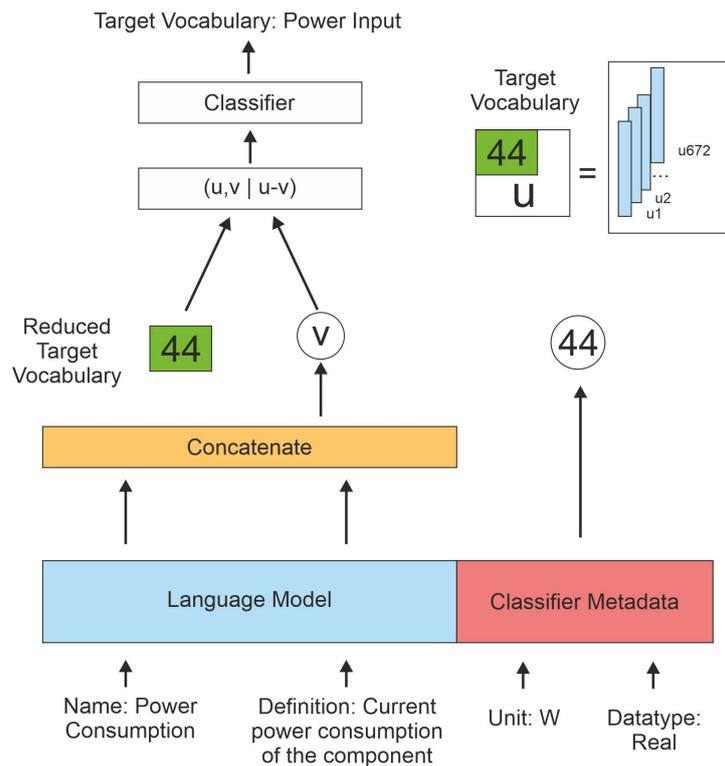


Abbildung 6.6: Ablauf Evaluation SEMs mit Metadaten

Name und Definition eines SE einer unbekanntes AAS werden verarbeitet und SEMs gebildet, welche im Anschluss verkettet werden. Parallel werden die Attribute des SE, Einheit und Datentyp, verarbeitet und auf ein Metadatum abgebildet. Dieses setzt sich aus der Kombination der gebildeten übergeordneten Kategorien der Attribute zusammen, bspw. 44. Das Zielvokabular wird von 672 (Größe des gesamten Corpus) auf die Anzahl der SEs reduziert, welche ebenfalls dem Metadatum 44 zugeordnet werden. Das SEM des unbekanntes SE wird im Anschluss nur mit den Elementen der Klasse 44 des Zielvokabulars verglichen, und jeweils die Kosinus-Ähnlichkeit berechnet. Durch dieses Vorgehen kann die Größe des Zielvokabulars reduziert werden.

Die verschiedenen entwickelten Ansätze wurden anhand des ECLASS-Datensatzes evaluiert. Die Ergebnisse sind im folgenden Abschnitt beschrieben.

### 6.1.6 Evaluation

Zunächst werden einzelne Aspekte der verschiedenen Basismodelle gegenübergestellt, vgl. Tabelle 6.7.

Tabelle 6.7: Übersicht der Basismodelle

Modell Name	Basismodell	Erstelldatum Basismodell	Anzahl Parameter (in Mio.)	Pretraining Datensatz (in GB)	Finetuning Datensatz (in Mio.)
DistilBERT-SE	DistilBERT	2019	66	16	0,84
Eng-DistilBERT-SE	DistilBERT	2019	66	26	0,84
MPNet-base-SE	MPNet-Base	2020	110	160	0,84
all-mpnet-base-v2	MPNet-Base	2020	110	160	1.170
multi-qa-mpnet-base-dot-v1	MPNet-Base	2020	110	160	215
sentence-t5-xl	T5-3B	2020	3.000	750	2.000
sentence-t5-xxl	T5-11B	2020	11.000	750	2.000
T0_3B	T5-3B	2020	3.000	750	-

In der Tabelle sind alle Basismodelle der Ansätze aufgelistet. Zunächst sind die Modelle aus dem Bereich PI dargestellt. Das erste Modell der Tabelle ist das vortrainierte Modell *DistilBERT*, welches als Basis für die ersten Untersuchungen innerhalb dieser Arbeit fungierte. Erste Untersuchungen wurden hierzu ab 2020 durchgeführt und die Methoden sowie der Datensatz schrittweise erweitert. Das *DistilBERT*-Modell ist von der Anzahl der Parameter (66 Millionen), der Größe des Pretraining-Datensatzes und der Größe des Finetuning-Datensatzes das kleinste Modell. Zwischen 2020 und 2023 wurde die Größe der LMs und der allgemeinen Pretraining-Datensätze kontinuierlich gesteigert, was sich in den Werten der Tabelle widerspiegelt. Vergleicht man das ursprüngliche Basismodell *DistilBERT* mit dem größten Basismodell *T5-11B*, fällt auf, dass sich die Anzahl der Parameter der Modelle um den Faktor 167 vergrößert hat. Die Größe des Pretraining-Datensatzes hat sich um den Faktor 47 vergrößert und die Größe des Finetuning-Datensatzes um den Faktor 2.381. Untersuchungen der Modelle auf allgemeinen NLP-Benchmarks zeigen, dass sowohl die Größe der Modelle als auch der Datensätze einen großen Einfluss auf die Leistung der Modelle hat. Je größer die Modelle und der Datensatz, desto bessere

Vorhersagegenauigkeiten können erzielt werden [199, 206]. Um zu untersuchen wie sich die Größe der Modelle und Datensätze auf die Ergebnisse dieser Arbeit auswirken, wurden nicht nur eigene Modelle evaluiert, sondern ebenfalls Modelle, welche auf allgemeinen Datensätzen aus dem PI-Bereich die besten Ergebnisse erzielen konnten. Als Vergleichswerte einer ähnlichen Modellgröße zu den eigens trainierten Modellen *DistilBERT-SE* und *Eng-DistilBERT-SE*, wurden die beiden Modelle *all-mpnet-base-v2* und *multi-qa-mpnet-base-dot-v1* ausgewählt. Obwohl die Größe des Basismodells *MPNet* [196] fast doppelt so groß ist wie die des *DistilBERT*-Modells, verfügt es im Vergleich zu den Modellen *T5-3B* und *T5-11B* über eine ähnliche Anzahl an Parametern wie das *DistilBERT*-Modell. Im Vergleich dieser Modelle wird insbesondere analysiert, wie sich die Größe des Pretraining- und des Finetuning-Datensatzes auf die Ergebnisse auswirkt. Zur besseren Vergleichbarkeit der Evaluationsergebnisse wurde das Basismodell *MPNet* auf den gleichen drei Datensätzen (NLI, STS, QQP) wie die *DistilBERT*-Modelle trainiert (*MPNet-base-SE*). Hierdurch können weitere Erkenntnisse über die Effizienz des erweiterten Pretraining und die Größe des Finetuning-Datensatzes gewonnen werden. Zusätzlich wurden Modelle ausgewählt, welche ebenfalls sehr gute Ergebnisse im Bereich PI erzielen und zu der Kategorie der LLM zählen (*sentence-t5-xl*, *sentence-t5-xxl*) [192]. Außerdem wurden SEM-Modelle um EEs erweitert. Hierbei wurden der hybride und kontextualisierte Fall untersucht. Die folgende Tabelle 6.8 gibt einen Überblick über die Ergebnisse der verschiedenen Modelle auf dem Evaluationsdatensatz.

Tabelle 6.8: Ergebnisse SM auf dem Evaluationsdatensatz

Modell Name	Genauigkeit ohne Meta- daten in %	F1-Score oh- ne Metadaten in %	Genauigkeit mit Meta- daten in %	F1-Score mit Metadaten in %
DistilBERT-SE	81,37	77,08	83,61	79,69
DistilBERT-SE-Eng- CoLAKE	82,27	78,07	84,20	80,51
Eng-DistilBERT-SE	84,21	80,00	86,59	83,06
Eng-DistilBERT-SE- Eng-CoLAKE	83,90	79,92	89,89	83,46
Eng-CoLAKE-SE	78,54	73,27	81,82	77,16
MPNet-base-SE	81,67	77,24	84,20	80,41
all-mpnet-base-v2	83,61	79,50	85,69	82,20
multi-qa-mpnet-base- dot-v1	84,95	80,99	87,03	83,78
sentence-t5-xl	87,63	84,36	89,12	86,31
sentence-t5-xxl	88,38	85,44	89,42	86,77
T0_3B	85,12	81,82	87,48	84,46
DistilBERT-SE- ECLASS	90,31	87,93	92,25	90,35
DistilBERT-SE- ECLASS-Eng- CoLAKE	90,76	88,50	92,40	90,55
Eng-DistilBERT-SE- ECLASS	90,61	88,08	92,55	90,57
Eng-DistilBERT- SE-ECLASS-Eng- CoLAKE	90,91	88,43	92,55	90,62
Eng-CoLAKE-SE- ECLASS	87,18	83,99	89,57	87,12
MPNet-base-SE- ECLASS	90,31	87,78	92,10	90,19
all-mpnet-base-v2- ECLASS	91,51	89,06	93,00	90,93
multi-qa-mpnet-base- dot-v1-ECLASS	90,16	87,53	92,70	90,83

In der Tabelle sind die Ergebnisse jedes Modells mit und ohne Metadaten aufgeführt. Als Metriken wurden die Genauigkeit und der F1-Score verwendet. In den eigenen Veröffentlichungen zu den verschiedenen Ansätzen wurde die Genauigkeit als Metrik verwendet [182, 204, 207–209]. Der Datensatz wurde zwischen den Veröffentlichungen kontinuierlich erweitert. Zu Beginn [182] war der Datensatz noch ausgewogen, da für alle Korpuselemente die gleiche Anzahl an Paraphrasen erstellt wurden. Der final erstellte und in dieser Arbeit verwendete Datensatz [183, 184] enthält unausgewogene Daten, da nun eine unterschiedliche Anzahl von Paraphrasen für die Korpuselemente vorliegt. Um die Unausgewogenheit des Datensatzes zu berücksichtigen, wurde zusätzlich der F1-Score bestimmt. Die Tabelle zeigt, dass dieser bei allen Modellen einen niedrigeren Wert aufweist als die Genauigkeit. Mit zunehmender Vorhersagegenauigkeit der Modelle verringert sich jedoch der Unterschied zwischen F1-Score und Genauigkeit. Das Basismodell *DistilBERT-SE* erreicht eine Genauigkeit auf dem Evaluationsdatensatz von 81,37 % ohne Metadaten, und 83,61 % mit Metadaten. Diese Werte werden als Referenz angenommen, um die Güte der anderen Modelle zu bewerten. Über alle Modelle hinweg zeigt sich, dass durch die zusätzliche Klassifikation mit Metadaten eine durchschnittliche Verbesserung von zwei Prozentpunkten erreicht wird. Dieses Ergebnis ist auf die Reduzierung des Zielvokabulars aufgrund der vorab durchgeführten Klassifizierung basierend auf den Metadaten zurückzuführen. Durch das erweiterte Pretraining auf den ingenieurwissenschaftlichen Veröffentlichungen wird eine Erhöhung der Genauigkeit von ca. drei Prozentpunkten erreicht. Dies zeigt, dass das erweiterte Pretraining auf domänenspezifischer Literatur auch in diesem Anwendungsfall zu besseren Ergebnissen führt. Die Modelle *all-mpnet-base-v2* und *multi-qa-mpnet-base-dot-v1* erreichen ähnliche Werte wie das *Eng-DistilBERT-SE* Modell. Das Modell *MPNet-base-SE*, das auf den gleichen drei Datensätzen wie die beiden *DistilBERT*-Modelle verfeinert wurde, erreicht hingegen ähnliche Ergebnisse wie das *DistilBERT-SE*-Modell. Dies zeigt, dass sowohl ein erweitertes Pretraining, Vergleich *DistilBERT-SE* und *Eng-DistilBERT-SE*, als auch eine Vergrößerung des Finetuning-Datensatzes, Vergleich *MPNet-base-SE* und *all-mpnet-base-v2* bzw. *multi-qa-mpnet-base-dot-v1*, einen positiven Einfluss auf die Leistung der Modelle hat. Sowohl durch ein erweitertes Pretraining, als auch eine Vergrößerung des Finetuning-Datensatzes, kann die Genauigkeit um ca. drei Prozentpunkte gesteigert werden. Die Erweiterung der SEM-Modelle mit EEs hat hingegen einen sehr geringen Einfluss auf die Vorhersagegenauigkeiten der Modelle. Die Erweiterung des *DistilBERT*-Modells mit EEs erreicht eine um einen Prozentpunkt bessere Genauigkeit (ohne Metadaten) bzw. 0,6 Prozentpunkte (mit Metadaten). Die Erweiterung des *Eng-DistilBERT*-Modells mit EEs führt ebenfalls nur zu einer geringfügigen Verbesserung. Hieraus kann geschlossen werden, dass durch die Verarbeitung von KGs in Form von EEs und die Kombination mit SEMs keine hilfreichen Informationen generiert werden können. Hier gilt es in Zukunft zu untersuchen, ob das formalisierte Wissen der KGs in anderer Form extrahiert bzw. mit Modellen kombiniert werden kann, sodass dieses Wissen im Bereich des SM verwendet werden

kann. Die beiden LLMs, die auf den Anwendungsfall PI verfeinert wurden (*sentence-t5-xl* und *sentence-t5-xxl*), erzielen die besten Ergebnisse im Zero-Shot Verfahren. Im Vergleich zum *Eng-DistilBERT-SE*-Modell kann die Genauigkeit um ca. drei Prozentpunkte gesteigert werden. Die Steigerung der Parameter der Modelle um den Faktor 167, führt demnach zu einer Verbesserung der Leistung der Modelle. Das LLM *T0\_3B*, welches nicht mittels Sentence-Transformer verfeinert wurde, sondern welches Prompting verwendet, erreicht Genauigkeiten, welche ca. einen Prozentpunkt besser sind als die des *Eng-DistilBERT-SE* Modells. Das Modell basiert auf demselben Basismodell wie *sentence-t5-xl* und erzielt Ergebnisse, die etwa zwei Prozentpunkte schlechter sind. Dies zeigt, dass ein Finetuning auf den Anwendungsfall PI sinnvoll ist. Die größte Steigerung der Vorhersagegenauigkeit wird durch das Finetuning auf dem Ausschnitt des ECLASS-Datensatzes erreicht. Fast alle Modelle, die auf diesem Datensatz verfeinert wurden, erreichen Genauigkeiten zwischen 90 und 92 Prozent (mit der Ausnahme *Eng-CoLAKE-SE-ECLASS*). Die Genauigkeit kann somit um bis zu neun Prozentpunkte, im Vergleich zum Modell ohne spezifisches Finetuning auf dem ECLASS-Datensatz, gesteigert werden.

Übergeordnet lässt sich feststellen, dass durch ein erweitertes Pretraining auf domänenspezifischen Texten die Vorhersagegenauigkeiten der Modelle gesteigert werden können. Dieser Aspekt wird in Zukunft jedoch immer geringer ausfallen, da die Größe der Pretraining-Datensätze der Modelle im Allgemeinen stetig zunimmt (vgl. Tabelle 6.7). Ausgenommen hiervon sind Texte wie Normeninhalte (vgl. Kapitel 7.3.3) oder firmeninterne Dokumente, welche nicht frei zugänglich sind. Durch das Training mit diesen spezifischen Daten können weiterhin Verbesserungen der Vorhersagegenauigkeit erzielt werden. Größere Modelle lernen besser zu generalisieren und können in unbekanntem Domänen eingesetzt werden. Auch die Größe des Finetuning-Datensatzes beeinflusst die Leistung der Modelle. Allerdings muss beachtet werden, dass der Datensatz erheblich vergrößert werden muss (von 0,84 Millionen. auf 1.170 Millionen.), um Verbesserungen zu erzielen. Ein Finetuning auf Daten des konkreten Anwendungsfalls führt zu einer deutlichen Leistungssteigerung der Modelle. Hier reicht auch ein vergleichsweise kleiner Datensatz (1.040), um diese Verbesserungen der Genauigkeit zu erreichen.

In diesem Kapitel wurden die entwickelten Ansätze zum NLP-basierten SM vorgestellt. Damit diese Modelle in einer I4.0-Umgebung eingesetzt werden können, wurden I4.0-Interfaces entworfen, welche diese Modelle integrieren und ein SM der Elemente semantisch heterogener AAS durchführen.

## 6.2 Semantic Matching als Industrie 4.0 Service

Die Umsetzung des SM-Dienstes erfolgt in drei Schritten: Zunächst wird ein I4.0-Interface entworfen, das drei Operationen enthält, die für das SM eingesetzt werden können (Kapitel 6.2.1). Darauf aufbauend wird ein Service beschrieben, welcher das Interface umsetzt (Kapitel 6.2.2). Zum Abschluss wird eine prototypische Implementierung des Service umgesetzt, um dessen Funktionalitäten zu demonstrieren (Kapitel 6.2.3).

Das im Folgenden beschriebene Interface wurde gemeinsam mit Mitarbeitern des „Lehrstuhls für Informations- und Automatisierungssysteme für die Prozess- und Werkstofftechnik“ der RWTH Aachen entworfen und wird dort weiterentwickelt [210].

### 6.2.1 Spezifikation des Industrie 4.0 Interfaces für Semantic Matching

In Kapitel 4.2.4 ist das I4.0-Service-Model beschrieben. Dieses enthält vier verschiedene Level, die für ein Interface spezifiziert werden müssen, damit es in einem I4.0-Service eingesetzt werden kann. In dieser Arbeit wird ein Konzept vorgestellt, welches das SM formal als I4.0-Interface spezifiziert, und zusätzlich in einem prototypischen Service umsetzt. Eine Übersicht über das entworfene Interface und die drei dazugehörigen Operationen ist in Tabelle 6.9 gegeben.

Tabelle 6.9: Übersicht über das SM-Interface

Operation name	Description
SemanticQueryAAS	Search Asset Administration Shell with specific semantics and return Asset Administration Shell from server
SemanticQuerySubmodelElement	Search within an Asset Administration Shell for submodel elements that correspond to a specific semantics and return corresponding elements
SemanticQueryMatchObject	Check whether two semantic identifiers match

Jede Operation wird definiert durch mögliche Input- und Output-Parameter, welche optional oder verpflichtend sein können. Die erste Operation wird als *Semantic-Query-AAS* bezeichnet (vgl. Tabelle 6.10).

Tabelle 6.10: Operation Semantic-Query-AAS

Operation Name	SemanticQueryAAS			
Explanation	Search Asset Administration Shell with specific semantics and return Asset Administration Shell from server			
Name	Description	Mand.	Type	Card.
<i>Input Parameter</i>				
asset-Description	Short description or name of the asset to be searched for	no	string	0..1
<i>Output Parameter</i>				
statusCode	Status code	yes	Status-Code	1
payload	Matched Asset Administration Shell	yes	Identifier	0..*

Als möglicher Input-Parameter kann eine textuelle Beschreibung, die *Asset-Description*, angegeben werden. Diese enthält in natürlicher Sprache eine Beschreibung des Asset-Typen. Die Operation wird verwendet, um innerhalb eines Netzwerks nach AAS zu suchen, welche Assets repräsentieren, die bestimmte Fähigkeiten haben. Ein Beispiel wäre ein Ingenieur, der für die Auslegung einer Anlage eine Pumpe benötigt. Mittels der Operation Semantic-Query-AAS kann das Netzwerk nach AAS durchsucht werden, welche Pumpen repräsentieren. Je nach Angabe des Inputs und Ausprägung des SM-Service kann dann ein Matching durchgeführt werden. Zum Beispiel kann auf Basis der Asset-Description ein Matching mittels NLP erfolgen. Als Output-Parameter werden der *Status-Code* und der *Payload* zurückgegeben, welche beide verpflichtend sind. Im Payload sind die Identifizierer der AAS enthalten, welche vom SM-Service als passende Antworten zurückgegeben wurden. Dabei ist es auch möglich, dass keine AAS zurückgegeben wird. Die zweite der drei Operationen wird als *Semantic-Query-Submodel-Element* bezeichnet und ist in Tabelle 6.11 definiert.

Tabelle 6.11: Operation Semantic-Query-Submodel-Element

Operation Name	SemanticQuerySubmodelElement			
Explanation	Search within an Asset Administration Shell for submodel elements that correspond to a specific semantics and return corresponding elements			
Name	Description	Mand.	Type	Card.
<i>Input Parameter</i>				
aasId	Identifier of the Asset Administration Shell in which a matching submodel element is to be found.	yes	Identifier	1
preferred-Name	Marking consisting of one or more words assigned to a component.	no	string	0..1
definition	Specification that unambiguously and uniquely describes the meaning of an information object in order to allow its differentiation from other information objects.	no	string	0..1
dataType	Identifies a specific attribute of the value of a property	no	string	0..1
unitOf-Measure	Regulation of the unit in which the value of a property should be indicated	no	string	0..1
<i>Output Parameter</i>				
statusCode	Status code	yes	Status-Code	1
payload	Matched submodel element	yes	idShort-Path	0..*

Diese Operation kann nach der ersten Operation durchgeführt werden. Wurden mittels der AAS-Suche eine oder mehrere AAS gefunden, welche dem gesuchten Typen entsprechen, können diese nach Fähigkeiten oder Informationen durchsucht werden. Wurden dem Ingenieur mehrere Pumpen zurückgegeben, kann er in diesen AAS nach Informationen, wie der maximalen Förderhöhe oder der maximalen Umgebungstemperatur, suchen. Mittels dieser Operation wird ein Matching auf Basis der semantischen Inhalte und Metadaten durchgeführt. Die Operation kann im Bereich der heterogenen Semantik angewandt werden, da nicht nach passenden Semantic-IDs gesucht wird, sondern ein Matching auf Basis der semantischen Informationen durchgeführt wird. Als mögliche Input-Parameter können hier der *Preferred-Name*, *Definition*, *Data-Type* und *Unit* angegeben werden. Ähnlich wie bei der ersten Operation sind die Output-Parameter der Status-Code und ein Payload. Dieser enthält nun den *ID-Short-Path* der SEs, die als Match zurückgegeben wurden. Die Namen und Beschreibungen der jeweiligen Input-Parameter sind [125] entnommen. Die dritte Operation *Semantic-Query-Match-Object* (Tabelle 6.12) wird für die Überprüfung der Übereinstimmung zweier Semantic-IDs verwendet.

Tabelle 6.12: Operation Semantic-Query-Match-Object

Operation Name	SemanticQueryMatchObject			
Explanation	Check whether two semantic identifiers match			
Name	Description	Mand.	Type	Card.
<i>Input Parameter</i>				
aasId	Identifier of the Asset Administration Shell in which a matching submodel element is to be found.	yes	Identifier	1
semanticId	Identifier of the semantic definition	yes	Identifier	1
<i>Output Parameter</i>				
statusCode	Status code	yes	Status-Code	1
payload	Matched submodel element	yes	idShort-Path	0..*

Im Gegensatz zur zweiten Operation findet hier ein Matching zweier Semantic-IDs statt. Diese Operation wird im homogenen Bereich eingesetzt, da nur ein exaktes Matching passender IDs durchgeführt werden kann. Weitere semantische Informationen wie der Name des SE oder die Definition werden nicht angegeben. Als Input-Parameter werden hier die *AAS-ID* der AAS, und die Semantic-ID des SE, angegeben. Als Output wird dann wiederum der ID-Short-Path zu dem passenden Element angegeben, falls eines gefunden werden konnte.

### 6.2.2 Umsetzung des Interfaces als Industrie 4.0 Service

Auf Basis der abstrakt definierten Operationen wurde eine prototypische Implementierung für einen SM-Service umgesetzt. Es wurde ein Service implementiert, welcher ein exaktes Matching von Semantic-IDs (Operation Tabelle 6.12) und ein Matching auf Basis von NLP (Operationen Tabellen 6.10 und 6.11) durchführen kann. Bevor ein Matching zwischen einer Anfrage und einer AAS, bzw. SEs dieser, durchgeführt werden kann, müssen die Elemente der zum Matching bereitgestellten AAS zunächst verarbeitet werden (vgl. Abbildung 6.7).



Nutzers entweder ein exaktes Matching auf Basis der Semantic-IDs oder ein NLP-basiertes Matching durchgeführt werden. Die Suche nach passenden AAS-Typen ist in Abbildung 6.8 dargestellt.

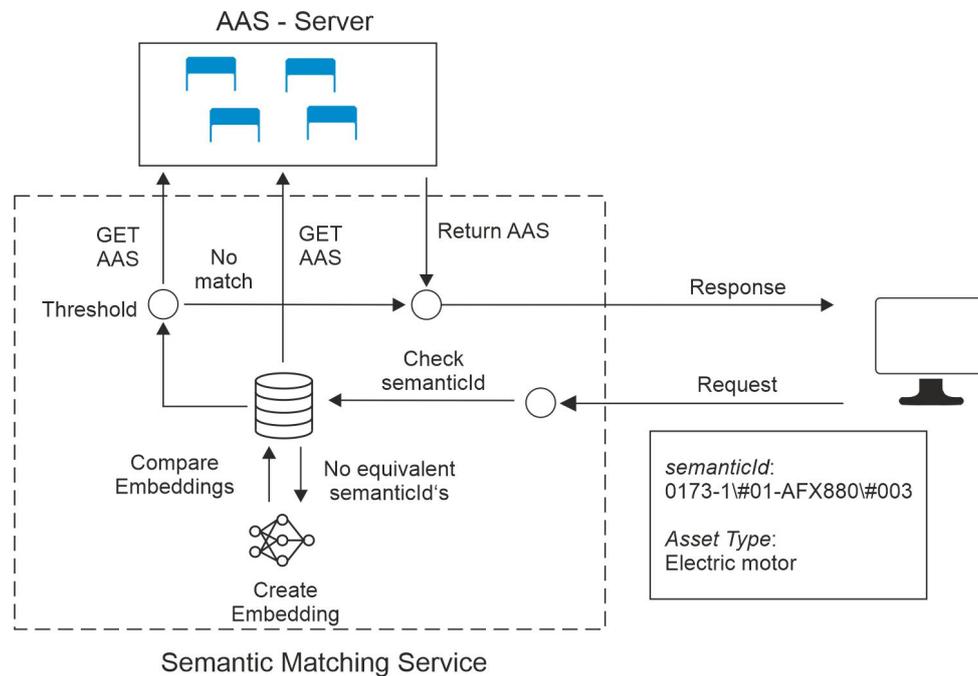


Abbildung 6.8: SM auf AAS-Ebene

Zur Suche nach einer AAS, welche ein bestimmtes Asset repräsentiert, können auf Basis der definierten Klasse *Query-Parameter* eine Semantic-ID und eine Asset-Description des Asset-Typen angegeben werden. Ist eine Semantic-ID angegeben, wird zunächst nach dieser in der Datenbank gesucht. Werden in der Datenbank, auf Basis der Klasse *Semantic-Query-Match-Object*, passende Semantic-IDs gefunden, werden diese AAS im Server abgefragt. Kann jedoch keine passende Semantic-ID gefunden werden, wird die NLP-Pipeline gestartet. Die textuelle Beschreibung des Asset-Typen durchläuft das LM und ein SEM wird gebildet. Dieses wird mit allen vorab erstellten Embeddings, welche die ECLASS-Klasse der einzelnen AAS beschreiben, verglichen. Das Embedding mit der höchsten Kosinus-Ähnlichkeit wird als potenzielles Match übergeben. Da bei dieser Methode jedoch immer ein Match zurückgegeben wird, wird mittels einer Grenze (Threshold) evaluiert, ob das Match ausreichend hoch ist, um als passend angesehen zu werden. Die Höhe dieser Grenze kann vorab definiert werden. Wird der Wert der Grenze überschritten, also das Match als ausreichend hoch angesehen, wird die passende AAS vom Server abgefragt. Ist das Match jedoch zu niedrig, wird keine passende AAS zurückgegeben. Die Suche nach passenden SEs innerhalb einer AAS ist in Abbildung 6.9 dargestellt.

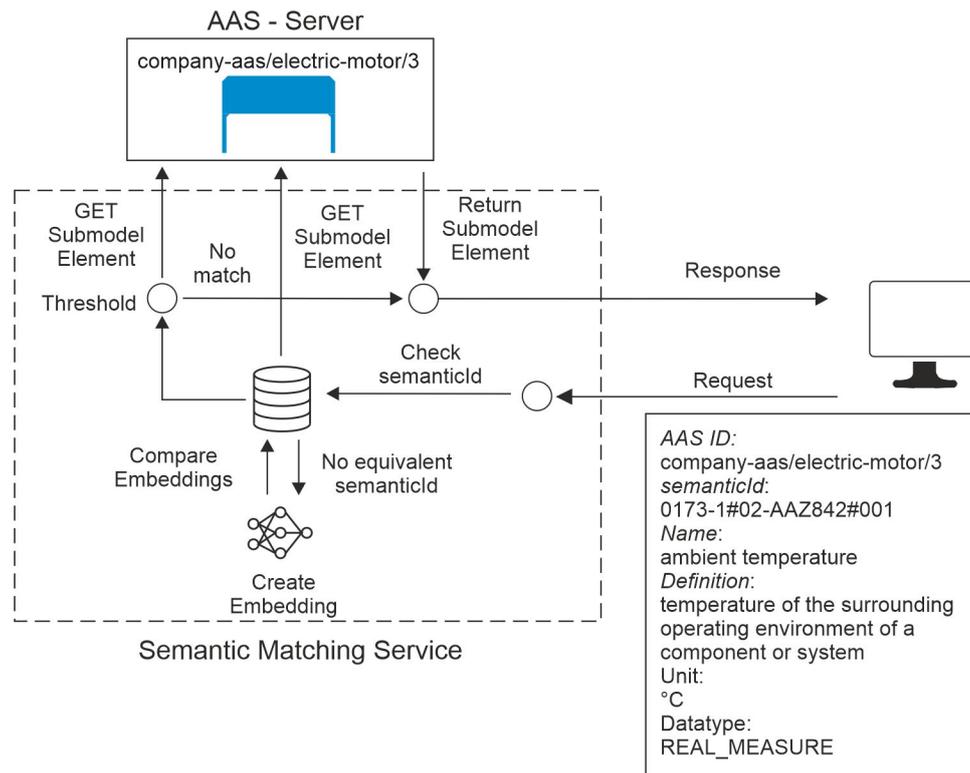


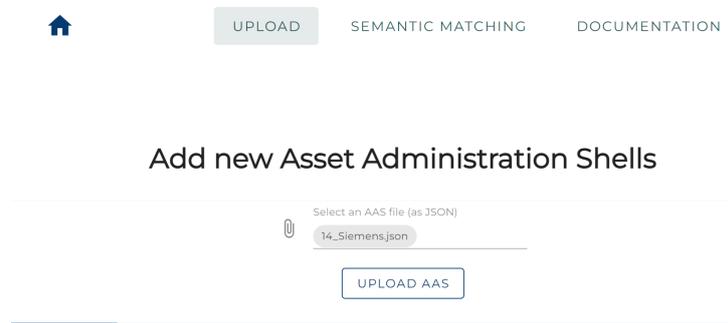
Abbildung 6.9: SM auf SE-Ebene

Basierend auf der abstrakten Klasse Query-Parameter wird eine Query definiert, welche die ID der zu durchsuchenden AAS enthält. Zusätzlich können eine Semantic-ID, der Name, die Definition, Einheit und Datentyp des SE angegeben werden. Zunächst wird versucht ein exaktes Matching mittels der Semantic-ID durchzuführen. Wird ein passendes Match identifiziert, wird im AAS-Server das SE abgefragt. Kann in der Datenbank, innerhalb der angefragten AAS, keine exakt passende Semantic-ID gefunden werden, wird die NLP-Pipeline gestartet. Name und Definition werden von dem LM verarbeitet und Datentyp und Einheit parallel klassifiziert. Auf Basis der Metadaten und des erstellten SEM wird in der Datenbank nach einem passenden SE gesucht. Falls ein exaktes Matching oder NLP-Match gefunden wurde, wird das passende SE als mögliche Antwort zurückgegeben. Wenn jedoch weder ein exaktes Matching noch ein NLP-Match gefunden wurde, wird kein SE als Antwort zurückgesendet.

### 6.2.3 Prototypische Implementierung des Semantic Matching Service

Der im vorherigen Kapitel beschriebene Service wurde als prototypische Anwendung umgesetzt. Hierzu wurde eine Anwendung erstellt, mittels derer AAS hochgeladen und für das SM bereitgestellt werden. In einem ersten Schritt werden die Informationen der hochgeladenen AAS verarbeitet. Hierbei wird der in Abbildung 6.7 dargestellte Schritt durchgeführt. Der AAS-Typ wird mittels NLP ermittelt, die Informationen der SEs der

Teilmodelle verarbeitet und in eine Vektordatenbank geschrieben, und die AAS wird auf einem BaSyx-Server bereitgestellt. Der Upload einer AAS und ein Ausschnitt dieser ist in folgender Abbildung 6.10 dargestellt.



(a) Upload einer neuen AAS



(b) Visualisierung einer AAS

Abbildung 6.10: Bereitstellen von AAS für das SM

Die Abbildung zeigt eine AAS vom Typ *Pressure sensor* (0173-1#01-AKE511#009). Zur Ermittlung des Typs wurden die Informationen der SEs der AAS analysiert und mittels der Werte die ECLASS-Produktklasse ermittelt (vgl. Kapitel 6.2.2). Die Teilmodelle sind ebenfalls visualisiert. Die auf dem BaSyx-Server bereitgestellten AAS können nun nach semantischen Informationen durchsucht werden. Abbildung 6.11 visualisiert die Funktionalitäten des SM-Services.

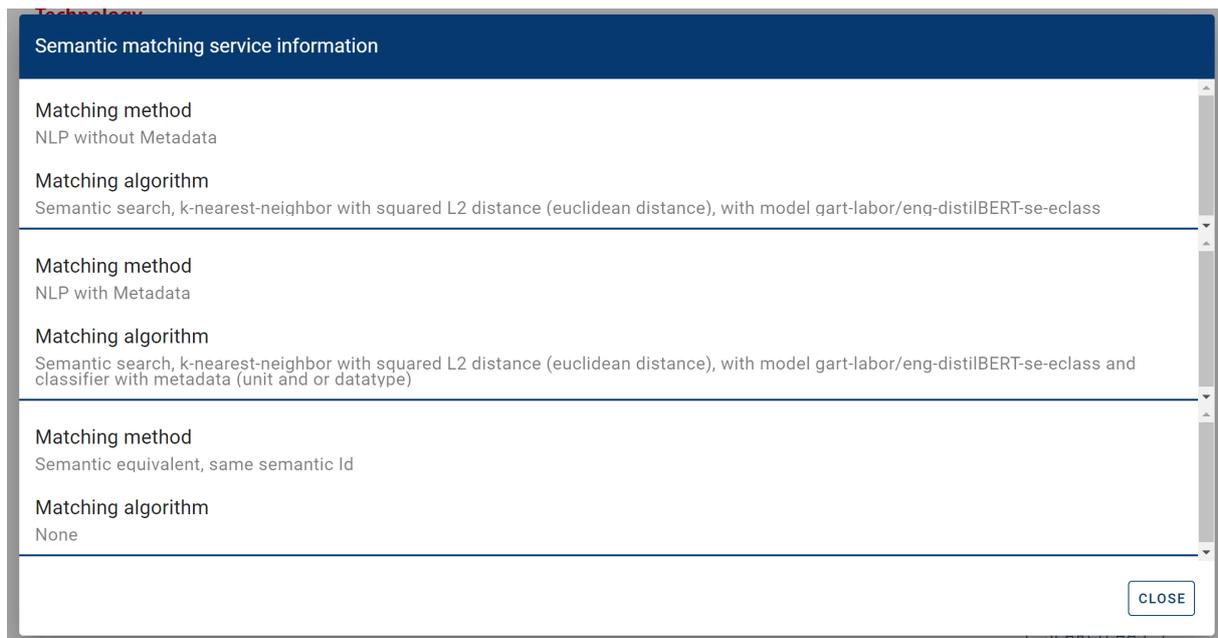


Abbildung 6.11: Informationen des SM-Service

Die Abbildung zeigt die drei möglichen Matching-Algorithmen, die verwendet werden können, um nach Informationen innerhalb einer AAS zu suchen. Innerhalb der Anwendung wurden diese drei so implementiert, dass sie nacheinander abgefragt werden. Wird vom Nutzer eine Semantic-ID eines SE angegeben, wird zunächst innerhalb der AAS nach einer passenden Semantic-ID gesucht. Kann hier eine passende ID gefunden werden, wird diese zurückgegeben, andernfalls werden die beiden anderen Algorithmen angewandt. Hierbei wird einerseits ein mit Metadaten unterstütztes Matching durchgeführt, andererseits ein Matching ohne Metadaten. Als NLP-Modell wird das hierfür trainierte Modell *Eng-DistilBERT-SE-ECLASS* verwendet. Die folgende Abbildung 6.12 zeigt die Eingabemaske des Nutzers und das Ergebnis des Matching.

Search AAS	
Preferred name	Manufacturer name
Definition	Name of the manufacturer of the product
Semantic Id	th-koeln.com/manufacturerName
Unit	
Datatype	
Number of results	3

(a) Eingabemaske des SM Service

Siemens\_SITRANS\_P320  
Pressure sensor

SEARCH AAS

Semantic matching result	
Matching method	NLP without Metadata
Matching algorithm	Semantic search, k-nearest-neighbor with squared L2 distance (euclidean distance), with model gart-labor/eng-distilBERT-se-e-class
Matching distance	0.06
Submodel ID	www.company.com/demo/sm/instance/F9CC779860FC1039
ID short path	Nameplate.ProducerName

ADD TO DOCUMENTATION

(b) Ergebnis des SM

Abbildung 6.12: SM innerhalb einer AAS

Die Maske zeigt die Suche nach dem Property des Herstellernamens, wobei der Name und die Definition auf keinem standardisierten Vokabular basieren. Auf der unteren Abbildung wird das passende Match visualisiert. Da keine passende Semantic-ID gefunden werden konnte, wurde ein NLP-basiertes Matching durchgeführt. Als Ergebnis werden das Teilmodell und der ID-Short-Path zu dem passenden SE innerhalb der AAS ausgegeben. Dieser zeigt, dass als Ergebnis der *Producer Name* mit einer Ähnlichkeit von 99,4%

zurückgegeben wurde.

Die in der Anwendung bisher händisch durchgeführten Suchen wurden innerhalb einer prototypisch implementierten Dokumentationsanwendung umgesetzt, um die Möglichkeit der automatisierten Abfrage und Integration der SEs zu demonstrieren. Hierzu wurden drei Definitionen und Namen von SEs ausgewählt, welche für ein mögliches Dokumentationsmanagement relevant sind (vgl. Abbildung 6.13).

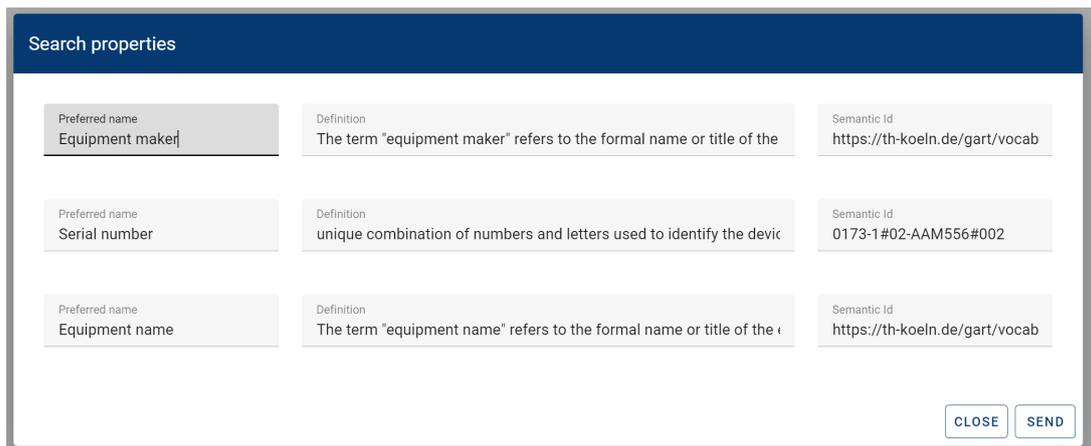


Abbildung 6.13: Visualisierung von drei möglichen Informationen des Dokumentationsmanagements

Hierbei wurden als Eingabe für die Suche zwei Namen und Definitionen gewählt, welche auf keinem Standard basieren, und eine Beschreibung aus dem ECLASS-Standard. Eine Übersicht der Ergebnisse ist in Abbildung 6.14 dargestellt.

## Documentation

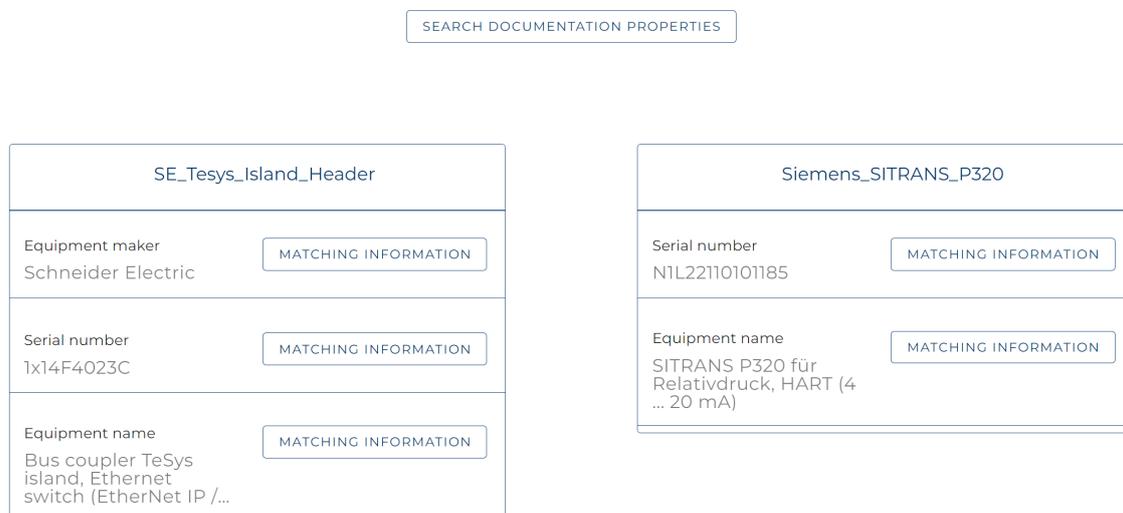


Abbildung 6.14: Übersicht Ergebnisse der gesuchten SEs

Für jede AAS des AAS-Servers wurde ein Eintrag generiert, welcher die gematchten Informationen enthält. Dargestellt sind die Namen der abgefragten SEs in Kombination mit den Werten der AAS. Die detaillierten Ergebnisse des Matching sind exemplarisch in den Abbildungen 6.15 und 6.16 dargestellt.

Matching information		
Own submodel element	Matched submodel element	
Name: Equipment maker	Name: Manufacturer Name	
Description: The term "equipment maker" refers to the formal name or title of the company or individual who creates or manufactures a product.	Description: legally valid designation of the natural or judicial person which is directly responsible for the design, production, packaging and labeling of a product in respect to its being brought into circulation;	
Semantic Id: https://th-koeln.de/gart/vocabulary/equipmentMaker/1/0	Semantic Id: 0173-1#02-AA0677#00	
Matching method NLP without Metadata	Matching algorithm Semantic search, k-nearest-neighbor with squared L2 distance (euclide...	Matching score 79 %
<input type="button" value="CLOSE"/>		

Abbildung 6.15: Ergebnis eines NLP-basierten Matchings

Der Name und die Definition der Abfrage, sowie die des gefunden SE der AAS sind dargestellt. Da keine passende Semantic-ID gefunden werden konnte, wurde ein NLP-basiertes Matching durchgeführt. Zur besseren Verständlichkeit wurde die Ähnlichkeit der beiden SEs in einen Score umgewandelt.

Matching information		
Own submodel element	Matched submodel element	
Name: Serial number	Name: Serial number	
Description: unique combination of numbers and letters used to identify the device once it has been manufactured.	Description: unique combination of numbers and letters used to identify the device once it has been manufactured;	
Semantic Id: 0173-1#02-AAM556#002	Semantic Id: 0173-1#02-AAM556#002	
Matching method Semantic equivalent, same se...	Matching algorithm None	Matching score 100 %
<input type="button" value="CLOSE"/>		

Abbildung 6.16: Ergebnis eines Matchings auf Basis gleicher Semantic-IDs

Das zweite Ergebnis zeigt ein Matching basierend auf homogener Semantik. Innerhalb der AAS konnte ein SE gefunden werden, welches die gleiche Semantic-ID wie in der Abfrage verwendet, sodass kein NLP-basiertes Matching durchgeführt werden musste.

### 6.3 Fazit heterogene Semantik

In diesem Kapitel wurde eine NLP-basierte Methode vorgestellt, die es ermöglicht heterogene Semantik automatisiert aufeinander abzubilden. Es wurden verschiedene Möglichkeiten (SEms, EEs, LLMs) untersucht, die zur Abbildung heterogener Semantik eingesetzt werden können. Zur Evaluation wurde ein Datensatz erstellt, welcher aus ECLASS-Merkmalen und händisch erstellten Paraphrasen derselben besteht. Bei der Evaluation konnte ein F1-Score von bis zu 90% erreicht werden. Die erzielten Ergebnisse zeigen, dass dies ein vielversprechender Ansatz ist, um Elemente semantisch heterogener Vokabulare aufeinander abzubilden. Zur Akzeptanz in der Praxis muss die Genauigkeit des Modells jedoch weiter erhöht werden. Diese reicht bspw. zur Anwendung in automatisierungstechnischen Funktionen wie der Prozessführung nicht aus. Der entwickelte Ansatz wurde als I4.0-Service implementiert. Hierzu wurde ein SM-Interface mit drei Operationen spezifiziert. Dieses wurde innerhalb eines I4.0-Services umgesetzt und in einer prototypischen Anwendung implementiert.

In diesem Kapitel wurde die in Kapitel 4.5 herausgearbeitete Herausforderung „Abbildung heterogener semantischer Beschreibungen aufeinander“ adressiert. Die Anforderung 10 aus Kapitel 3 ist ebenfalls erfüllt, da durch die automatisierte Abbildung mittels NLP semantisch heterogene Modelle aufeinander abgebildet werden können.

In der implementierten Anwendung wurde gezeigt, dass sich der Ansatz gut für AAS eignet, die umfangreiche Beschreibungen der Informationen der Komponenten aufweisen, die sie repräsentieren. Bestandsanlagen werden aktuell jedoch zum einen weder durch einen digitalen Zwilling repräsentiert, noch verfügen ihre Datenpunkte über ausführliche textuelle Beschreibungen. Im folgenden Kapitel wird ein allgemeines Konzept vorgestellt, welches AAS für bestehende Anlagen und Komponenten erstellt, deren heterogene Datenpunkte auf ein einheitliches Vokabular abbildet, und auf Basis der gewonnenen Informationen Strukturinformationen generiert. Dieses Konzept wird am Beispiel der TGA konkret umgesetzt.

## 7 Digitale Zwillinge für Bestandsanlagen

In den folgenden Kapiteln wird ein Konzept beschrieben, das bestehende Anlagen mit digitalen Zwillingen ausstattet. Die beiden vorherigen Kapitel 5 und 6 haben gezeigt, wie zum einen Teilmodelle allgemein abgeleitet und spezifiziert werden können und zum anderen, wie heterogene Semantik verarbeitet werden kann. Die beiden Konzepte sind dabei auf neu zu errichtende Anlagen bzw. zu installierende Komponenten anwendbar. Insbesondere das Vorgehensmodell für die Ableitung von Teilmodellen ist für neue Komponenten ausgelegt, die in Zukunft produziert werden. Das Konzept zur Verarbeitung heterogener Semantik basiert auf der Voraussetzung, dass digitale Zwillinge existieren. Diese können dabei unterschiedliche Vokabulare verwenden, jedoch muss die Semantik der einzelnen SEs durch einen Namen und eine Definition definiert sein. Nur so kann das SM durchgeführt werden. Für Bestandsanlagen ist diese Voraussetzung jedoch nicht gegeben. Es existieren keine virtuellen Repräsentanten in Form digitaler Zwillinge für Anlagen verschiedener Industrien wie beispielsweise der TGA, Prozessindustrie, Maschinenbau oder Chemieanlagen. Diese Anlagen haben eine Lebensdauer von mehreren Jahrzehnten. Auch wenn neu zu installierende Komponenten oder Anlagen über einen digitalen Zwilling verfügen, dauert es noch lange, bis die Vision dynamischer und flexibler Wertschöpfungsnetzwerke realisiert werden kann. Bestandsanlagen können derzeit nicht in diese Netzwerke integriert werden. Es wird ein Konzept benötigt, um Bestandsanlagen in I4.0-Umgebungen zu integrieren, damit die Vision dynamischer Netzwerke schneller umgesetzt werden kann. In dieser Arbeit wird ein solches Konzept entwickelt. Zunächst wird das allgemeine Konzept beschrieben (Kapitel 7.1). Es müssen verschiedene Aspekte berücksichtigt werden, um das Konzept erfolgreich umzusetzen. Die herausgearbeiteten Aspekte werden am Beispiel der TGA in den Kapiteln 7.2-7.4 erläutert. Innerhalb des entwickelten Konzepts wird die AAS als digitaler Zwilling verwendet.

### 7.1 Konzept zur Generierung digitaler Zwillinge für Bestandsanlagen

In diesem Kapitel wird zunächst die allgemeine, branchenunabhängige, Idee beschrieben. In den darauf folgenden Kapiteln wird dargestellt, wie dieses grundsätzliche Konzept für Anlagen der TGA umgesetzt wird.

### 7.1.1 Spezifikation des branchenunabhängigen Konzepts

Das übergeordnete Ziel besteht darin, Bestandsanlagen mit digitalen Zwillingen auszustatten, damit sie in I4.0-Wertschöpfungsnetzwerken teilnehmen können. Hierzu ist es nicht ausreichend, die Betriebsinformationen der Bestandsanlagen nur syntaktisch auf einen digitalen Zwilling abzubilden. Damit die Informationen von anderen Teilnehmern des Netzwerks interpretiert werden können, muss ebenfalls semantische Interoperabilität hergestellt werden. Die Zielvorstellung ist in folgender Abbildung 7.1 dargestellt.

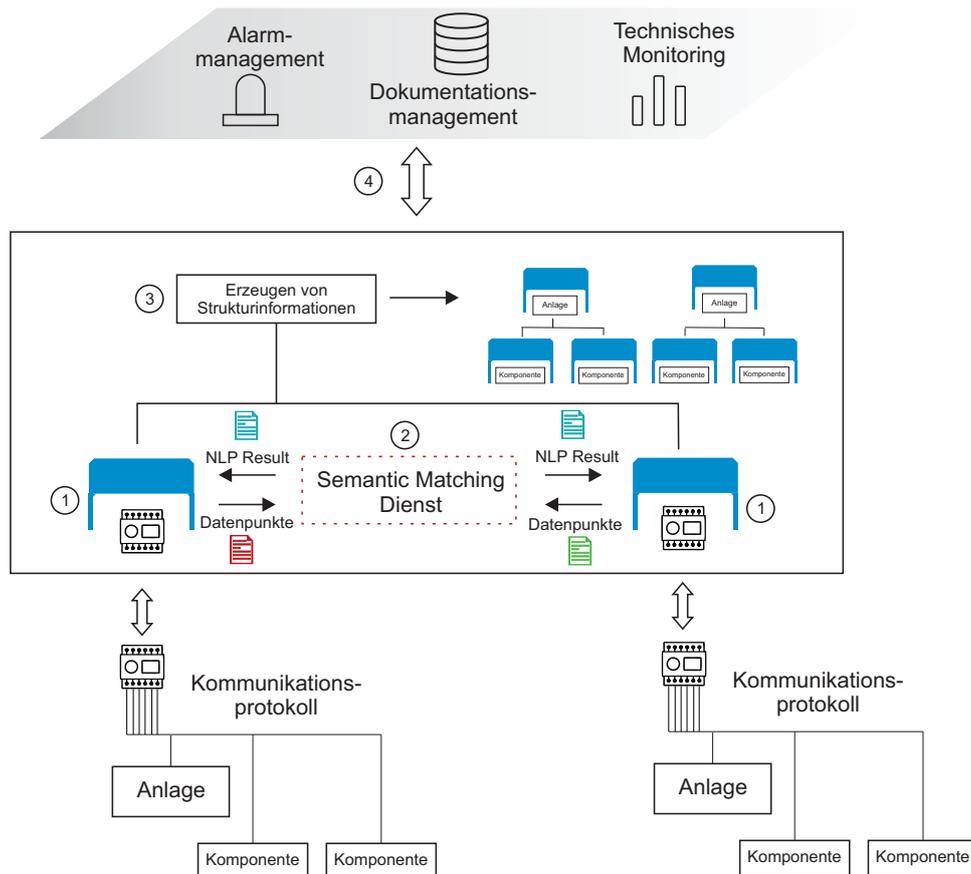


Abbildung 7.1: Integration von Bestandsanlagen in ein I4.0-Wertschöpfungsnetzwerk

Eines der Ziele von I4.0 ist die Implementierung dynamischer und flexibler Wertschöpfungsnetzwerke anstelle starrer Strukturen. Derzeit ist die Kommunikation technischer Systeme in verschiedenen Branchen jedoch durch das starre System der *Automatisierungspyramide* geprägt. Unternehmen teilen Prozesse innerhalb der Automatisierungspyramide auf verschiedene Ebenen auf. Um AAS für Bestandsanlagen zu erstellen, müssen Informationen auf einer der Ebenen erfasst und in Teilmodelle integriert werden. In [211] wird ein Überblick über verschiedene Definitionen dieser Ebenen gegeben. In dieser Arbeit werden Informationen auf der dritten Ebene abgerufen, welche nach [211] häufig als *Steuerungsebene* oder *Prozessleitebene* bezeichnet wird. Im RAMI 4.0 wird diese Ebene als *Control Device* bezeichnet (vgl. Kapitel 4.2.1). Die Ebene dient der Beobachtung und Steuerung physischer

Prozesse und wird durch Speicherprogrammierbare Steuerungen (SPS) umgesetzt [212]. Auf einer SPS werden die Betriebsdaten der Anlagen zusammengeführt. Diese Ebene bietet einen guten Ausgangspunkt zur Generierung von AAS. Die Abbildung 7.1 zeigt symbolisch Anlagen und Komponenten, die über eine SPS gesteuert werden. Die technischen Anlagen kommunizieren mit der SPS über ein Kommunikationsprotokoll. Zunächst wird für jede SPS eine AAS generiert und die Informationen der SPS werden in Teilmodellen bereitgestellt. Hierzu können die Teilmodelle AID und AIMC des IDTA verwendet werden. Diese bilden die Informationen verschiedener Kommunikationsprotokolle als Teilmodelle ab, sodass auf diese einheitlich zugegriffen werden kann. Die Generierung von AAS sowie der Teilmodelle AID und AIMC stellt den ersten Schritt dar (1).

Allerdings ist dieser Schritt allein nicht ausreichend, da die Inhalte der Teilmodelle semantisch nicht einheitlich vorliegen. In Kapitel 6 wurde eine NLP-basierte Methode entwickelt, die Elemente semantisch heterogener Vokabulare aufeinander abbildet. Hierzu wurden der Name und die Definition der jeweiligen Elemente verwendet. Die Betriebsinformationen bestehender SPSen liegen oft nicht in verständlichen Namen und Definitionen vor, sondern in Kennzeichnungsschlüsseln, die spezifisch für diese Datenpunkte und ihre Funktionalitäten vom Hersteller oder Betreiber erstellt wurden. Der Ansatz aus Kapitel 6 ist zur Verarbeitung dieser nicht geeignet. Aus diesem Grund muss eine andere Methode zur Abbildung semantisch heterogener Datenpunkte entwickelt werden. Für die Abbildung der Datenpunkte ist ein Zielvokabular erforderlich. In verschiedenen Industrien werden unterschiedliche Vokabulare verwendet. Aufgrund der Vielfalt an Anlagen, technischen Systemen und Prozessen ist es unrealistisch, ein allumfassendes Zielvokabular zu entwickeln. Vielmehr wird der Ansatz domänenspezifischer Vokabulare verfolgt, welche jeweils das Vokabular eines bestimmten Industriezweigs enthalten. Im Vergleich zu einem allgemeinen Zielvokabular wird der Anwendungsbereich und damit die Komplexität der Aufgabe und Modelle eingeschränkt. Die Aufgabe des SM-Dienstes besteht darin, die heterogenen Datenpunkte auf ein spezifisches Zielvokabular der jeweiligen Domäne abzubilden. Dieser Schritt kann in verschiedenen Domänen unterschiedlich ausgeprägt sein. In dieser Arbeit wird das Konzept am Beispiel der TGA umgesetzt. Für diesen Zweck wurde ein vierstufiger Klassifizierungsalgorithmus entwickelt, der einen Datenpunkt einem TGA-spezifischen Vokabular zuordnet. Als Ergebnis wird ein Teilmodell generiert, das die Ergebnisse des NLP-Algorithmus enthält. Dieser Prozess stellt den zweiten, in der Abbildung 7.1 dargestellten, Schritt dar (2).

Auf Basis des NLP-Teilmodells werden Strukturinformationen generiert. Bisher wurde für jede SPS eine AAS generiert und die Betriebsinformationen wurden mittels NLP verarbeitet. Das Ziel dieser Arbeit sind AAS der tatsächlichen Anlagen und nicht nur der SPS, die diese Anlagen steuern. In dieser Arbeit bezieht sich der Begriff Strukturinformation darauf, dass die Struktur eines Systems dargestellt wird. Es kann sich hierbei um einzelne Anlagen,

Teilanlagen oder ganze Gebäude handeln. Die Entscheidung darüber hängt davon ab, wie groß der betrachtete Systemrahmen ist. Für die Bildung der Strukturinformationen muss ein Konzept entwickelt werden, welches speziell für den jeweiligen Anwendungsfall, also den Industriezweig, ausgeprägt werden muss. In Kapitel 7.4.2 wird ein Konzept zur Bildung von Strukturinformationen innerhalb von Gebäuden und der TGA beschrieben. Das Ziel besteht darin, die physischen Systeme virtuell abzubilden. Die Informationen der NLP-Teilmodelle der AAS der SPSen werden genutzt, um die Systeme und ihre Zusammenhänge als virtuelle Repräsentation nachzubilden (3). Die Informationen der AAS können nun erkundet und in verschiedene Anwendungen wie bspw. das Alarmmanagement oder das TMon eingebunden werden (4). Dies wird in dieser Arbeit prototypisch für eine TMon-Anwendung umgesetzt (vgl. Kapitel 8). Die allgemeingültigen Ansätze dieses Kapitels werden im folgenden Kapitel auf die TGA übertragen.

### **7.1.2 Umsetzung des allgemeinen Konzepts für Anlagen der technischen Gebäudeausrüstung**

Die folgende Abbildung 7.2 zeigt die verschiedenen Komponenten, die im vorherigen Kapitel beschrieben wurden, und nun speziell für die TGA ausgeprägt werden.

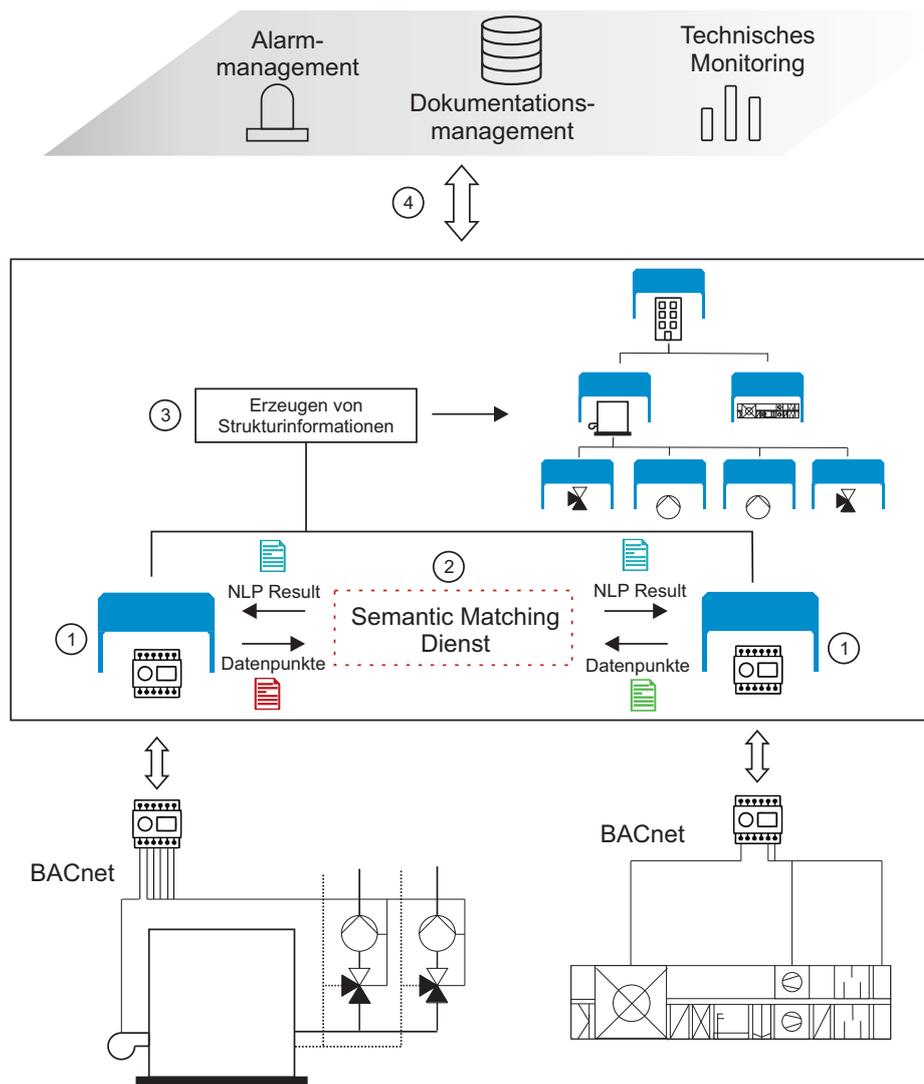


Abbildung 7.2: Generierung von Strukturinformationen für Bestandsanlagen

Die Betriebsinformationen von Anlagen der TGA werden mittels des BACnet-Kommunikationsprotokolls bereitgestellt. Im industriellen Sektor ist die Abkürzung SPS gebräuchlich, während in der GA der Begriff *Direct Digital Control (DDC)* oder *Automationsstation* verwendet wird [133]. Im Folgenden wird der Begriff der Automationsstation genutzt. Für einen einheitlichen Zugriff auf die Informationen werden die Automationsstationen durch eine AAS repräsentiert (1). Damit automatisiert ausführbare Regeln für den Abruf und die Einbindung der Informationen in TMon-Anwendungen implementiert werden können, müssen die Informationen zunächst auf einen einheitlichen semantischen Standard abgebildet werden. Dies wird durch den SM-Dienst erreicht (2). Die heterogenen Datenpunkte werden vom SM-Dienst verarbeitet und auf ein einheitliches Vokabular abgebildet. Auf Basis der nun homogen vorliegenden Informationen (NLP-Result) können Strukturinformationen eines Gebäudes generiert werden (3). Bis zu diesem Punkt werden die Anlagen der TGA durch die AAS der Automationsstationen repräsentiert. Um eine Zugehörigkeit zu übergeordneten Funktionalitäten zu erreichen, müssen Strukturinformationen gebildet

werden. Die Strukturinformationen repräsentieren das Gebäude, Anlagen und dazugehörige Komponenten. Auf diese kann wiederum durch Anwendungen wie das TMon zugegriffen werden (4). Bei der Umsetzung der verschiedenen Aspekte des Konzepts müssen Teilmodelle erstellt werden. Hierbei werden teilweise existierende IDTA-Teilmodelle verwendet, teilweise werden neue Teilmodelle ausgeprägt. Die Vorgehensweise bei der Definition neuer Teilmodelle ist im folgenden Abschnitt beschrieben.

### 7.1.3 Ableitung von Teilmodellen für Bestandsanlagen auf Basis grundlegender Lebenszyklusprozesse

In Kapitel 5.1 wurde ein allgemeines Vorgehensmodell zur Ableitung von Teilmodellen beschrieben. Um Bestandsanlagen in ein I4.0-Netzwerk zu integrieren, werden Teilmodelle benötigt, welche die Datenpunkte der Automationsstationen innerhalb von AAS bereitstellen, die Ergebnisse der Klassifizierung abbilden und die strukturierten Informationen und Anforderungen des TMon repräsentieren. Für die Integration der Datenpunkte in die AAS wurden existierende IDTA-Teilmodelle erweitert (vgl. Kapitel 7.2.1), die anderen Teilmodelle wurden neu spezifiziert und dafür von zwei grundlegenden Lebenszyklusprozessen des Modells nach [166] abgeleitet, welches dem Vorgehensmodell als Basis dient. Der erste Prozess wird als *System-Analysis-Process* bezeichnet, und soll eine Daten- und Informationsbasis für die Bewertung von Systemen schaffen [166]. Von diesem Prozess werden die beiden Teilmodelle *BACnet-Datapoints-Information* und *NLP-Classification-Result* abgeleitet. Das BACnet-Teilmodell repräsentiert die Informationen des GA-Netzwerks, sodass diese einheitlich weiterverarbeitet werden können. Die Daten werden von dem Klassifizierungsalgorithmus verarbeitet, und durch das NLP-Teilmodell zur Verfügung gestellt. Dieses bildet die Basis um die Struktur eines Gebäudes abzuleiten. Diese wird in verschiedenen AAS ausgedrückt, welche teilweise über die Teilmodelle *Operating-Information* und *Technical-Monitoring* verfügen. Diese werden von dem *Operation-Process* aus [166] abgeleitet. Wie in Kapitel 5.2 beschrieben, wird dieser verwendet, um Informationen des Betriebs abzubilden. Ein Aspekt des Betriebsprozesses betrifft zudem das Monitoring der Systeme. Dieses Monitoring wird eingesetzt, um zu überwachen, ob Systeme innerhalb vorgegebener Grenzen arbeiten und wirtschaftlich und nachhaltig eingesetzt werden [166]. Hiervon können Monitoring-Teilmodelle wie das in dieser Arbeit spezifizierte Teilmodell *Technical-Monitoring* abgeleitet werden. Die Teilmodelle der Betriebsinformationen und des Monitorings sind für verschiedene Ebenen und Entitäten spezifiziert worden, sodass es Teilmodelle wie *Operating-Information-Valve* und *Operating-Information-Heating-Circuit*, sowie *Technical-Monitoring-Building* gibt. Die verschiedenen Teilmodelle sind detaillierter in den Kapiteln 7.2.2, 7.4.1, 7.4.3 und 7.4.4 beschrieben.

In den folgenden Kapiteln wird die Abbildung des Kommunikationsprotokolls BACnet auf

die AAS (Kapitel 7.2), die Verarbeitung der semantisch heterogenen Datenpunkte mittels NLP (Kapitel 7.3), und die Generierung von Strukturinformationen (Kapitel 7.4) beschrieben. Auf Basis dieser kann dann ein automatisiertes TMon der Anlagen implementiert werden.

## 7.2 Abbildung des BACnet Protokolls auf die Verwaltungsschale

Um die Anforderungen gemäß Kapitel 3 umzusetzen, müssen Daten aus GA-Netzwerken in I4.0-Systemen verfügbar gemacht werden. Innerhalb der GA werden verschiedene Kommunikationsprotokolle eingesetzt, wie beispielsweise BACnet, KNX oder Modbus. In dieser Arbeit wird ein Ansatz präsentiert, der eine Abbildung von Informationen aus dem BACnet-Protokoll auf die AAS ermöglicht.

### 7.2.1 BaSyx Data Bridge als Basis für die Abbildung von BACnet auf die Verwaltungsschale

Die BaSyx-Data-Bridge ermöglicht den Abruf von Asset-Daten aus Informationsnetzwerken und die Übertragung an einen AAS-Server mittels Teilmodellen. Sie wurde um das BACnet-Kommunikationsprotokoll erweitert [213]. Zur Konfiguration der Data-Bridge sind sowohl Schnittstelleninformationen vom physischen Asset, als auch Mapping-Informationen der AAS, erforderlich. Diese werden in den standardisierten IDTA-Teilmodellen *AID* [158] und *AIMC* [159] beschrieben. Indem diese Teilmodelle in einer AAS dargestellt werden, kann die Data-Bridge auf alle notwendigen Informationen zugreifen, um BACnet-Informationen aus einem Netzwerk abzurufen und an eine AAS zu übertragen. Durch den Einsatz eines BaSyx-AAS-Servers werden AAS als HTTP-Laufzeitumgebung ausgeführt und erfüllen somit die Anforderungen *Gemeinsame Netzwerkinfrastruktur* und *Verfügbarkeit von Informationen für andere Systemteilnehmer* aus Kapitel 3.

### 7.2.2 Teilmodell BACnet Datapoints Information

Eine AAS stellt Informationen über die BACnet-Datenpunkte einer Automationsstation über das Teilmodell *BACnet-Datapoints-Information* bereit. Die Data-Bridge überträgt BACnet-Informationen des GA-Netzwerks an dieses Teilmodell. Das Teilmodell wurde auf Basis des BACnet zugrunde liegenden Informationsmodells [136] spezifiziert. Jedes BACnet-Objekt wird als SMC im Teilmodell dargestellt und enthält BACnet-Eigenschaften als SEs (vgl. Abbildung 7.3).

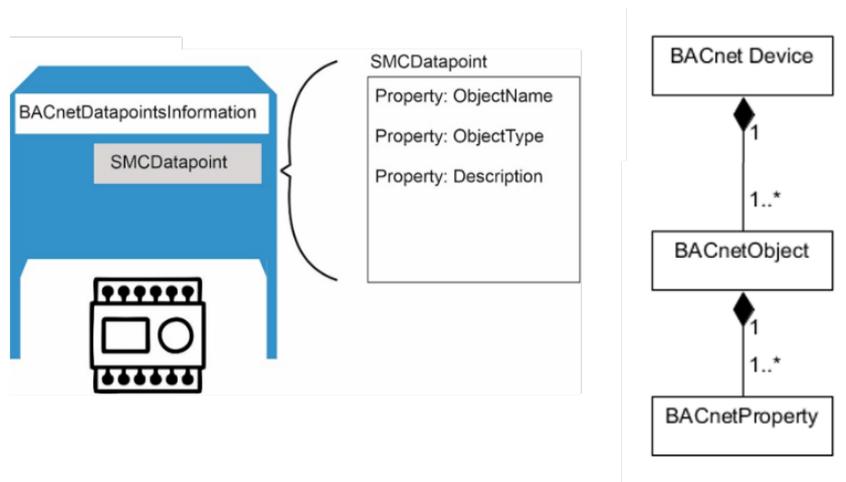


Abbildung 7.3: Visualisierung Teilmodell BACnet Datapoints Information

Durch die Verfügbarkeit des BACnet-Teilmodells werden die Anforderungen 6 und 7 erfüllt, wie sie in Kapitel 3 über die Verfügbarkeit von Modellen beschrieben werden. Anforderung 6 bezieht sich dabei auf Informationen und Fähigkeiten eines Teilnehmers. Die Bereitstellung der Informationen durch einen virtuellen Vertreter wird gemäß Anforderung 7 durch die Integration des Teilmodells in eine AAS und die Verwendung des BaSyx AAS-Servers erreicht.

### 7.3 Natural Language Processin zur automatisierten Abbildung semantisch heterogener Datenpunkte

Die Daten der Automationsstationen sind über das Teilmodell *BACnet-Datapoints-Information* einheitlich zugänglich. Um Anforderung 8 aus Kapitel 3 zu erfüllen, müssen diese Informationen auch von anderen Teilnehmern interpretiert werden können, um sie weiterverarbeiten zu können. Das *BACnet-Datapoints-Information*-Teilmodell genügt dieser Anforderung jedoch nicht. Das *AID*-Teilmodell und das *AIMC*-Teilmodell bilden die relevanten Daten der GA auf die AAS ab. Es gibt jedoch keine semantische Abbildung auf einen Standard, sodass die Datenpunkte weiterhin semantisch heterogen vorliegen. Um die Datenpunkte in TMon-Anwendungen zu integrieren, müssen der Object-Name und andere BACnet-Eigenschaften manuell analysiert und in eine Anwendung integriert werden, auch wenn sie bereits innerhalb einer AAS verfügbar sind. Semantische Modelle (Anforderung 9) sind nur verfügbar, wenn AKS, wie sie beispielsweise in der VDI 3814 [214] beschrieben werden, verwendet werden. Die Einhaltung standardisierter AKS ist nicht gesetzlich vorgeschrieben, es existieren zudem eine Vielzahl von hersteller- oder betreiberspezifischen Schlüsseln. Da hierbei unterschiedliche semantische Modelle entstehen, ist gemäß Anforderung 10 ein Mechanismus erforderlich, der diese Modelle automatisiert

aufeinander abbildet. Zur Erreichung dieses Ziels wurde eine Methode entwickelt, um die semantischen Informationen von Datenpunkten zu verarbeiten.

### 7.3.1 Entwurf vierstufiger Klassifizierungsprozess

Ziel ist es jeden Datenpunkt aus diversen BACnet-Netzwerken in ein einheitliches Vokabular zu überführen. Zur Umsetzung wurde ein Klassifizierungsalgorithmus mit vier Ebenen entwickelt (vgl. Abbildung 7.4).



Abbildung 7.4: Überblick über den vierstufigen Klassifizierungsprozess mit den Stufen Grundfunktion, Unterfunktion, Entität und Datenpunkt

Ein Gebäude besteht aus verschiedenen Systemen, die unterschiedliche Funktionen erfüllen. Zum Beispiel benötigen die meisten Gebäude eine Wärmeversorgung. Die Funktion, die ein Gebäude erfüllen muss, ist somit mit *Wärme versorgen*. Jeder Datenpunkt wird zunächst einer Grundfunktion zugeordnet. Anschließend werden die Grundfunktionen in Unterfunktionen unterteilt, die genauere Funktionsspezifikationen liefern. Die Funktion mit *Wärme versorgen* lässt sich in die vier Unterfunktionen *Wärme erzeugen*, *Wärme verteilen*, *Wärme beziehen* und *Wärme speichern* unterteilen. Um diese Funktionen in Gebäuden auszuführen, können verschiedene Systeme oder technische Komponenten eingesetzt werden. So kann z. B. die Wärmeerzeugung durch diverse Wärmeerzeuger wie Wärmepumpen oder Pelletkessel realisiert werden. Daher wird auf dieser Ebene der Klassifizierung jedem Datenpunkt seine zugehörige Entität (welche die entsprechende Funktion erfüllt) zugewiesen. Da Datenpunkte auf dieser Ebene nicht nur technischen Komponenten, wie bspw. Pumpen, zugeordnet werden, sondern auch übergeordneten Systemen, wie bspw. einem Heizkreis, wird der Begriff Entität und nicht Komponente verwendet. Auf der vierten Ebene wird jedem Datenpunkt eine eindeutige Bezeichnung aus einem einheitlich definierten Standard möglicher Datenpunktbezeichnungen zugewiesen. In Abbildung 7.5 ist der Ablauf des Klassifizierungsprozesses beispielhaft dargestellt.

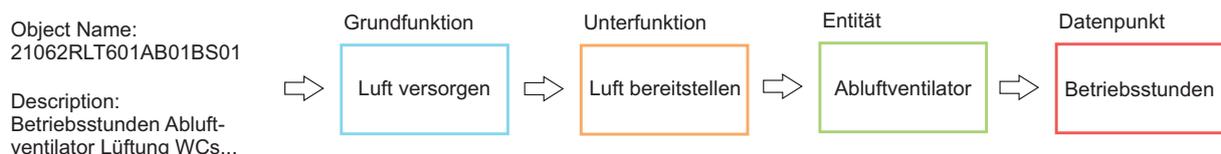


Abbildung 7.5: Beispielhafte Klassifizierung eines Datenpunkts bestehend aus Object Name und Description

Die Eingabe für den vierstufigen Prozess besteht aus dem Objektname (Object-Name) und der entsprechenden Beschreibung (Description) aus BACnet. Der Datenpunkt wird

zunächst einer Grundfunktion zugeordnet und anschließend einer Unterfunktion, einer Entität und schließlich dem entsprechenden Element im Vokabular. Die Umsetzung dieser Methode erfordert das Training von NLP-Modellen. Hierzu muss zunächst ein Datensatz erstellt werden.

### **7.3.2 Datensatz zum Training der Modelle**

Für den Datensatz wurden Datenpunkte (BACnet-Objekte) aus bestehenden Systemen der Städte Köln und Hamburg verwendet. Für diese Arbeit wurden die Objekttypen auf Analog-Input, -Output, -Value, Binary-Input, -Output, -Value und Accumulator beschränkt. Insgesamt wurden 54.125 Datenpunkte klassifiziert, dargestellt in Tabelle 7.1.

Tabelle 7.1: Übersicht über den erstellten Datensatz mit Namen und Anzahl der Vorkommnisse aus den drei Ebenen: Grundfunktion, Unterfunktion, Entität

Grundfunktion	Unterfunktion	Entität
Wärmeversorgen (22.540)	Wärme erzeugen (2.147)	BHKW (304), Kessel (1.182), Pelletkessel (33), Wärmepumpe (46), Wärmeversorger allgemein (582)
	Wärme verteilen (19.758)	Druckhaltestation (7), Heizkreis allgemein (3.734), Heizkurve (1.818), Kältemengenzähler (6), Pumpe (5.266), Raum (2.672), Regler (1.427), Rücklauf (759), Übertrager (39), Ventil (1.638), Vorlauf (2.003), Wärmemengenzähler (97), Warmwasserbereitung (292)
	Wärme speichern (176)	Speicher (176)
	Wärme beziehen (459)	Fernwärme (459)
Luftversorgen (14.849)	Luft bereitstellen (11.331)	Abluft Allgemein (766), Abluftfilter (105), Abluftklappe (87), Abluftventilator (1.900), Außenluftfilter (116), Außenluftklappe (465), Befeuchter (110), Erhitzer (236), Filter (12), Fortluftklappe (454), Gerät allgemein (1.660), Kältemengenzähler (4), Klappe allgemein (87), Kühler (141), Regler (1.850), Umluft (39), Ventilator (155), Wärmemengenzähler (5), Wärmerückgewinnung (516), Zuluft allgemein (1.185), Zuluftfilter (43), Zuluftklappe (92), Zuluftventilator (1.291)
	Luft verteilen (3.518)	Auslass (49), Raum (1.632), Volumenstromregler Abluft (834), Volumenstromregler Raum (87), Volumenstromregler Zuluft (916)
Medienversorgen (736)	Medien bereitstellen (106)	Bereitstellung allgemein (23), Dosieranlage (14), Entgasung (4), Enthärtung (9), Entsalzung (1), Frischwassermodul (6), Kraftstoffreinigung (5), Nachfüllstation (10), Präsenzmelder (2), Regler (11), Thermische Desinfektion (12), Wasseraufbereitung (9)
	Medien entsorgen (91)	Drainage (2), Fettabscheider (6), Hebeanlage (52), Neutralisation Kondensat (4), Regenwasserpumpe (3), Schmutzwasserpumpe (20), WC Abwasser (4)
	Medien speichern (51)	Speicher (51)

Tabelle 7.1 - fortgesetzt

Grundfunktion	Unterfunktion	Entität
Medien versorgen (736)	Medien verteilen (488)	Druckhaltestation (15), Hygienespülung (257), Kraftstoffpumpe (6), Rohrbegleitheizung (5), Ventil (29), Wasserzähler (18), Zirkulation allgemein (34), Zirkulationspumpe (124)
Kälte versorgen (1.319)	Kälte erzeugen (722)	Kälteanlage (29), Kältekreis allgemein (6), Kältemaschine (284), Kältemengenzähler (19), Klappe (60), Pumpe (56), Regler (41), Rückkühlwerk (87), Rücklauf (18), Ventil (8), Vorlauf (11), Wärmemengenzähler (3)
	Kälte verteilen (582)	Frequenzumrichter (1), Kältekreis allgemein (11), Kältemaschinenanschluss (28), Klappe (229), Pumpe (81), Raum (69), Regler (13), Rücklauf (54), Umluftkühlgerät (3), Ventil (21), Vorlauf (61), Wärmemengenzähler (11)
	Kälte speichern (15)	Eisspeicher (3), Eisspeicheranschluss (6), Kältemengenzähler (6)
Strom versorgen (3.531)	-	-
Sichern (9.337)	Sichern (9.337)	Brandmeldeanlage (117), Brandschutzklappe (8.848), Entrauchung-Ventilator (26), Einbruchmeldeanlage (3), Feuerlöschanlage (10), Gaswarnanlage (3), Notruf (7), Rauchmeldeanlage (334)
Transport (15)	-	-
Andere Anlagen (1.798)	-	-

Für jeden Datenpunkt musste der vierstufige Labelingprozess durchgeführt werden. Mögliche Grundfunktionen waren mit *Wärme-*, *Luft-*, *Kälte-*, *Medien-* und *Strom versorgen* sowie *Transport* und *Sichern*. Datenpunkte mit nicht nachvollziehbaren Namen oder Beschreibungen, die keiner Funktion zugeordnet werden konnten, wurden in die Gruppe *Andere Anlagen* einsortiert. Indem unverständliche Datenpunkte der Klasse *Andere Anlagen* zugeordnet und nicht gelöscht wurden, sollen die Modelle lernen, Datenpunkte, die nicht den sieben Grundfunktionen entsprechen, als Sonstige zu klassifizieren. Die Grundfunktionen wurden weiter in Unterfunktionen unterteilt (zweite Spalte). Die Grundfunktion *Sichern* hat keine weitere Unterfunktion, sodass diese Ebene übersprungen und eine direkte Klassifizierung auf der Entitätsebene durchgeführt wird. Für die Grundfunktionen *Transport* (keine aus-

reichenden Daten im Datensatz vorhanden), mit *Strom versorgen* (Datenpunkte können nicht weiter zugeordnet werden) und *Andere Anlagen* (Datenpunkte können nicht weiter zugeordnet werden) wurden keine weiteren Ebenen definiert. Jeder Datenpunkt innerhalb der Funktion der zweiten Ebene wurde einer Entität zugeordnet (dritte Spalte). Die Tabelle listet für jede Entität die Anzahl ihrer Vorkommnisse im Datensatz auf. Im letzten Schritt wurde die konkrete Funktion der Datenpunkte beschrieben. Dabei wurde für alle Entitäten eine einheitliche Terminologie verwendet, um eine klare und verständliche Darstellung zu ermöglichen. Ein Beispiel hierfür ist der Schaltbefehl für einen Abluftventilator, der nicht als *Schaltbefehl Abluftventilator* bezeichnet wird, sondern als *Schaltbefehl*. Dies ermöglicht die Wiederverwendung des Merkmals Schaltbefehl für andere Komponenten wie Wärmeerzeuger, Pumpen und Ventilatoren. Dadurch müssen weniger Label erstellt werden, was zu einem einfacheren Labelingverfahren führt. Insgesamt wurden 38.708 Datenpunkte jeweils einem von 467 verfügbaren Datenpunktlabeln zugeordnet. Die Liste aller möglichen Datenpunkte, sowie die Datenpunktzugeordnungen zu den jeweiligen Entitäten, können dem Anhang C entnommen werden.

### 7.3.3 Training und Evaluation der Modelle

Für das Finetuning muss zunächst ein PLM ausgewählt werden. Es wurden nur deutsche Modelle berücksichtigt, da der Datensatz aus deutschen Namen und Beschreibungen besteht. In [215] werden mehrere deutsche Modelle beschrieben. Von diesen wurde das *gBERT-base-model* aufgrund von vorab durchgeführten Tests auf einem Ausschnitt des Datensatzes ausgewählt, da es die besten Ergebnisse erzielte. Das Modell verwendet eine BERT-basierte Architektur und ist ein Encoder-Modell. Encoder-Modelle eignen sich insbesondere für Klassifizierungsaufgaben wie TC. Wie in Kapitel 6.1.2 beschrieben, kann durch ein erweitertes Pretraining auf domänenspezifischen Texten eine bessere Vorhersagegenauigkeit erreicht werden [187]. Das Basismodell *gBERT-base* wurde auf Texten wie Wikipedia trainiert, um ein allgemeines Sprachverständnis zu erlangen. Es verfügt jedoch lediglich über rudimentäre technische Kenntnisse. In einem, im Zuge dieser Arbeit, durchgeführten Projekt mit den Verbänden DIN und VDE wurde innerhalb der *Initiative Digitale Standards (IDiS)* ein verlängertes Pretraining auf deutschen Normen durchgeführt [216]. Es wurden Datensätze mit technisch relevanten Passagen aus circa 25.000 deutschen DIN- und 7.000 deutschen VDE-Normen erstellt. Basierend auf dem Modell *gBERT-base* wurde ein erweitertes Pretraining durchgeführt, welches dem in Kapitel 6.1.2 beschriebenen Prinzip folgt. Zur Überprüfung, ob das erweiterte Pretraining bessere Ergebnisse liefert, wurde das in diesem Projekt trainierte Modell (im Weiteren bezeichnet als *IDiS*) als Vergleichsvariante zu dem allgemeinen *gBERT-base* genutzt. Die Modelle für die ersten drei Ebenen (Grundfunktion, Unterfunktion und Entität) wurden mittels des TC-Anwendungsfalls trainiert. Die letzte Ebene, die Datenpunktebene, wurde

mittels eines Natural-Language-Inference (NLI)-Ansatzes trainiert. Die Änderung der Finetuning Aufgabe für die letzte Klassifizierungsebene der Datenpunkte ist einerseits auf die hohe Anzahl der resultierenden Modellen und andererseits auf den Mangel an Trainingsdaten für die einzelnen Entitäten zurückzuführen. Tabelle 7.1 zeigt, dass insgesamt 111 Entitäten in der dritten Stufe als mögliche Entitäten in Frage kommen. Würde für die letzte Stufe der Datenpunktklassifizierung der TC Ansatz verwendet werden, müsste für jede der 111 Entitäten ein separates Modell trainiert werden. Des Weiteren sind die Datenpunkte für die aufgeführten Komponenten ungleichmäßig verteilt, sodass beispielsweise der Entität *Heizkreis allgemein* 3.734 Datenpunkte zugeordnet sind, während die *Abluftklappe* nur 87 Datenpunkte aufweist. Unterrepräsentierte Entitäten verfügen nicht über ausreichende Daten, um leistungsstarke Modelle zu trainieren. Der Einsatz der NLI-Methode ermöglicht das Training eines einzigen Modells, welches für jede Entität verwendet werden kann, dargestellt in Abbildung 7.6.

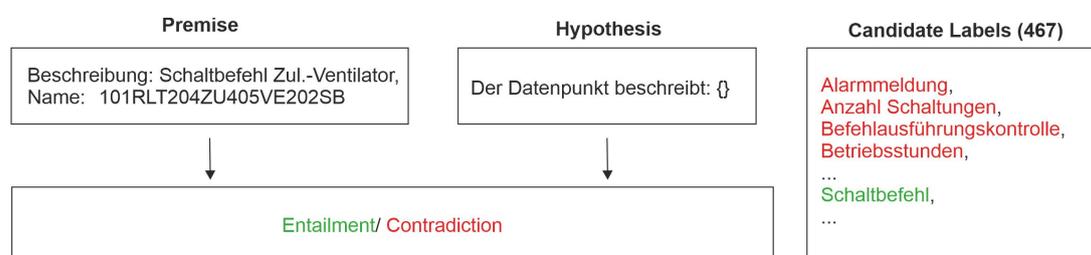


Abbildung 7.6: Beispiel für die entwickelte Methodik zur Klassifizierung eines Datenpunkts mithilfe von NLI

Der NLI-Ansatz nutzt den Namen und die Beschreibung des Datenpunkts als Prämisse (*Premise*) und vergleicht diese mit einer Hypothese (*Hypothesis*), bestehend aus einer statischen Einleitung *Der Datenpunkt beschreibt:* und einem variablen *Candidate-Label*, bspw. *Alarmmeldung*. Das Modell erlernt daraufhin die Vorhersage, ob die Hypothese zum Datenpunkt passt (*Entailment*) oder nicht (*Contradiction*). Da es 467 Label zu berücksichtigen gibt, kann jede Prämisse mit 467 Hypothesen (1 Entailment und 466 Contradictions) verglichen werden. Aufgrund der großen Anzahl an Trainingsdaten (38.708), die so etwa 19 Millionen Trainingsbeispiele erzeugen würden, wurde die Anzahl der Hypothesen jedoch reduziert. Hierbei wurden Variationen in der Anzahl der Contradictions (5, 10, 20) durchgeführt, um die optimale Anzahl an Contradictions zu ermitteln, die für das Training eines leistungsstarken Modells erforderlich sind. Durch den vierstufigen Klassifizierungsprozess kann im Anwendungsfall die Anzahl der in Frage kommenden Candidate-Labels auf die Anzahl reduziert werden, die für die jeweilige Entität relevant sind. Ein Datenpunkt der Entität *Pumpe* kann bspw. auf eins von 43 möglichen Labels abgebildet werden (z.B. Alarmmeldung, Schaltbefehl, Messwert Leistungsaufnahme), sodass die Beschreibung des Datenpunktes mit 43 Candidate-Labels verglichen wird. Die Zuordnung der möglichen Datenpunkte zu den jeweiligen Entitäten ist im Anhang C enthalten.

Um die beste Variante zu ermitteln, wurden für jede Klasse und Ebene mehrere Modellvarianten trainiert. Da der Prozess des Labeling sehr komplex und zeitaufwändig ist, wurde untersucht, wie viele Trainingsbeispiele benötigt werden, um einen ausreichend hohen F1-Score zu erreichen. Als Schwellenwerte wurden F1-Scores von 95% und 99% herangezogen. Die Variation der Anzahl der Trainingsdaten wurde zusätzlich mit einer unterschiedlichen Anzahl an Epochen und der Größe der Batch-Size kombiniert (vgl. Tabelle 7.2).

Tabelle 7.2: Variation von Hyperparametern für jeden Klassifizierungsschritt, die zum Trainieren verschiedener Modelle verwendet werden

Parameter	Grundfunktion	Unterfunktion	Entität	Datenpunkt
Anzahl Trainingsbeispiele	500, 1.000, 2.500, 5.000, 10.000, 20.000, 30.000, 43.444	500, 1.000, 2.500, 5.000, 10.000, Maximum	100, 500, 1.000, 2.500, 5.000, 10.000, Maximum	1.000, 5.000, 10.000, 25.000, 30.966
Epochen	3, 5	3, 5	3, 5	2
Batch-Size	64, 128	32, 64	8, 16, 32, 64	128
Contradictions	-	-	-	5, 10, 20

Bei der NLI-Klassifizierung auf Datenpunktebene wurden die Anzahl der Epochen und die Batch-Size konstant gehalten, da aufgrund der hohen Anzahl an Trainingsdaten keine signifikanten Unterschiede, durch eine Variation dieser Parameter, im Ergebnis zu erwarten waren. Stattdessen wurde die Anzahl der Trainingsdaten variiert und mit einer Variation an Contradictions kombiniert. Die einzelnen Datensätze wurden für das Training in Trainings-, Test- und Validierungsdatensätze mit einer Verteilung von 80%, 10% und 10% aufgeteilt. In den folgenden Tabellen ist der F1-Score des Validierungssatzes enthalten, welcher nicht zum Training der Modelle verwendet wurde und somit zur Bewertung dieser hinsichtlich ihrer Vorhersagegenauigkeit auf unbekanntem Daten verwendet werden kann. Tabelle 7.3 gibt eine Übersicht über die Anzahl der notwendigen Trainingsbeispiele, um einen F1-Score von 95% oder 99% zu erreichen.

Tabelle 7.3: Übersicht über die Anzahl der Trainingsbeispiele und anderer Hyperparameter, die zum Erreichen der angegebenen Schwellenwerte erforderlich sind

Ebene Klassi- fikation	Daten- satz	Basis- modell	Grenz- wert in %	Anzahl Bei- spiele	Batch- Size	Epoch- en	Anzahl Label	Beispiele pro La- bel
Grund- funktion	Grund- funktion	gBERT- base	95	1.000	128	5	8	125
			99	10.000	64	5	8	1.250
		IDiS	95	500	128	5	8	63
			99	10.000	64	3	8	1.250
Unter- funktion	Wärme versor- gen	gBERT- base	95	500	64	5	4	125
			99	5.000	64	5	4	1.250
		IDiS	95	500	64	5	4	125
			99	5.000	64	3	4	1.250
	Luft versor- gen	gBERT- base	95	500	32	5	2	250
			99	2.500	64	3	2	1.250
		IDiS	95	500	64	5	2	250
			99	2.500	64	3	2	1.250
	Medien versor- gen	gBERT- base	95	610	64	3	4	153
			99	-	-	-	-	-
		IDiS	95	-	-	-	-	-
			99	-	-	-	-	-
	Kälte versor- gen	gBERT- base	95	500	32	3	3	167
			99	-	-	-	-	-
		IDiS	95	500	16	3	3	167
			99	-	-	-	-	-
	Sichern	gBERT- base	95	100	64	3	8	13
			99	1.000	64	5	8	125
		IDiS	95	100	8	5	8	13
			99	500	16	5	8	63

Tabelle 7.3 - fortgesetzt

Ebene Klassifikation	Datensatz	Basismodell	Grenzwert in %	Anzahl Beispiele	Batch-Size	Epochen	Anzahl Label	Beispiele pro Label
Entität	Wärme erzeugen	gBERT-base	95	100	32	5	5	20
			99	1.000	32	5	5	200
		IDiS	95	100	8	5	5	20
			99	1.000	16	5	5	200
	Wärme verteilen	gBERT-base	95	1.000	32	3	13	77
			99	15.816	32	3	13	1.217
		IDiS	95	500	32	5	13	38
			99	10.000	16	5	13	769
	Luft bereitstellen	gBERT-base	95	2.500	32	3	23	109
			99	-	-	-	-	-
		IDiS	95	2.500	3	16	23	109
			99	-	-	-	-	-
	Luft verteilen	gBERT-base	95	500	16	3	5	100
			99	2.500	32	3	5	500
		IDiS	95	500	32	3	5	100
			99	2.500	16	3	5	500
	Kälte erzeugen	gBERT-base	95	500	64	3	12	42
			99	-	-	-	-	-
		IDiS	95	500	16	3	12	42
			99	-	-	-	-	-

Es wurden nur Modelle für Klassen trainiert, für die mehr als 500 Trainingsbeispiele verfügbar waren. Keine Modelle wurden demnach für die Klassen *Kälte verteilen*, *Kälte speichern*, *Medien bereitstellen*, *Medien entsorgen*, *Medien verteilen* und *Medien speichern* trainiert. Um eine Aussage treffen zu können, wie viele Trainingsbeispiele durchschnittlich über alle Klassen hinweg benötigt werden, um eine Genauigkeit von 95% bzw. 99% zu erreichen, wurden der *Median* und der *Mittelwert* für die *gBERT*- und *IDiS*-Modelle, sowie für die Kombination beider, berechnet (vgl. Tabelle 7.4).

Tabelle 7.4: Mittelwert und Median der Anzahl der Trainingsbeispiele pro Label, die zum Erreichen der angegebenen Schwellenwerte erforderlich sind

Modell	Mittelwert 95%	Median 95%	Mitellwert 99%	Median 99%
gBERT	107	109	828	1.217
IDiS	93	81	755	827
Kombination	100	100	791	993

Um Unterschiede zwischen verschiedenen Klassifizierungsebenen nicht zu stark zu gewichten, wurde neben dem Mittelwert auch der Median berücksichtigt. Bei der Klassifizierung der Grundfunktion *Sichern* benötigt das *gBERT*-Modell bspw. lediglich 13 (95%) und 125 (99%) Label pro Klasse. Innerhalb der Klasse *Sichern* macht die *Brandschutzklappe* etwa 95% der Datenpunkte aus (vgl. Tabelle 7.1). Im Vergleich zur Klasse *Wärme verteilen* werden bspw. deutlich weniger Daten benötigt, um die gewünschten Ergebnisse zu erzielen. Bei einem Vergleich der *gBERT*- und *IDiS*-Modelle zeigt sich, dass letztere im Durchschnitt etwas weniger Trainingsdaten benötigen, um 95% bzw. 99% zu erreichen. Dies deutet darauf hin, dass durch die Erweiterung des Pretrainings auf deutschen Standards eine bessere Anpassung der Modelle an technische Domänen erzielt werden kann. Ein kombinierter Median beider Modelltypen gibt einen guten Hinweis auf die nötigen Datenpunkte. Um einen F1-Score von 95% oder 99% zu erreichen, müssen jeweils etwa 100 oder 1.000 Datenpunkte pro Label und Klasse erfasst werden. Es fällt auf, dass für eine Steigerung des F1-Score um vier Prozentpunkte die Anzahl der Daten verzehnfacht werden muss. Automatisierte TMon-Anwendungen erfordern eine beinahe 100-prozentige Genauigkeit, um vollständig automatisiert erstellt werden zu können. Um diese Genauigkeit zu erreichen, ist zunächst ein deutlich erhöhter Aufwand für den Labelingprozess erforderlich. Abbildung 7.7 veranschaulicht die Beziehung zwischen der Anzahl der Trainingsbeispiele und der Steigerung des F1-Scores am Beispiel der Grundfunktionen des *gBERT*-Modells.

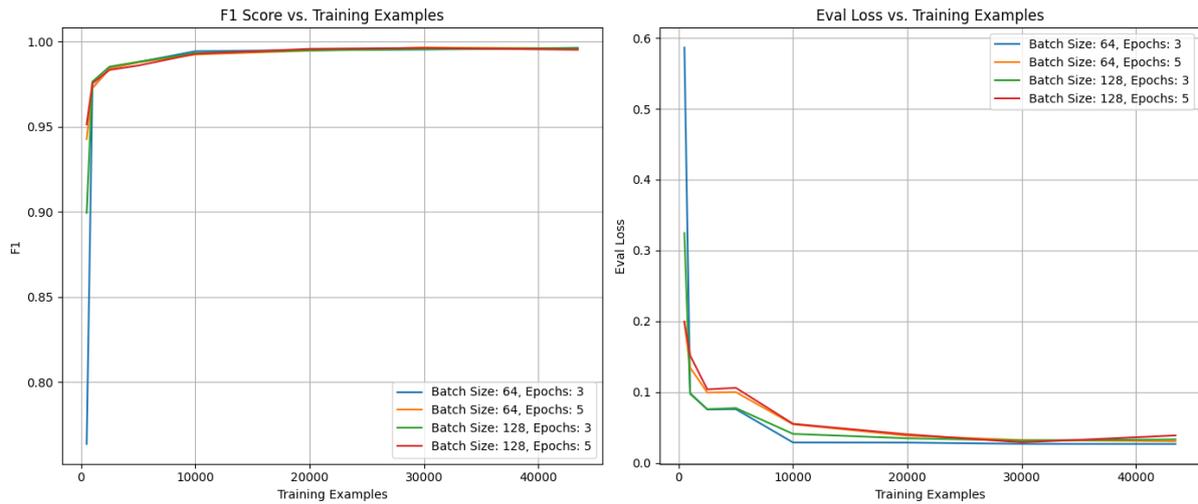


Abbildung 7.7: Entwicklung des F1-Score und Evaluation-Loss in Kombination mit zunehmender Anzahl von Trainingsbeispielen (gBERT Grundfunktionen)

Die linke Seite der Abbildung zeigt den Verlauf des F1-Scores in Abhängigkeit von der Anzahl der Trainingsbeispiele. Es wird deutlich, dass der F1-Score zu Beginn bei einer Erhöhung der Trainingsdaten stark ansteigt. Über etwa 10.000 Trainingsbeispiele hinaus verlangsamt sich der Anstieg jedoch und ist weniger stark ausgeprägt. Ähnlich verhält es sich mit dem absteigenden Trend des *Evaluation-Loss* (Verlustfunktion). Zunächst sinkt der Verlust stark, doch nimmt die Abnahme mit zunehmender Anzahl an Trainingsdaten ab. Ergebnisse für die unterschiedlichen Modelle (*IDiS* und *gBERT*) können dem Anhang D entnommen werden. Zur Klassifikation der Datenpunkte wurde der NLI-Trainingsansatz genutzt. Wie aus Abbildung 7.8 ersichtlich wird, variiert der F1-Score in Abhängigkeit von der Anzahl der Trainingsbeispiele und Contradictions.

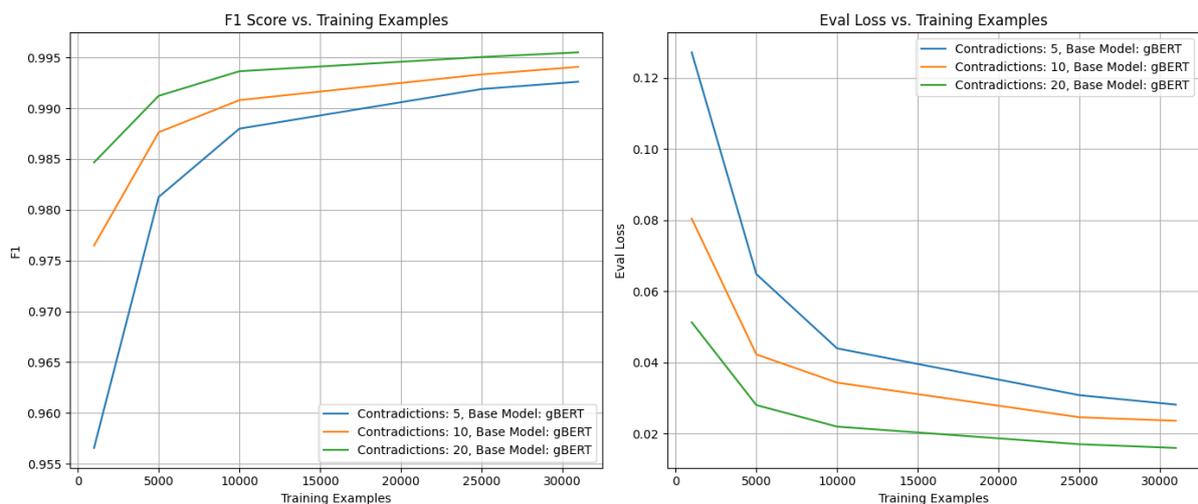


Abbildung 7.8: Entwicklung des F1-Score und Evaluation-Loss in Kombination mit zunehmender Anzahl von Trainingsbeispielen (gBERT Datenpunkte)

Die Abbildung verdeutlicht, dass eine Erhöhung der Anzahl an Trainingsbeispielen und

Contradictions zu besseren Ergebnissen führt. Ähnlich wie bei den mit TC trainierten Modellen kann festgestellt werden, dass mit einer Anzahl von 1.000 Datenpunkten ein F1-Score von 95% erreicht werden kann. Um den Score auf 99% zu steigern, bedarf es einer deutlich größeren Menge an Datenpunkten. Die Erhöhung der Contradictions führt ebenfalls zu einer Verbesserung des F1-Scores. Über die gesamte Anzahl der Trainingsbeispiele hinweg, werden die besten Ergebnisse mit 20 Contradictions erreicht, während 5 Contradictions die schlechtesten Ergebnisse liefern.

Innerhalb aller Klassifizierungsebenen sind die Daten unausgeglichen verteilt. Bspw. treten die Klassen mit *Wärme-* (22.540) und *Luft versorgen* (14.849) in der Grundfunktionsebene sehr häufig auf, während die Klasse mit *Medien versorgen* nur 736 mal vertreten ist. Um die ungleiche Verteilung zu berücksichtigen, wurde der F1-Score als Metrik verwendet, da dieser im Vergleich zu anderen Metriken wie der Genauigkeit, die in dem Datensatz unausgewogene Verteilung berücksichtigt [217]. Der Datensatz wurde auf Basis realer Gebäude erstellt und spiegelt somit die Realität wider. Dennoch wurden zusätzliche Versuche durchgeführt, um zu untersuchen wie sich eine gleichmäßige Verteilung der Daten auf die Vorhersagegenauigkeit der Modelle auswirkt. Es existieren verschiedene Techniken, um mit unausgeglichenen Datensätzen umzugehen [217]. Im Folgenden wird die Methode des *Undersampling* verwendet. Dabei werden die vorhandenen Daten auf die am wenigsten vertretene Kategorie skaliert, um einen ausgewogenen Datensatz zu erhalten. Für jeden Modelltyp wurden maximal 50, 100 und 200 Trainingsbeispiele pro Kategorie verwendet. Es standen jedoch nicht für alle Klassen innerhalb einer Stufe genügend Daten zur Verfügung, um Trainingsdurchläufe mit der entsprechenden Anzahl an Trainingsbeispielen durchführen zu können. Die Klasse *Transport* verfügt bspw. nur über 15 Datenpunkte (12 im Trainingsdatensatz). Die Versuche für die Ebene der Grundfunktion wurden mit 362 Datenpunkten (7 mal 50 pro Kategorie und 12 für *Transport*), 712 Datenpunkten und 1.412 Datenpunkten durchgeführt. Bei den anderen Stufen wurde ebenso verfahren, wenn nicht genügend Trainingsdaten einer Klasse vorhanden waren. Die Ergebnisse sind in Tabelle 7.5 aufgeführt.

Tabelle 7.5: Ergebnisse für den Validierungsdatensatz des undersampled Ansatzes mit 50, 100 und 200 Beispielen pro Klasse

Ebene Klassi- fikation	Daten- satz	Basismodell	F1-Score in %, max. 50 Bei- spiele pro Klas- se (Trainings- beispiele)	F1-Score in %, max. 100 Bei- spiele pro Klas- se (Trainings- beispiele)	F1-Score in %, max. 200 Bei- spiele pro Klas- se (Trainings- beispiele)
Grund- funktion	Grund- funktion	gBERT-base	90,83 (362)	94,13 (712)	96,18 (1.412)
		IDiS	93,70 (362)	93,86 (712)	96,74 (1.412)
Unter- funktion	Wärme versorgen	gBERT-base	90,50 (200)	96,17 (400)	97,08 (733)
		IDiS	91,72 (200)	95,82 (400)	97,37 (733)
	Luft versorgen	gBERT-base	81,08 (100)	93,18 (200)	96,09 (400)
		IDiS	85,59 (100)	93,93 (200)	96,41 (400)
	Medien versorgen	gBERT-base	80,68 (188)	88,13 (303)	92,35 (403)
		IDiS	90,25 (188)	93,84 (303)	93,84 (403)
	Kälte versorgen	gBERT-base	94,30 (112)	97,73 (212)	97,73 (412)
		IDiS	92,01 (112)	96,97 (212)	98,48 (412)
	Sichern	gBERT-base	98,23 (193)	99,23 (337)	99,52 (537)
		IDiS	95,79 (193)	99,44 (337)	99,70 (537)
Entität	Wärme erzeugen	gBERT-base	97,30(209)	97,30 (359)	99,09 (659)
		IDiS	97,76 (209)	96,90 (359)	99,09 (659)
	Wärme verteilen	gBERT-base	92,63 (540)	96,35 (1.020)	97,65 (1.920)
		IDiS	95,9 (540)	96,92 (1.020)	97,74 (1.920)
	Luft be- reitstellen	gBERT-base	89,94 (982)	93,63 (1.788)	95,73 (2.797)
		IDiS	93,30 (982)	94,43 (1.788)	96,10 (2.797)
	Luft verteilen	gBERT-base	90,70 (237)	98,89 (403)	99,18 (703)
		IDiS	97,75 (237)	99,19 (403)	99,43 (703)
Kälte erzeugen	gBERT-base	94,08 (293)	98,02 (367)	98,02 (467)	
	IDiS	94,43 (293)	96,60 (367)	98,02 (467)	

Für die verschiedenen Modelle wurden drei Epochen und eine Batch-Size von 8 oder 16 verwendet. Um eine Vergleichbarkeit zwischen dem Originaldatensatz und dem angepassten Datensatz sicherzustellen, wurde der Validierungsdatensatz unverändert beibehalten. Die Ergebnisse der Trainingsdurchläufe bestätigen die Ergebnisse der vorherigen Untersuchungen. Um einen F1-Score von 95% zu erreichen, werden im Originaldatensatz und angepassten Datensatz ungefähr die gleiche Anzahl an Trainingsdaten benötigt. Bei Betrachtung der Ergebnisse der verschiedenen Klassifizierungsstufen fällt auf, dass eine ähnliche Anzahl an Trainingsbeispielen notwendig sind, um einen Score von 95% zu er-

zielen. Des Weiteren belegen die Daten, dass der Gebrauch von Normen zur Erweiterung des Pretrainings (*IDiS*-Modell) durchschnittlich schneller zu besseren Ergebnissen führt. Jedoch lässt sich bei zunehmender Anzahl an Trainingsdaten erneut eine Konvergenz der Ergebnisse erkennen. Die Ergebnisse des angepassten Datensatzes zeigen, dass die während des Trainings erzielten guten Resultate nicht auf den unausgeglichene Datensatz zurückzuführen sind. Die folgende Tabelle 7.6 zeigt die Ergebnisse der besten Modelle pro Klassifizierungsaufgabe mit den jeweiligen Trainingsparametern.

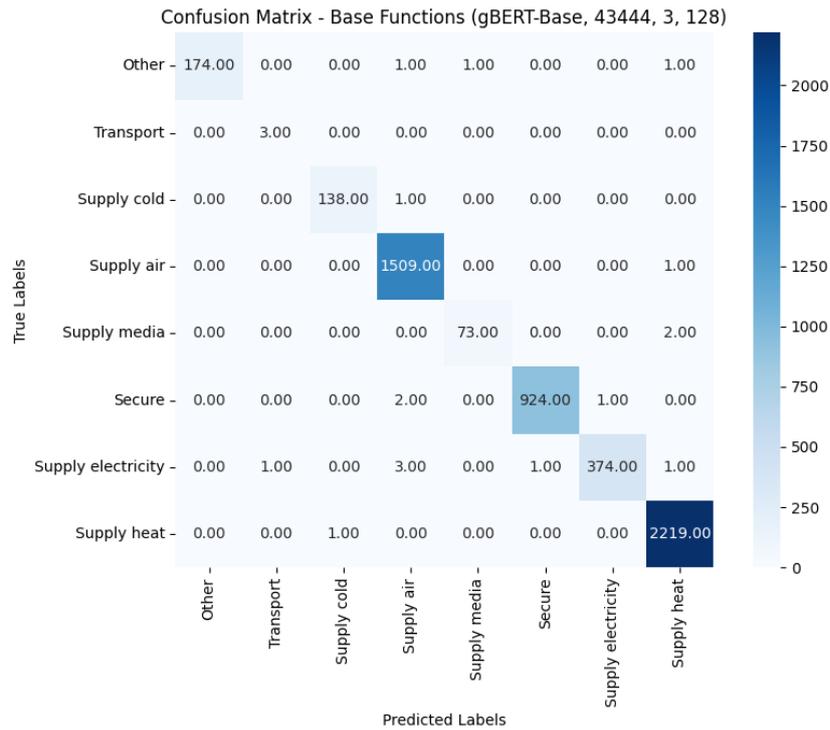
Tabelle 7.6: Übersicht über die besten Ergebnisse (Validierungsdatensatz) mit verwendeten Hyperparametern der verschiedenen Klassifizierungsstufen

Ebene Klassifikation	Datensatz	Basismodell	Anzahl Beispiele	Epochen	Batch- Size	F1- Score in %
Grund- funktion	Grund- funktion	gBERT-base	43.300	3	128	99,69
		IDiS	43.300	3	64	99,63
Unter- funktion	Wärme versorgen	gBERT-base	18.032	3	32	99,60
		IDiS	18.032	3	64	99,65
	Luft versorgen	gBERT-base	11.879	3	64	99,39
		IDiS	11.879	3	32	99,80
	Medien versorgen	gBERT-base	588	3	8	95,88
		IDiS	588	3	8	93,13
	Kälte versorgen	gBERT-base	1.055	5	16	98,48
		IDiS	1.055	3	64	97,72
	Sichern	gBERT-base	5.000	3	32	99,32
		IDiS	5.000	3	64	99,82
Entität	Wärme erzeugen	gBERT-base	1.718	3	32	99,53
		IDiS	1.718	3	32	99,53
	Wärme verteilen	gBERT-base	15.806	3	32	99,15
		IDiS	15.806	3	32	99,03
	Luft bereitstellen	gBERT-base	9.065	3	32	98,84
		IDiS	9.065	3	64	98,58
	Luft verteilen	gBERT-base	2.814	3	64	99,70
		IDiS	2.814	3	8	99,70
	Kälte erzeugen	gBERT-base	650	3	64	98,54
		IDiS	650	3	64	98,54
Datenpunkt	Datenpunkt	gBERT-base	30.966	2	128	99,55
		IDiS	30.966	2	128	99,53

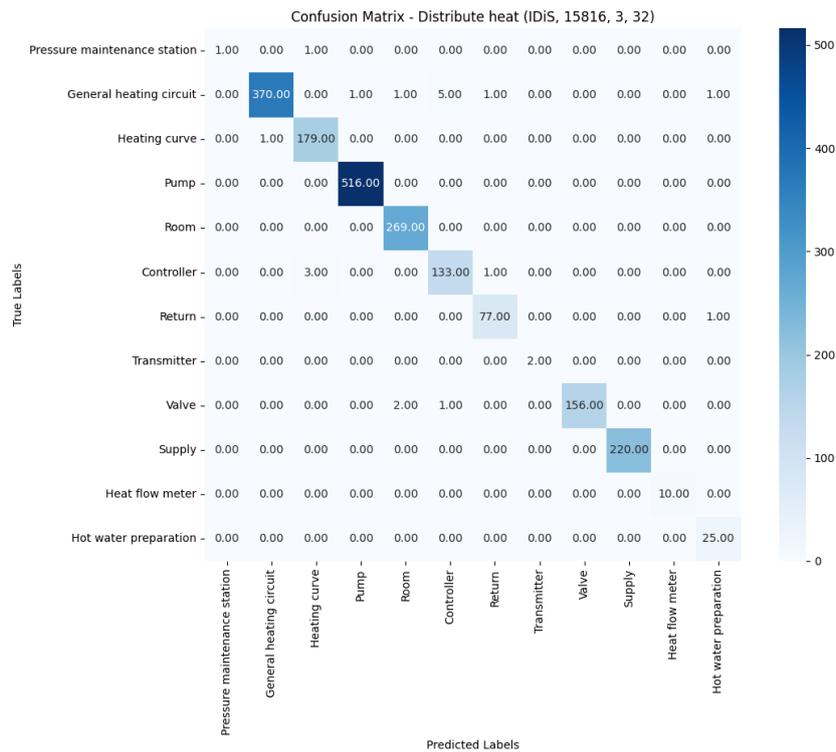
Die Ergebnisse zeigen, dass die ausgewählten Trainingsmethoden sehr gute Resultate liefern.

Für jede Klasse konnte mindestens ein Modell trainiert werden, welches einen F1-Score von über 95% erreicht. Mit Ausnahme der Klasse mit *Medien versorgen*, erreichen alle Klassen einen Wert von über 98%. Zusätzlich zur zuvor untersuchten Anzahl notwendiger Trainingsbeispiele wurden in den Experimenten auch die Anzahl der Epochen und die Größe der Batch-Size untersucht. Die Studien zeigen, dass nur in einem Fall drei Epochen schlechtere Ergebnisse liefern als fünf Epochen. Da die Ergebnisse auf dem Validierungsdatensatz ab einer Anzahl von drei Epochen schlechter werden, kann festgestellt werden, dass ab diesem Zeitpunkt eine Überanpassung der Modelle auf den Trainingsdatensatz stattfindet. Bei der Auswahl der Batch-Size lässt sich in den Ergebnissen der Trend erkennen, dass je größer die Anzahl der Trainingsbeispiele desto größer kann die Batch-Size gewählt werden, um gute Ergebnisse zu erzielen. Beim Ergebnisvergleich der *gBERT*- und *IDiS*-Modell lässt sich kein großer Unterschied bei der Vorhersagegenauigkeit der Modelle feststellen. Es gibt Kategorien, in denen die *gBERT*-Modelle ein etwas besseres Ergebnis erzielen (z.B. Grundfunktion oder *Wärme verteilen*) und andere, in denen die *IDiS*-Modelle etwas besser abschneiden (z.B. mit *Luft versorgen* oder *Sichern*). Tabelle 7.3 zeigt, dass die *IDiS*-Modelle bei einer geringeren Anzahl an Trainingsbeispielen bessere Ergebnisse als die *gBERT*-Modelle liefern. Allerdings reduziert sich dieser Effekt oder verschwindet ganz, wenn die Anzahl der Trainingsbeispiele zunimmt. Somit ist ein erweitertes Pretraining auf domänenspezifischen Texten von Vorteil, wenn nur begrenzte Daten für das Finetuning verfügbar sind, ansonsten jedoch nicht erforderlich.

Um die Verteilung von korrekt und fehlerhaft klassifizierten Daten zu visualisieren, wurde für jedes Modell eine *Wahrheitsmatrix* erstellt. Die Beispiele der Ergebnisse für die Stufe *Grundfunktion* (*gBERT*) und *Wärme verteilen* (*IDiS*) werden als Beispiele in den Abbildungen 7.9a und 7.9b gezeigt. Die Matrizen der anderen Modelle können dem Anhang E entnommen werden.



(a) Wahrheitsmatrix der Stufe Grundfunktion (gBERT)



(b) Wahrheitsmatrix der Stufe Wärme verteilen (IDiS)

Abbildung 7.9: Wahrheitsmatrizen mit korrekt und inkorrekt klassifizierten Datenpunkten

Die guten Ergebnisse werden durch die zwei beispielhaften Matrizen belegt. Fast alle Daten im Validierungssatz wurden ihren jeweiligen Klassen entsprechend zugeordnet. Nur

gelegentlich kam es zu falschen Klassifizierungen von Datenpunkten. Die trainierten Modelle bilden die Grundlage für semantische Interoperabilität und erleichtern die automatisierte Interaktion zwischen AAS und TMon-Anwendungen.

## 7.4 Digitale Zwillinge technischer Systeme als Grundlage für ein technisches Monitoring

Auf Basis des NLP-Algorithmus sollen Strukturinformationen eines Gebäudes generiert werden. Die folgenden Kapitel beschreiben die Generierung eines NLP-Teilmodells, die darauf basierende Ableitung der Strukturinformationen, sowie die Teilmodelle, welche Betriebsinformationen und Informationen des TMon enthalten.

### 7.4.1 Teilmodell NLP Classification Result

Die trainierten NLP-Modelle werden in eine NLP-Pipeline integriert, um im laufenden Betrieb Daten aus dem *BACnet-Datapoints-Information*-Teilmodell zu klassifizieren. Die Pipeline gibt bei diesem Prozess das Teilmodell *NLP-Classification-Result* aus, welches die Ergebnisse der Klassifizierung enthält (vgl. Abbildung 7.10).

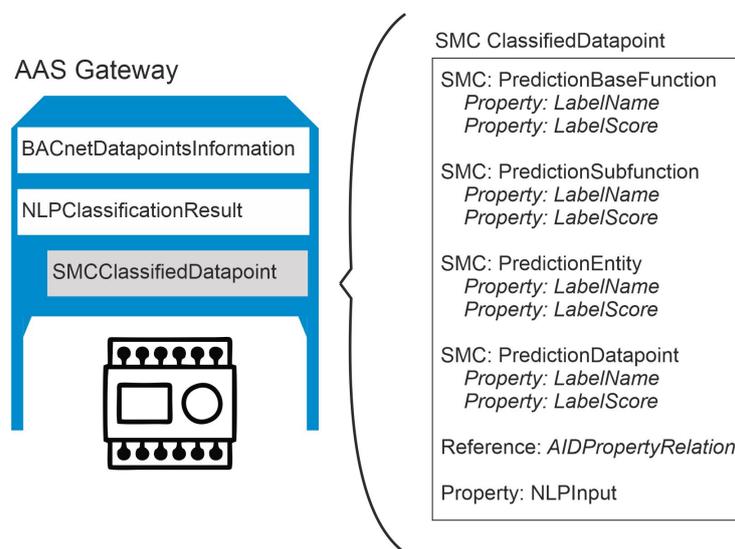


Abbildung 7.10: Visualisierung des Teilmodells NLP Classification Result

Für jeden Datenpunkt des Teilmodells *BACnet-Datapoints-Information* wird im Teilmodell *NLP-Classification-Result* eine entsprechende *SMC Classified-Datapoint* erstellt. Die *SMC* verweist über die Referenz *AID-Property-Relation* auf die zugehörige Sammlung im *BACnet*-Teilmodell. Jeder *SMC Classified-Datapoint* sind vier weitere *SMCs* zugeordnet, welche die Ergebnisse der NLP-Klassifizierung enthalten. Jede dieser *SMCs* enthält sowohl den

vorhergesagten *Label-Name* jeder Stufe als auch den vom jeweiligen Modell vorhergesagten *Label-Score*. Außerdem wird der *NLP-Input* als zusätzliche Eigenschaft hinzugefügt.

#### 7.4.2 Generierung Strukturinformationen

Die Ergebnisse ermöglichen die Generierung von AAS, welche die technischen Systeme eines Gebäudes darstellen und Betriebsinformationen sowie -details für das TMon enthalten. In diesem Kapitel wird die grundlegende Struktur der verschiedenen AAS vorgestellt. Der Zusammenhang der AAS wird im Folgenden als Strukturinformation bezeichnet. Gemäß den Anforderungen 1 und 2 in Kapitel 3 müssen Systeme und ihre Komponenten erweiterbar sein und während ihres Lebenszyklus miteinander kompatibel und kombinierbar bleiben. Um diese Anforderungen zu erfüllen, wurde ein Konzept entwickelt, das die Kombination verschiedener AAS ermöglicht (vgl. Abbildung 7.11).

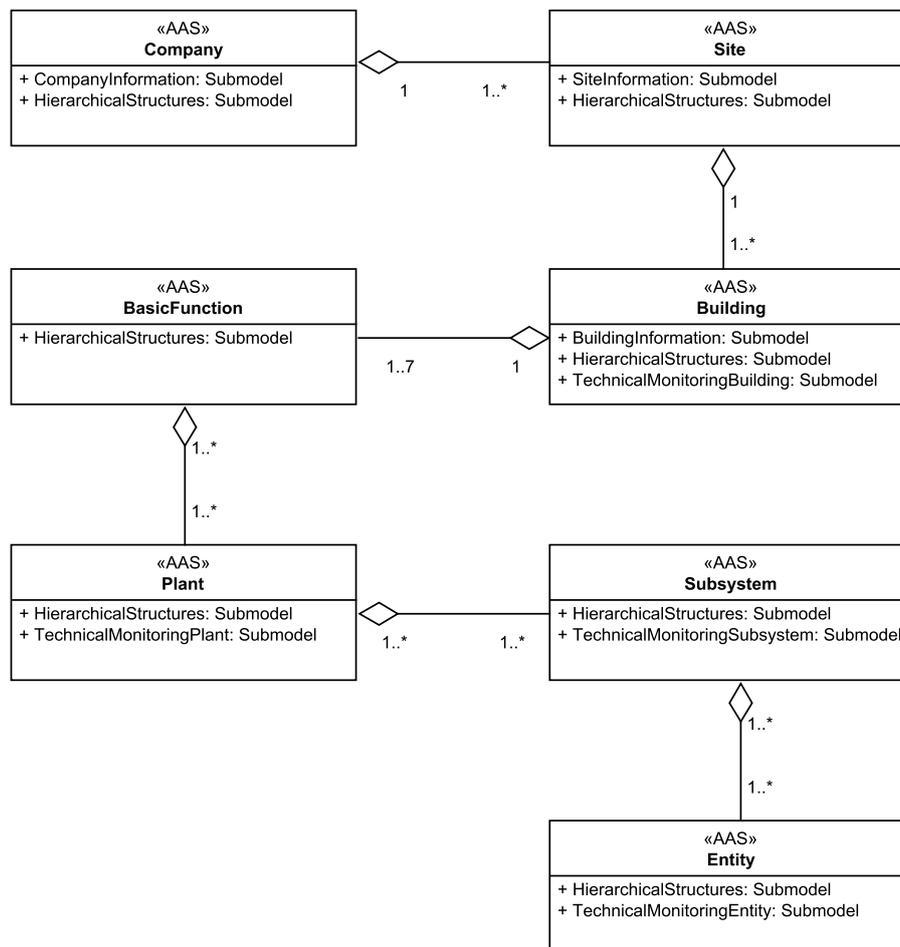


Abbildung 7.11: Überblick über das entwickelte Konzept für Strukturinformationen

Die ersten drei AAS repräsentieren Unternehmen, Standorte und Gebäude. Um die Informationen dieser drei Assets zu strukturieren, wurden drei Teilmodelle (*Company-*

*Information, Site-Information, Building-Information*) entwickelt. Das IDTA-Teilmodell *Hierarchical-Structures* [157] kann verwendet werden, um hierarchische Beziehungen von AAS auszudrücken. Mittels dieses Teilmodells kann eine klare Zuordnung der einzelnen Gebäude innerhalb eines Standorts vorgenommen werden, und Standorte einem Unternehmen zugeordnet werden. Diese AAS müssen vom Nutzer der Anwendung erstellt werden. Derzeit ist das *NLP-Classification Result*-Teilmodell ausschließlich der AAS der jeweiligen Automationsstation zugeordnet. Zur Erstellung von Strukturinformationen werden die Teilmodelle aller Stationen des Gebäudes genutzt. Für jede Grundfunktion, die in einem Gebäude gefunden wird, wird zunächst eine AAS erstellt. Wenn *NLP-Classification-Result*-Teilmodelle nur die Grundfunktionen mit *Luft-* und *Wärme versorgen* enthalten, werden für diese Funktionen zwei AAS erstellt. Die AAS der Klassen Anlage (*Plant*), Teilanlage (*Subsystem*), und Entität (*Entity*) werden ebenfalls auf Basis der Klassifizierungsergebnisse erstellt. Wenn ein Gebäude über drei Heizungssysteme verfügt, sind drei AAS des Typs Anlage erforderlich. Die AAS vom Typ Anlage kombiniert mehrere Teilanlagen, die in einem systemspezifischen Zusammenhang stehen. Im Beispiel der Heizungssysteme sind Teilanlagen mögliche Erzeuger oder Verteilkreise. Zum Beispiel können drei Heizkreise und ein Gaskessel in einem System enthalten sein. Jede Teilanlage wiederum setzt sich aus Entitäten zusammen. Dies können physische Komponenten (bspw. Pumpen oder Ventile) oder Entitäten sein, die Anlagenbereiche wie den Vorlauf oder den Rücklauf repräsentieren. Der AAS-Baum und seine abgeleiteten Hierarchien sollen durch den NLP-Algorithmus automatisch generiert werden. Auf Basis des NLP-Algorithmus können Grundfunktionen, die zweite Funktionsebene, Entitäten und Datenpunkte klassifiziert werden. Die Zugehörigkeit der Teilanlagen und übergeordneten Anlagen kann aktuell jedoch noch nicht automatisiert generiert werden. Hierzu müssen weitere Modelle entwickelt werden. Die Betriebsinformationen und Informationen des TMon sind in den Teilmodellen *Operating-Information* und *Technical-Monitoring* der AAS enthalten und werden in den folgenden Kapiteln beschrieben.

### 7.4.3 Teilmodelle Operating Information

Die Teilmodelle *BACnet-Datapoints-Information* und *NLP-Classification-Result* dienen dazu, die Datenpunkte eines Gebäudes innerhalb von AAS bereitzustellen, zu klassifizieren und die Ergebnisse zu repräsentieren. Anschließend erfolgt die Zuordnung dieser Daten zu den Anlagen und Systemen. Das Teilmodell *Operating-Information* gibt an, welche Datenpunkte aus den anderen Teilmodellen in dieses integriert werden. Wenn in einem vorherigen Schritt verschiedene Pumpen identifiziert wurden, wird für jede Pumpe ein Teilmodell vom Typ *Operating-Information-Pump* mit allen relevanten Informationen erstellt. Die Instanz des Teilmodells wird mithilfe der Informationen vervollständigt, die in den *NLP-Classification-Result*-Teilmodellen enthalten sind. Wenn Betriebsstunden,

Schaltbefehl und Betriebsrückmeldung diesem Teilmodell zugeordnet wurden, werden nur diese Daten im Teilmodell *Operating-Information-Pump* instanziiert. In diesem Prozess werden also automatisch mehrere Teilmodelle erstellt, die je nach bereitgestellten Informationen variieren. Die Komponenten und Anlagen verfügen nun über ihre relevanten Betriebsinformationen und können diese für die weitere Verarbeitung bereitstellen.

### 7.4.4 Teilmodelle Technical Monitoring

Die Teilmodelle des TMon wurden gemäß der AMEV-Richtlinie [105] erstellt. Diese Richtlinie legt objektive Prüfparameter für verschiedene technische Systeme und das Gebäude selbst fest, die im Rahmen des TMon ermittelt werden müssen. Zur Bewertung, ob die technischen Anlagen den spezifischen Anforderungen genügen, ist es notwendig, diese Größen während des Gebäudebetriebs zu erfassen. Insgesamt werden Prüfparameter für 18 verschiedene Systeme definiert, darunter Vollklimaanlagen und Heizkreise. Die Prüfparameter der verschiedenen Systemarten wurden auf I4.0-Teilmodelle abgebildet. Die Instanziierung der Teilmodelle erfolgt auf Basis der *Operating-Information*-Teilmodelle. Ein Beispiel ist die Vorlauftemperatur, die als Prüfgröße für Heizkreise definiert wird. Wenn das Teilmodell *Operating-Information-Heating-Circuit* die Vorlauftemperatur des Heizkreises beinhaltet, wird eine Referenz auf den aktuellen Messwert durch das Teilmodell *Technical-Monitoring-Heating-Circuit-AMEV* erstellt, sodass der Wert dort ebenfalls verfügbar ist. In Abbildung 7.12 ist das Teilmodell *Technical-Monitoring-Heating-Circuit-AMEV* dargestellt.

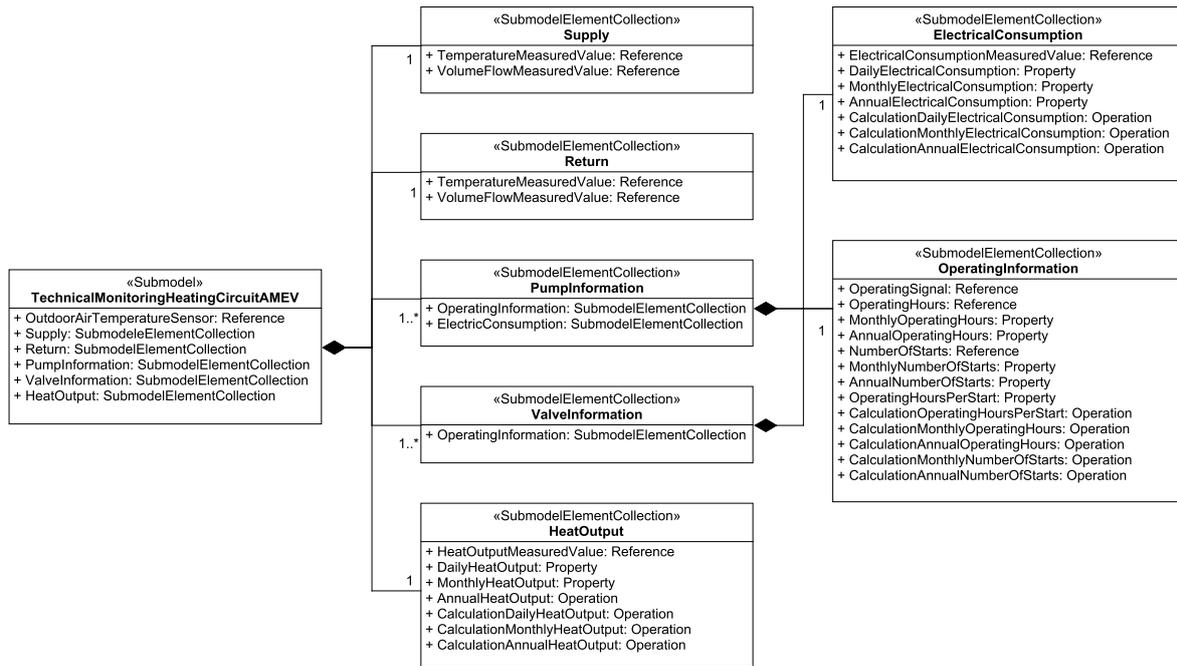


Abbildung 7.12: Teilmodell Technical Monitoring Heating Circuit AMEV

Das Teilmodell besteht aus fünf SMCs und einer Referenz. Wie bereits erläutert, beziehen sich TMon-Teilmodelle auf die aktuellen Werte relevanter Elemente in *Operating-Information*-Teilmodellen. Die SMCs enthalten darüber hinaus zusätzliche Informationen. Die Strukturierung innerhalb von SMCs ermöglicht die Wiederverwendbarkeit über verschiedene Teilmodelle hinweg. Die SMC *Pump-Information* und SMC *Valve-Information* nutzen beide die SMC *Operating-Information*, die Daten wie die Anzahl der Starts enthält. Da diese Werte optional sind, müssen sie nicht unbedingt in der Teilmodellinstanz enthalten sein und könnten bspw. für das Ventil nur teilweise angegeben werden. Es wird erneut auf die aktuellen Werte zurückgegriffen, die Berechnungen werden jedoch innerhalb der entsprechenden SMC im Monitoring-Teilmodell durchgeführt, z.B. *Calculation-Annual-Operating-Hours*. Diese werden als SEs vom Typ *Operation* dargestellt und verfügen über klar definierte Ein- und Ausgänge (in der Abbildung nicht dargestellt). Die verschiedenen TMon-Teilmodelle folgen in ihrem Design diesem Muster.

## 7.5 Fazit Bestandsanlagen

In diesem Kapitel wurde eine Methode vorgestellt, um bestehende TGA-Anlagen durch AAS zu repräsentieren und ihre heterogenen Datenpunktbeschreibungen mittels NLP zu klassifizieren. Dazu wurde ein vierstufiger Klassifikationsprozess entwickelt, der es ermöglicht, die Datenpunkte den grundlegenden Funktionen der TGA, Entitäten und der konkreten Funktion des Datenpunktes zuzuordnen. Basierend auf den klassifizierten Daten-

punkten wurde ein Konzept erstellt, welches die Generierung von Strukturinformationen für Gebäude ermöglicht. Diese Informationen können anschließend von TMon-Anwendungen erkundet werden. In diesem Kapitel wurde die in Kapitel 4.5 herausgearbeitete Herausforderung „Ausstattung bestehender technischer Systeme mit digitalen Zwillingen“ adressiert. Aus Kapitel 3 wurden die Anforderungen 1-10 erfüllt. Hierbei wurde ein Konzept entwickelt, um syntaktische Interoperabilität mittels AAS zu erreichen. Existierende Konzepte aus dem I4.0-Bereich wurden verwendet, um Bestandsanlagen mit digitalen Zwillingen auszustatten, wodurch die Anforderungen 1-7 erfüllt sind. Um die Anforderungen 8-10 zu erfüllen, wurden Teilmodelle und der vierstufige Klassifizierungsalgorithmus entwickelt, welcher die heterogenen Datenpunktbeschreibungen auf die Teilmodelle abbildet. Die elfte Anforderung (Erkundungsmechanismen) wird im folgenden Kapitel adressiert, in dem die prototypische Implementierung einer TMon-Anwendung umgesetzt wird.

## 8 Prototypische Realisierung im Anwendungsfall Technisches Monitoring

In diesem Kapitel wird eine prototypische Anwendung beschrieben, welche die Ergebnisse aus Kapitel 7 in einer Software-Umgebung implementiert (vgl. Abbildung 8.1).

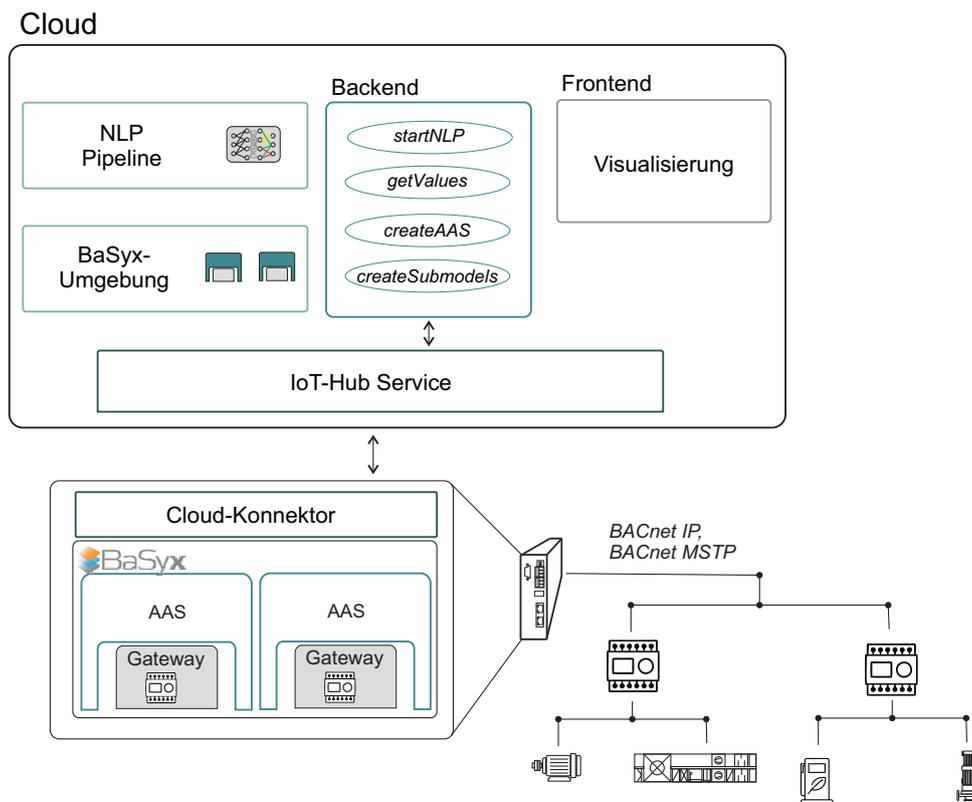


Abbildung 8.1: Visuelle Darstellung der Softwarearchitektur einschließlich einzelner Komponenten

Die Daten der Feldebene werden von den BACnet-Automationsstationen erfasst und an ein Gateway übertragen, um die AAS der Stationen zu generieren (Kapitel 8.1). Anschließend werden die Informationen der AAS zur Weiterverarbeitung an eine *Cloud* übermittelt und über das *Backend* an verschiedene Services vermittelt, bzw. zur Visualisierung für das *Frontend* bereitgestellt. Zunächst wird über das Backend die NLP-Pipeline gestartet (Kapitel 8.2). AAS werden angelegt und ihre Informationen für den Zugriff aus dem Frontend bereitgestellt, sodass diese von Nutzern abgefragt und visualisiert werden können (Kapitel 8.3). Die Implementierung des Backend war nicht Teil dieser Arbeit und wird deshalb in den folgenden Abschnitten nicht näher beschrieben.

## 8.1 Generierung von Verwaltungsschalen für Automationsstationen

Jedes BACnet-Gerät innerhalb eines GA-Netzwerks wird durch eine eigene AAS repräsentiert. Die Erstellung dieser AAS basiert auf den verpflichtenden Netzwerk-Erkundungsmechanismen des BACnet-Protokolls („Who-Is/I-Am“). Eine Client-Anwendung sendet eine „Who-Is“-Anfrage in ein IP-Netzwerk und erhält von vorhandenen BACnet-Geräten eine „I am“-Nachricht, in der alle benötigten Informationen vorhanden sind, um Daten aus einem BACnet-Gerät auszulesen. Diese sind:

1. IP-Adresse des BACnet-Gerätes
2. Objekt-Typ eines Datenpunkts (bspw.: Analog-Input, Analog-Output,...)
3. Instanznummer des BACnet-Objekts
4. Name des auszulesenden Property

Die „I am“-Nachrichten aller BACnet-Geräte werden von der Client-Anwendung genutzt, um auf einem BaSyx-AAS-Server für jedes BACnet-Gerät eine entsprechende AAS anzulegen. Mit den vorhandenen Informationen über das BACnet-Gerät kann für jede AAS das *AID*-Teilmodell angelegt werden, welches die Kommunikationsschnittstelle eines physischen Assets beschreibt. Die Client-Anwendung legt für jede AAS das Teilmodell *BACnet-Datapoints-Information* an. Als drittes Teilmodell wird die AIMC erstellt. In diesem Teilmodell wird beschrieben, welche Merkmale einer Asset-Schnittstelle (beschrieben durch das Teilmodell *AID*) an welches SE in einer AAS übertragen werden sollen. Dieses Teilmodell entspricht in diesem Szenario dem Mapping zwischen den Teilmodellen *AID* und *BACnet-Datapoints-Information*. Mit Hilfe der Erkundungsmechanismen des BACnet-Kommunikationsprotokolls und der BaSyx-Komponenten kann eine vollautomatisierte Erstellung von AAS umgesetzt werden, welche die Informationen aus einem GA-Netzwerk bereitstellen.

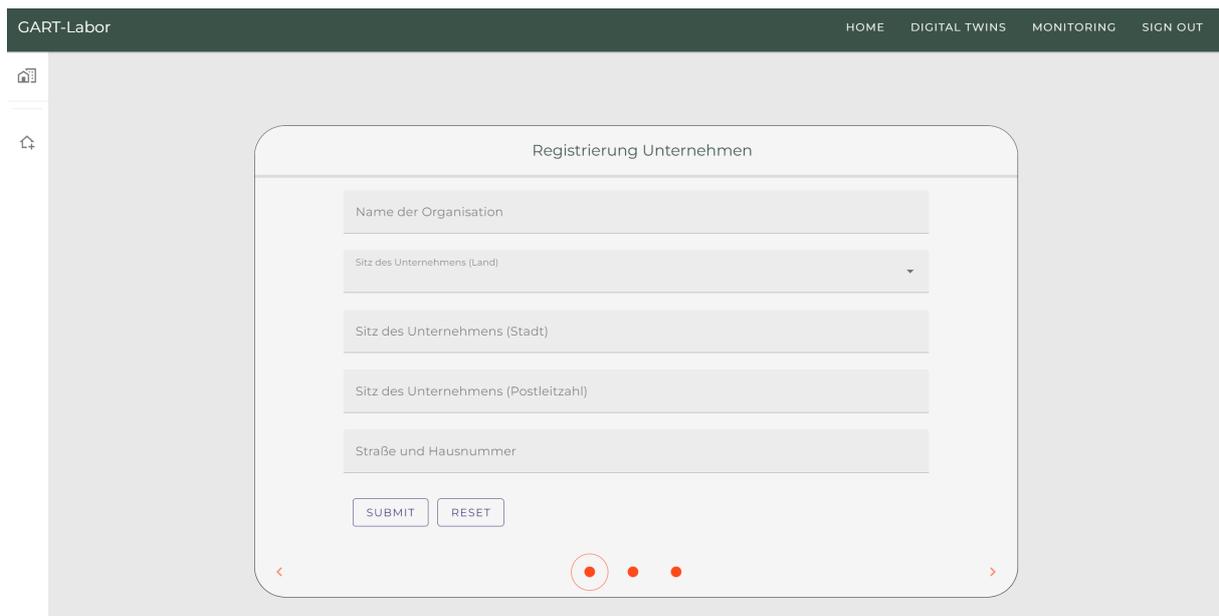
## 8.2 Verarbeitung heterogener Datenpunkte

Das Teilmodell *BACnet-Datapoints-Information* wird in die NLP-Pipeline eingespeist. Die einzelnen SEs, als Repräsentanten der BACnet-Properties, des Teilmodells werden von der Pipeline verarbeitet. Hierbei durchläuft jedes SE nacheinander die vier Stufen des Klassifikationsalgorithmus. Die NLP-Modelle sind auf einem separaten Server verfügbar und werden von diesem abgefragt. Basierend auf den Ergebnissen jeder Stufe wird das passende Modell der nächsten Stufe aufgerufen. Wenn ein SE in der ersten Stufe als Grundfunktion mit *Luft versorgen* klassifiziert wird, wird im zweiten Schritt das Modell

mit *Luft versorgen* vom Server abgerufen und das SE in der zweiten Ebene klassifiziert. Die Ergebnisse jedes SE werden im Zwischenspeicher abgelegt. Sobald alle SEs die Pipeline durchlaufen haben, wird das Teilmodell *NLP-Classification-Result* erstellt. Das Teilmodell wird aus den Ergebnissen im Zwischenspeicher gemäß den Beschreibungen aus Kapitel 7.4.1 erstellt. Anschließend wird das erstellte Teilmodell von der Pipeline zurückgegeben und der korrekten AAS zugeordnet, um es abrufen zu können.

## 8.3 Technische Monitoring Anwendung

Damit die Strukturinformationen eines Gebäudes korrekt zugeordnet werden können, müssen zunächst vom Anwender Angaben über das Unternehmen, die Liegenschaften und die Gebäude der Liegenschaften gemacht werden. Auf Basis dieser Informationen werden die AAS gemäß den Beschreibungen aus Kapitel 7.4.2 gebildet und Strukturinformationen basierend auf dem NLP-Teilmodell erstellt. Der Prozess der Registrierung von Unternehmen etc. erfolgt über Nutzerangaben innerhalb des Frontends (vgl. Abbildungen 8.2 und 8.3).



The screenshot shows a web application interface for 'CART-Labor'. The top navigation bar includes 'HOME', 'DIGITAL TWINS', 'MONITORING', and 'SIGN OUT'. The main content area displays a form titled 'Registrierung Unternehmen'. The form contains the following input fields: 'Name der Organisation', 'Sitz des Unternehmens (Land)' (a dropdown menu), 'Sitz des Unternehmens (Stadt)', 'Sitz des Unternehmens (Postleitzahl)', and 'Straße und Hausnummer'. Below the fields are 'SUBMIT' and 'RESET' buttons. At the bottom of the form, there are three red dots, with the first one circled in red, indicating the current step in a multi-step process. The interface also features a home icon and a hamburger menu icon on the left side.

Abbildung 8.2: Eingabemaske Registrierung Unternehmen

Abbildung 8.3: Visualisierung registrierter Liegenschaften und Gebäude

In den beiden Abbildungen sind beispielhaft die Eingabemaske des Nutzers zu Unternehmensangaben und die ausgefüllten Angaben zu Gebäuden dargestellt. Für die Registrierung der Liegenschaften und Gebäude werden ähnliche Masken wie die zur Unternehmensregistrierung bereitgestellt. Neue Gebäude und Liegenschaften können jederzeit hinzugefügt oder editiert werden. Im Backend werden auf Basis der Angaben des Nutzers AAS generiert (vgl. Abbildung 8.4).

Abbildung 8.4: AAS des Unternehmens, Liegenschaften und Gebäude im Backend

Den AAS werden verschiedene Teilmodelle hinzugefügt. Die in der Abbildung dargestellte AAS des Unternehmens erhält die Teilmodelle *Company-Information*, in dem Informationen über das Unternehmen abgespeichert sind, und das Teilmodell *Hierarchical-Structures*. Über die Angaben des Frontends werden der AAS des Unternehmens die AAS der Liegenschaften als Kinder zugeordnet. Diesen werden wiederum die passenden Gebäude als Kinder zugeordnet. Die Visualisierung der AAS ist in folgender Abbildung 8.5 dargestellt.

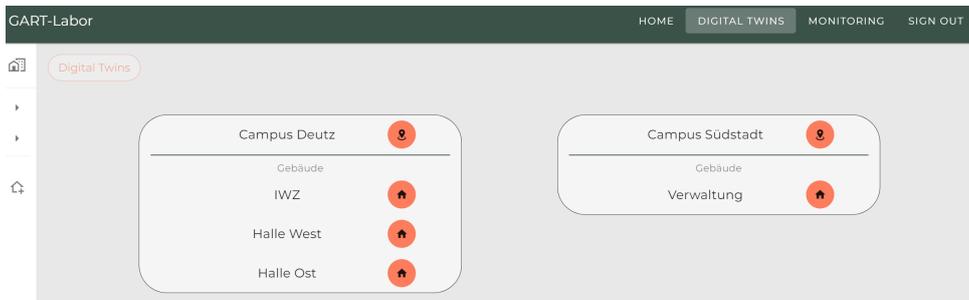


Abbildung 8.5: AAS der Liegenschaften und Gebäude im Frontend

Diese Übersicht ist der Einstiegspunkt, um die erstellten digitalen Zwillinge der Anlagen zu erkunden. Auf Basis der NLP-Teilmodelle eines Gebäudes wurden die Strukturinformationen generiert, sodass nun über das Frontend durch die verschiedenen AAS navigiert werden kann (vgl. Abbildungen 8.6 und 8.7).

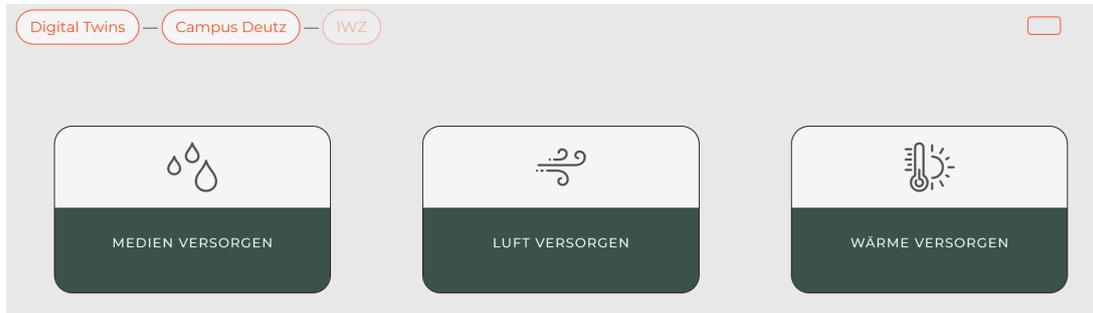


Abbildung 8.6: Visualisierung der erkannten Grundfunktionen eines Gebäudes

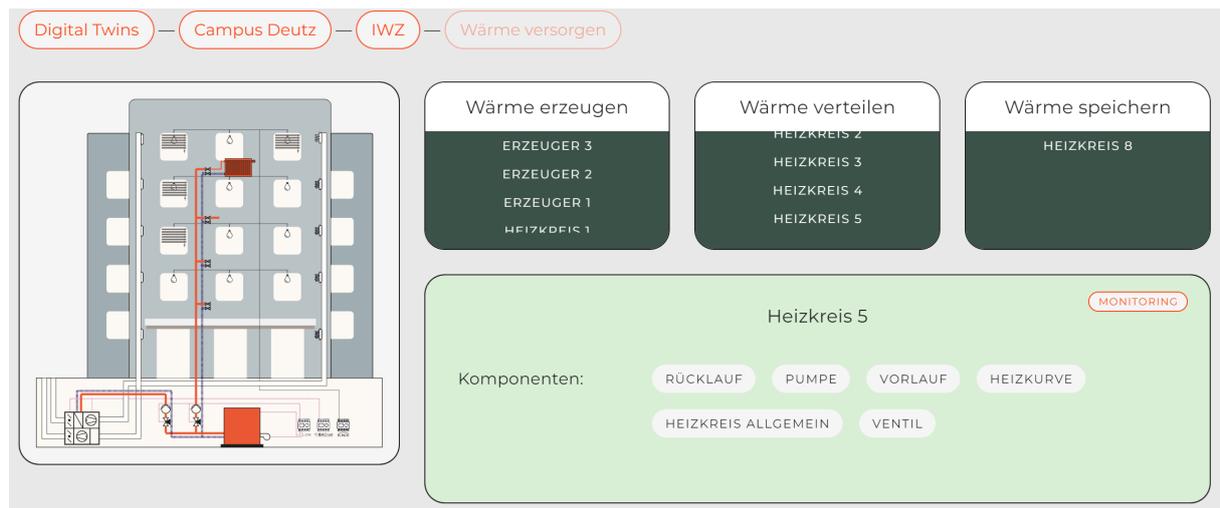


Abbildung 8.7: Visualisierung eines Heizkreises mit zugehörigen Komponenten

Innerhalb des Gebäudes IWZ wurden auf Basis der NLP-Teilmodelle der BACnet-Geräte die drei Grundfunktionen mit *Medien-*, *Luft-* und *Wärme versorgen* erkannt (vgl. Abbildung 8.6). Für diese wurde dementsprechend jeweils eine eigene AAS generiert, welche der AAS des IWZ zugeordnet sind. Innerhalb dieser AAS sind wiederum die gefundenen zweiten Funktionen als eigenständige AAS zugeordnet. Beispielfhaft dargestellt sind die Unterfunktionen der Grundfunktion mit *Wärme versorgen* (vgl. Abbildung 8.7). Hier wurden die drei Funktionen *Wärme erzeugen*, *-verteilen* und *-speichern* erkannt. Den jeweiligen Funktionen sind dann die dazugehörigen Anlagen zugeordnet. Dies können Erzeuger oder Heizkreise sein. In Abbildung 8.8 ist eine Komponente mit ihren Datenpunkten dargestellt.

Name	Object Name	Description	Edit
Rückmeldung Betrieb	21062HZG404PU01BM01	Betriebsmeldung Pumpe Heizkreis Fußbodenheizung Bauteil D	
Rückmeldung Betriebsstunden	21062HZG404PU01BS01	Betriebsstunden Pumpe Heizkreis Fußbodenheizung Bauteil D	
Rückmeldung Handschaltung	21062HZG404PU01RM01	Handschaltung Schaltbefehl Pumpe Heizkreis Fußbodenheizung Bauteil D #SO#	
Schaltbefehl	21062HZG404PU01SB01	Schaltbefehl Pumpe Heizkreis Fußbodenheizung Bauteil D	

(a) Visualisierung der Komponente Pumpe und dazugehöriger Datenpunkte

(b) Editier-Möglichkeit bei falscher Zuordnung

Abbildung 8.8: Beispielhafte AAS einer Komponente mit Datenpunkten

Die Abbildung 8.8a zeigt die AAS einer Pumpe mit den klassifizierten Datenpunkten. In der Tabelle der Datenpunkte sind jeweils der klassifizierte Name, sowie der Name und die

Beschreibung des BACnet-Objektes enthalten. Diese wurden als Input-Parameter für den vierstufigen Klassifizierungsprozess verwendet, auf dessen Basis die Strukturinformationen gebildet wurden. In Kapitel 7.3.3 wurden die Vorhersagegenauigkeiten der verschiedenen Modelle beschrieben. Auch wenn ein F1-Score von 98% über alle Modelle hinweg erreicht werden konnte, können bei den einzelnen Klassifizierungen Fehler auftreten. Aus diesem Grund wird dem Nutzer die Möglichkeit eingeräumt, die vorhergesagten Label zu korrigieren (vgl. Abbildung 8.8b). Jeder Datenpunkt kann über eine Maske editiert werden. Hierbei kann die jeweilige Stufe auf der der Fehler aufgetreten ist, korrigiert werden. Entweder kann eine händische Korrektur aller Ebenen vorgenommen werden, oder die NLP-Pipeline wird durch die händische Korrektur auf einer bestimmten Ebene neu gestartet. Der Aufbau der verschiedenen Entitäten-AAS im Backend ist in Abbildung 8.9 dargestellt.

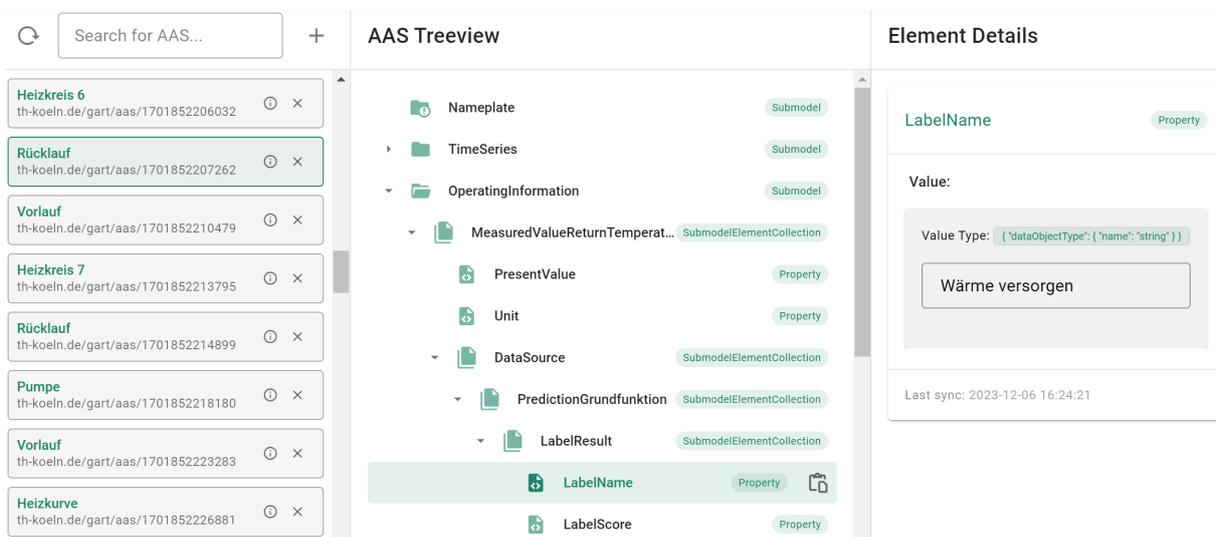


Abbildung 8.9: AAS einer Komponente mit dazugehörigen Teilmodellen im Backend

Die AAS eines Rücklaufs enthält vier Teilmodelle, wobei das *Hierarchical-Structures*-Teilmodell in der Abbildung nicht dargestellt ist. Das Teilmodell *Nameplate* ist aktuell ein Platzhalter für Informationen über verbaute Komponenten. Im Teilmodell *Operating-Information* werden die Informationen gemäß Abschnitt 7.4.3 strukturiert. Wird die AAS der Komponente des Rücklaufs erstellt, wird ein leeres Teilmodell vom Typ *Operating-Information* hinzugefügt. Auf Basis der NLP-Pipeline werden klassifizierte Datenpunkte, welche dem Rücklauf zugeordnet werden können, dem *Operating-Information*-Teilmodell hinzugefügt. In diesem sind dann die aktuellen Werte aus BACnet, die Unit, sowie Informationen über die Ergebnisse der Klassifizierung enthalten. Diese wurden in der SMC *Data-Source* hinzugefügt, damit im Frontend einfacher auf die Informationen zugegriffen werden kann. Darüber hinaus wurde das Teilmodell *Time-Series* [218] hinzugefügt. In diesem werden Trends der aufgenommenen Werte abgespeichert, sodass auf diese zugegriffen werden kann.

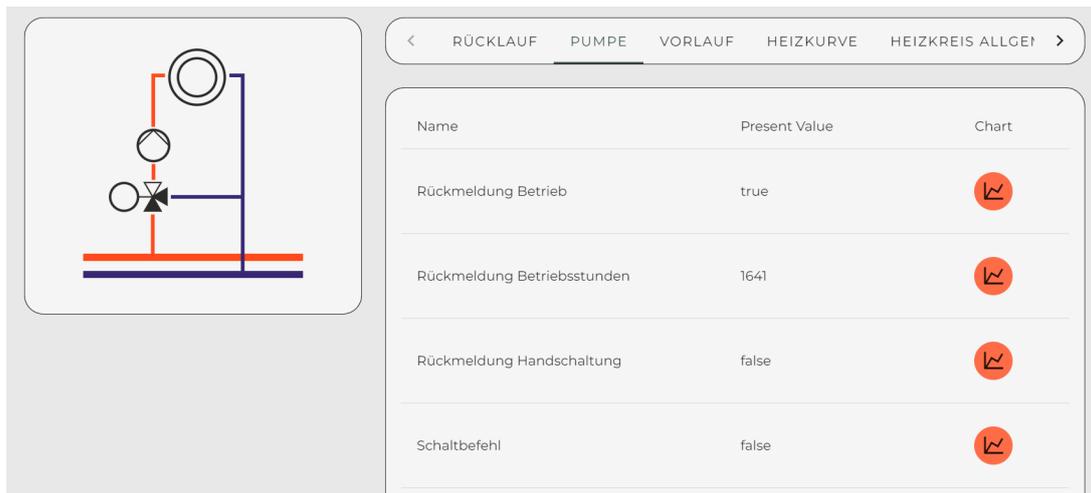
Die bisherigen Funktionalitäten der Anwendung dienen ausschließlich der Visualisierung

und möglichen Editierung der verschiedenen AAS. Hier kann der Nutzer über Anpassungen der Label die Strukturinformationen selbstständig verändern. Das eigentliche TMon wird über den Reiter *Monitoring* der Anwendung erreicht (vgl. Abbildung 8.5). Der Aufbau dieser Seite entspricht grundlegend dem der Seite *Digital Twins*. Erst ab der Ebene der Anlagen-AAS unterscheidet sich die Monitoring-Seite von der der digitalen Zwillinge. In der folgenden Abbildung 8.10 ist die Monitoring-Seite eines Heizkreises dargestellt.

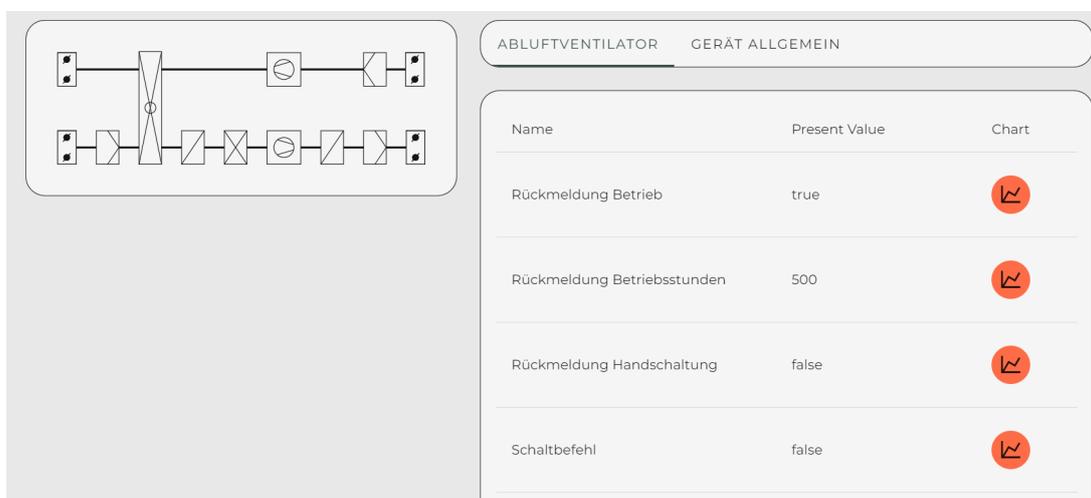


Abbildung 8.10: Trend der Vor- und Rücklauftemperatur eines Heizkreises

Es ist der Verlauf der Vor- und Rücklauftemperatur ersichtlich. Wenn ein Wert nicht gefunden wird (bspw. die Vorlauftemperatur), wird dieser ausgelassen. Über eine Referenz der Werte des Teilmodells *Operating-Information* kann auf die abgespeicherten Verläufe des *Time-Series*-Teilmodells zugegriffen werden. Der Nutzer kann den Zeitraum (Tag, Woche, Monat) einstellen, der visualisiert werden soll. Neben der Übersicht über verschiedene Verläufe, werden die aktuellen Werte der einzelnen Datenpunkte dargestellt (vgl. Abbildung 8.11).



(a) Monitoring eines Heizkreises



(b) Monitoring einer RLT-Anlage

Abbildung 8.11: Visualisierung des Monitorings zweier Anlagen

In Abbildung 8.11a sind alle einem Heizkreis zugeordneten Komponenten dargestellt. Über die Navigation können einzelne Komponenten ausgewählt werden, um die Datenpunkte und deren aktuelle Werte anzuzeigen. Für jeden Datenpunkt wird der aktuelle Wert aus dem *Operating-Information*-Teilmodell abgerufen. Zusätzlich kann der Verlauf des Wertes, auf Basis des *Time-Series*-Teilmodells, visualisiert werden. Die entsprechende Visualisierung für RLT-Anlagen ist in Abbildung 8.11b dargestellt. Die zwei erkundeten Komponenten Abluftventilator und das allgemeine Gerät sind zusammen mit allen gefundenen Datenpunkten dargestellt.

In diesem Kapitel wurde die prototypische Implementierung einer TMon-Anwendung beschrieben. Dabei wurde die in Kapitel 7 beschriebene Methode umgesetzt. Die Anwendung zeigt, wie die entwickelten Ansätze in ein System integriert werden können und eine automatisierte Bereitstellung ermöglicht wird. Abgesehen von den Nutzereingaben

zu Unternehmen, Liegenschaften und Gebäuden wird die Anwendung ohne Konfigurationsaufwand automatisiert generiert. Das aufwändige Datenengineering der Integration heterogen bezeichneter Datenpunkte in eine Anwendung entfällt vollständig. Im folgenden Kapitel 9 werden die Ergebnisse der Arbeit zusammengefasst, der wissenschaftliche Beitrag beschrieben und ein Ausblick darauf gegeben, welche Aspekte zukünftig noch untersucht werden können.

## 9 Zusammenfassung und Ausblick

Das abschließende Kapitel ist in drei Abschnitte gegliedert. Zunächst werden die Ergebnisse der Arbeit zusammengefasst (Kapitel 9.1). Anschließend wird der wissenschaftliche Beitrag der Arbeit beschrieben (Kapitel 9.2). Dabei wird der Beitrag der Arbeit in den Kontext der beiden aktuellen Handlungsfelder der Bundesregierung „Klimaschutz“ und „Digitalisierung“ gestellt, die in Kapitel 1.1 beschrieben wurden und zu denen die Arbeit thematisch gehört. Das abschließende Kapitel 9.3 beschreibt offene Forschungsfragen, die im Rahmen dieser Arbeit nicht beantwortet werden konnten.

### 9.1 Zusammenfassung

In dieser Arbeit wurde ein Konzept entwickelt, das semantische Interoperabilität von digitalen Zwillingen und darauf aufbauend automatisierte Erkundungsmechanismen und Interaktionen ermöglicht. Dabei wurde sowohl der homogene als auch der heterogene Sprachraum berücksichtigt. Für den homogenen Fall wurde ein lebenszyklusorientiertes Vorgehensmodell entwickelt, auf dessen Basis Teilmodelle abgeleitet werden können. Für den Bereich der heterogenen Semantik wurde eine NLP-basierte Methode entwickelt, die heterogene Semantik automatisiert auf einheitliches Vokabular abbildet. Zudem wurde eine Methode erarbeitet, um Bestandsanlagen durch digitale Zwillinge zu repräsentieren und die heterogenen Datenpunktbeschreibungen mittels NLP auf ein einheitliches Vokabular abzubilden. Basierend auf diesem einheitlichen Vokabular werden Strukturinformationen generiert, die von Anwendungen erkundet werden können, sodass diese automatisiert erstellt werden. Dieses Konzept wurde beispielhaft für die TGA umgesetzt und eine TMon-Anwendung erstellt.

In Kapitel 2 wurden die Grundlagen beschrieben, auf denen diese Arbeit aufbaut. Hierbei wurden insbesondere die Interaktionsmechanismen zwischen Menschen sowie die Grundlagen der Linguistik, Computerlinguistik und Interoperabilität erläutert. Aufbauend auf diesen Grundlagen wurde in Kapitel 3 eine Anforderungsanalyse durchgeführt. Hierbei wurde analysiert, welche Anforderungen erfüllt sein müssen, um Interoperabilität zwischen Entitäten zu gewährleisten, damit automatisierte Interaktionen realisiert und Informationen erkundet werden können. Der Fokus der Beschreibungen in Kapitel 4 lag auf der

Darstellung des aktuellen Stands der Technik im Bereich der I4.0 und wie hier Interoperabilität und semantische Interoperabilität aktuell erreicht werden sollen. Zum Abschluss des Kapitels wurden die Ergebnisse der ersten drei Kapitel diskutiert und fünf offene Herausforderungen herausgearbeitet:

1. Abbildung unterschiedlicher digitaler Zwilling Konzepte
2. Einheitliche Modellierung von Assetinformationen
3. Verarbeitung heterogener Semantik
4. Einbindung existierender Anlagen in I4.0-Umgebungen
5. Realisierung autonomer Interaktionen

In dieser Arbeit wurden Konzepte entwickelt, welche die Herausforderungen 2, 3, und 4 adressieren. Die Konzepte sind jeweils in einem eigenständigen Kapitel beschrieben.

Im fünften Kapitel wurde die Problematik der einheitlichen Modellierung von Asset-Informationen behandelt. Damit Teilmodelle von AAS einheitlich definiert und Überschneidungen innerhalb der Teilmodelle vermieden werden, wird ein Modell benötigt, das eine einheitliche Ableitung von Teilmodellen ermöglicht. Dazu wurde in dieser Arbeit ein Vorgehensmodell zur Ableitung von Teilmodellen entworfen, welches auf verschiedene Assets anwendbar ist. Das Modell basiert auf einer prozess- und lebenszyklusorientierten Modellierung von Assetfunktionen, sowie auf einer Klassifikation, mit der Assets innerhalb von Gruppen hierarchisch strukturiert werden können. Als Basis der prozess- und lebenszyklusorientierten Modellierung wurde die Lebenszyklusachse des RAMI 4.0 verwendet. Der dort definierten Unterscheidung zwischen Typ- und Instanz-Phase eines Assets wurden grundlegende Prozesse aus [166] zugeordnet. Diese Prozesse können für verschiedene Assets angewendet werden. In Kombination mit der vorgeschlagenen Klassifikation von Assets nach [170], entstehen Prozesse wie *Systemanforderungen an Kreiselpumpen spezifizieren* oder *Systemanforderungen an Pumpen spezifizieren*. Hiervon können einheitliche Teilmodelle abgeleitet und hierarchisiert werden, sodass Überschneidungen bei der Spezifikation der Teilmodelle vermieden werden können. Auf Basis des Modells wurden 13 Teilmodelle für Pumpen spezifiziert. Die semantischen Beschreibungen wurden in den ECLASS-Standard integriert, sodass die Inhalte der Teilmodelle semantisch eindeutig identifizierbar sind.

Im sechsten Kapitel wurde die Herausforderung der Verarbeitung heterogener Semantik beschrieben. Dass sich alle Teilnehmer globaler, unternehmensübergreifender I4.0-Wertschöpfungsnetzwerke auf einheitliche Teilmodelle einigen, erscheint unwahrscheinlich. Vielmehr wird es weiterhin proprietäre und semantisch heterogen ausgeprägte Teilmodelle geben. Damit Interaktionen zwischen AAS trotzdem automatisiert durchgeführt werden können, wird ein Übersetzungsmechanismus für heterogene Semantik benötigt. In dieser Arbeit

wurde eine Methode entwickelt, die die automatisierte Abbildung heterogener Semantik auf ein Zielvokabular ermöglicht. Hierzu wurden verschiedene Konzepte (Embeddings, KGs, LLMs) aus dem Bereich des NLP verwendet. Name und Definition eines unbekanntem SE werden durch ein LM verarbeitet, SEms gebildet und in Kombination mit Metadaten auf ein Zielvokabular abgebildet. Für die verschiedenen Ansätze wurden Modelle trainiert, ein Datensatz erstellt und die Ergebnisse evaluiert. Die besten Modelle erreichen hierbei einen F1-Score von 90,93%. Die entwickelte Methode wurde in ein I4.0-Interface überführt, welches als Service implementiert werden kann. Dieser wurde in einer prototypischen Implementierung umgesetzt, um die Funktionsweise der Methode zu demonstrieren.

Die Herausforderung der Integration von Bestandsanlagen in I4.0-Umgebungen wurde im siebten Kapitel beschrieben. Hierzu wurde ein allgemeines Konzept entwickelt, das für Anlagen der TGA umgesetzt wurde. Im Folgenden wird direkt die konkrete, TGA-spezifische, Umsetzung zusammengefasst. Jede Automationsstationen eines Gebäudes, das in ein BACnet-Netzwerk integriert ist, wird mit einem digitalen Zwilling in Form einer AAS ausgestattet. Durch eine entwickelte NLP-Methode können die semantisch heterogen vorliegenden Datenpunkte auf einen Standard abgebildet werden. Zur Abbildung wurde ein vierstufiger Algorithmus entwickelt. Zunächst wird jeder Datenpunkt einer Grundfunktion wie mit *Wärme versorgen* zugeordnet, dann einer detaillierteren Funktion wie *Wärme verteilen*, dann einer Entität wie einer *Pumpe* und schließlich der eigentlichen Funktion des Datenpunkts, wie z.B. *Schaltbefehl*. Dieses neue Verfahren ermöglicht die automatisierte Abbildung jedes Datenpunkts und ist die Grundlage für die automatisierte Erstellung von Strukturinformationen eines Gebäudes. Für den NLP-Algorithmus wurde ein Datensatz mit 54.125 gelabelten Datenpunkten erstellt. Die Ergebnisse zeigen, dass sich der entwickelte Ansatz sehr gut dazu eignet, semantisch heterogene Datenpunkte einem Zielvokabular zuzuordnen. Über alle Stufen hinweg wurde ein F1-Score von über 95% erreicht. Die durchgeführten Untersuchungen zeigen, dass durchschnittlich 100 Trainingsbeispiele pro Label erforderlich sind, um einen F1-Score von 95% zu erreichen, und 1.000 Beispiele pro Label für einen F1-Score von 99%. Diese Ergebnisse zeigen die Fähigkeit der Modelle, effektiv aus einer begrenzten Anzahl von Trainingsbeispielen zu lernen und auf unbekannte Daten aus dem Evaluationsdatensatz zu generalisieren. Auf Basis der klassifizierten Daten können Strukturinformationen eines Gebäudes generiert werden. Dazu wurde ein Konzept entwickelt, welches, ausgehend von der AAS eines Unternehmens, hierarchisch AAS bis auf die Ebene technischer Komponenten eines Gebäudes erstellt. Die gewonnene Struktur ermöglicht die automatisierte Generierung von Monitoring-Anwendungen, da nun Erkundungsmechanismen implementiert werden können, welche auf die Informationen der AAS zugreifen, und diese automatisiert integrieren. Im achten Kapitel wurde die prototypische Implementierung einer Monitoring-Anwendung beschrieben, welche die Funktionsweise des entwickelten Konzepts visualisiert.

## 9.2 Wissenschaftlicher Beitrag dieser Arbeit

In Kapitel 2.4.3 wurde ein Überblick über den aktuellen Stand zu den Interoperabilitätssebenen gegeben. Zusammenfassend wurde festgehalten, dass die Transport- und syntaktische Interoperabilität gut entwickelt sind, während die semantische und verhaltensbezogene Interoperabilität noch nicht ausreichend erforscht sind. Dies zeigt sich auch im aktuellen Stand der Technik zur Interoperabilität innerhalb der I4.0 (vgl. Kapitel 4.5). In dieser Arbeit wurde ein gesamtheitlicher Ansatz zur Gewährleistung semantischer Interoperabilität in homogenen sowie heterogenen Fällen vorgestellt, der eine Grundlage für automatisiert ausführbare Erkundungsmechanismen und autonome Interaktionen schafft. Hierfür wurde ein NLP-basiertes Matching entwickelt, das auf Namen und Definitionen basiert und grundsätzlich unabhängig vom verwendeten Konzept des digitalen Zwillinges eingesetzt werden kann. Konkret wurde dieser Ansatz für AAS umgesetzt. Die entwickelte Methode zur automatisierten Abbildung heterogener semantischer Beschreibungen auf ein Zielvokabular erweitert den aktuellen I4.0-Forschungsansatz. Bislang waren ausschließlich ein exaktes Matching (identische Semantic-IDs) oder ein manuelles Matching von Experten möglich (vgl. Kapitel 4.2.3). Allerdings ist die manuelle Vorgehensweise zeitaufwändig und kostenintensiv. Bisher gab es keinen Ansatz zur automatisierten Abbildung heterogener Semantik. Die entwickelte Methode schließt diese Lücke. Der Ansatz zur Verarbeitung heterogener Elemente mittels NLP-Methoden ermöglicht die automatisierte Abbildung und Verarbeitung heterogener Semantik und existiert so bisher noch nicht. Auf Basis der entwickelten Methode können Interaktionen und Erkundungsmechanismen in semantisch heterogenen Wertschöpfungsnetzwerken automatisiert realisiert werden.

Monitoring-Anwendungen sind ein wichtiges Instrument, um den Betriebszustand und die Energieeffizienz von Gebäuden zu bewerten. Durch ihren Einsatz kann der Betrieb optimiert sowie energieeffizienter und kostengünstiger gestaltet werden. Ein energieeffizienter Gebäudebetrieb ist von großer Bedeutung, da dieser für mehr als ein Drittel des deutschen Endenergieverbrauchs verantwortlich ist. Ein optimierter Betrieb ist somit ein wichtiger Baustein für die erfolgreiche Umsetzung der deutschen Klimawende. Aufgrund verschärfter Regularien innerhalb der EU zur Verbesserung der Energieeffizienz von Gebäuden gewinnen Monitoring-Anwendungen zusätzlich an Bedeutung (vgl. 2.5.2). Der breiten Verfügbarkeit solcher Anwendungen, stehen aktuell die hohen Aufwendungen beim Datenengineering im Weg. Die manuelle Analyse der Datenpunkte der Bestandsanlagen verhindert eine automatisierte und kostengünstige Bereitstellung der Anwendungen. Das entwickelte Konzept dieser Arbeit zur Integration von Bestandsanlagen in I4.0-Umgebungen, die automatisierte Klassifizierung von Datenpunkten und die Generierung von Strukturinformationen, adressiert die beschriebene Problematik. Die Ausstattung von Bestandsanlagen mit digitalen Zwillingen wird aktuell schon in Arbeiten rund um die AAS adressiert (vgl. Kapitel 4.2.5).

Bestehende Technologien wurden in dieser Arbeit angewandt und auf die GA übertragen, um Automationsstationen mit BACnet-Kompatibilität in eine I4.0-Umgebung einbinden zu können. Die Klassifikation von heterogenen Datenpunkten wird ebenfalls schon in anderen Arbeiten beschrieben (vgl. Kapitel 4.4). Der Ansatz dieser Arbeit unterscheidet sich von anderen Modellen durch die neu entwickelte vierstufige Klassifizierungsmethode. Außerdem können eine größere Anzahl von Klassen vorhergesagt werden, sowie die Modelle auf unbekannte Datenpunkte angewendet werden. Andere Ansätze beschränken sich üblicherweise auf die Vorhersage entweder der Punkt- oder der Typebene der jeweiligen Datenpunkte. Der alleinige Einsatz dieser Ansätze ermöglicht keine automatisierte Generierung digitaler Zwillinge, sondern erfordert die Kombination mit anderen Modellen. In dieser Arbeit wurde ein vierstufiger Klassifizierungsansatz präsentiert, der das Training mehrerer Modelle in einer Pipeline umfasst, um einen Datenpunkt zu klassifizieren. Dieser Ansatz ermöglicht die automatische Zuordnung von Datenpunkten zu Systemen und Entitäten und vereinfacht somit die Erstellung digitaler Zwillinge. Zudem sind die Modelle in der Lage, präzisere Datenpunktfunktionen zu generieren. In anderen Arbeiten liegt die höchste Anzahl an möglichen Labels in einem Datensatz bei 284 (vgl. Tabelle 4.1). Der innerhalb dieser Arbeit erstellte Datensatz umfasst 467 eindeutige Label. Im Gegensatz zu anderen Methoden, welche Datenpunkte ausschließlich auf Basis des AKS klassifizieren, wurde zusätzlich die optionale BACnet-Beschreibung verwendet. Durch den Einsatz von LM können diese Beschreibungen analysiert und Informationen extrahiert werden. Mit Ausnahme eines Ansatzes werden die Modelle der anderen Ansätze nur an wenigen Gebäuden trainiert und evaluiert. Für diese Arbeit wurden insgesamt 72 Gebäude zum Training und zur Bewertung der Modelle genutzt. Im Evaluationsdatensatz wurde ein durchschnittlicher F1-Score von 98,97% erreicht, welcher höher ist als alle Werte anderer Ansätze.

### 9.3 Ausblick

Als weitere Forschungsaktivitäten können zum einen die in dieser Arbeit entwickelten Methoden weiter ausgearbeitet werden, zum anderen können aufbauend auf den entwickelten Methoden neue Ansätze entwickelt werden.

Um eine ausreichende Zuverlässigkeit für autonome Interaktionen zu erzielen, ist es erforderlich, den erreichten F1-Score von etwa 91% zu erhöhen. In den letzten Jahren haben Entwicklungen im Bereich NLP gezeigt, dass rasch neue Erkenntnisse gewonnen werden, welche zu verbesserten Ergebnissen in verschiedenen Aufgabenbereichen des Forschungsfeldes führen. Wie in Kapitel 6.1.2 beschrieben, steigen die Parameteranzahl der Modelle und die Anzahl der Trainingsdaten stetig an. Es ist daher wichtig zu beobachten und zu evaluieren, wie neue Modelle eingesetzt werden können, um die Vorhersagegenauigkeit zu erhöhen. Weiterführende Untersuchungen sind notwendig, um zu prüfen, wie andere

Methoden, beispielsweise die Integration von KGs in die entwickelte Methode, umgesetzt werden können. In dieser Arbeit durchgeführte Untersuchungen zur Integration von KGs als weitere Wissensquelle haben keine Verbesserung der Genauigkeit erzielt. Hier können weitere Ansätze entwickelt werden, die zu einer Verbesserung führen. Um den entwickelten Ansatz zu evaluieren und ein erstes domänenspezifisches Finetuning durchzuführen, wurde innerhalb dieser Arbeit ein Datensatz erstellt. Dieser besteht aus Merkmalen von Pumpen und RLT-Anlagen sowie Paraphrasen dieser Merkmale. Dies stellt nur eine kleine Auswahl an möglichen Komponenten dar, die in I4.0-Wertschöpfungsnetzwerken verwendet werden können. Der Ansatz wurde so konzipiert, dass die trainierten Modelle auch in Domänen genutzt werden können, welche im Datensatz nicht vorkommen. Die Evaluation der Ergebnisse ergab jedoch, dass die Leistung der Modelle steigt, wenn sie mit Daten trainiert werden, die der Domäne des Anwendungsfalls entsprechen. Folglich muss der Datensatz um weitere technische Domänen ergänzt werden. Darüber hinaus wurden bisher ausschließlich englisch-sprachige Beschreibungen betrachtet und Modelle als Basis verwendet, welche auf englisch-sprachiger Literatur trainiert wurden. Da der I4.0-Sprachraum jedoch nicht auf Englisch beschränkt ist, müssen multilinguale Ansätze entwickelt werden, welche verschiedene Sprachen verarbeiten können. Die entwickelte Methode wurde als I4.0-Interface spezifiziert und als Service umgesetzt. Der Service ermöglicht sowohl ein exaktes Matching, mittels identischer Semantic-IDs zweier SEs, als auch ein NLP-basiertes Matching. Dies kann auf weitere Matchingarten, wie beispielsweise Linked-Data-Methoden oder Äquivalenztabelle, erweitert und abstrahiert werden. Durch die Abstraktion der Matching-Methoden kann ein Interface konzipiert werden, das in konkreten Anwendungsfällen die beste Matching-Methode auswählt. Wenn Äquivalenztabelle für Vokabulare wie ECLASS und IEC CDD vorhanden sind und von SEs, die auf diese Vokabulare verweisen, verwendet werden, ist dieser Ansatz aufgrund seiner höheren Genauigkeit gegenüber einem NLP-basierten Matching vorzuziehen.

Hinsichtlich der Integration von Bestandsanlagen in I4.0-Umgebungen und die automatisierte Klassifikation der Datenpunkte existieren ebenfalls mögliche Ansätze für eine Weiterentwicklung. In das aktuelle Konzept können ausschließlich Automationsstationen eingebunden werden, welche das BACnet-Protokoll verwenden. Dies muss um weitere Protokolle (bspw. KNX, LON) erweitert werden. Zum Trainieren der NLP-Modelle wurden reale Daten aus 72 Gebäuden von zwei Gebäudebetreibern verwendet. Um die Robustheit gegenüber unbekanntem Datenpunktbezeichnungen zu erhöhen, müssen zukünftige Arbeiten den Datensatz um weitere Datenpunkte erweitern. Je heterogener der Datensatz wird, desto besser können die Modelle unbekannte Namen und Beschreibungen verarbeiten. Dazu müssen Datenpunkte von Gebäuden möglichst vieler unterschiedlicher Betreiber genutzt werden, damit die Modelle lernen, unterschiedliche Beschreibungen zu verallgemeinern und zu verarbeiten. Für diese Methode müssen ebenfalls multilinguale Modelle trainiert werden, welche unterschiedliche Sprachen verarbeiten können. Die aktuellen Modelle ba-

sieren auf der Annahme, dass die beiden BACnet-Properties des Objektnamens und der Beschreibung eine textuelle Beschreibung aufweisen, aus der Informationen abgeleitet werden können. Wenn das BACnet-Protokoll in einem Gebäude implementiert ist, aber nur Objektnamen wie AnalogInput1, AnalogInput2 usw. als Datenpunktbezeichnungen verwendet werden, kann das Modell diese Datenpunkte keiner der sieben Grundfunktionen zuordnen, sondern würde sie als *Andere Anlagen* klassifizieren. Hier müssen andere Methoden untersucht werden, bspw. Ansätze zur Verarbeitung von Zeitreihen (vgl. Kapitel 4.4), und die entwickelte Methode um diese erweitert werden. Mittels des vierstufigen Klassifikationsprozesses werden Datenpunkte der Grundfunktion, einer Unterfunktion, einer Entität und der eigentlichen Funktion eines Datenpunkts zugeordnet. Eine Zuordnung, welche Datenpunkte zu welcher Anlage gehören, kann aktuell noch nicht durchgeführt werden. Die Klassifizierung muss um diesen Schritt erweitert werden. Durch die automatisierten Monitoring-Anwendungen soll der Zugang für Gebäudebetreiber:innen zu einfach zu implementierenden Lösungen erleichtert werden. Innerhalb dieser Arbeit wurde eine prototypische Monitoring-Anwendung implementiert, mit der die aktuellen Werte der TGA-Anlagen visualisiert, sowie Verläufe von bspw. Temperaturen in Charts dargestellt werden können. In der Zukunft muss die Anwendung um weitere Funktionalitäten erweitert werden. Die Implementierung der in dieser Arbeit entworfenen TMon-Teilmodelle würde die reine Visualisierung um aussagekräftige Analysen erweitern. Zusätzlich müssen Teilmodelle entwickelt werden, welche auf Basis der generierten Strukturinformationen Key-Performance-Indikatoren erstellen, die den Betrieb der Gebäude bewerten. Grundsätzlich liefern die automatisiert generierten Strukturinformationen die Basis für verschiedene weitere Anwendungen, bspw. ein Energie- oder Alarmmanagement.

## Literatur

- [1] Bundesministerium für Bildung und Forschung, Hrsg. *Zukunftsstrategie Forschung und Innovation: Kurzfassung*. Feb. 2023.
- [2] Bundesministerium für Wirtschaft und Klimaschutz, Hrsg. *Referentenentwurf des Bundesministeriums für Wirtschaft und Klimaschutz - Entwurf eines Zweiten Gesetzes zur Änderung des Bundes-Klimaschutzgesetzes*. Juni 2023.
- [3] Bundesministerium für Wirtschaft und Klimaschutz, Hrsg. *Energieeffizienz in Zahlen: Entwicklungen und Trends in Deutschland 2022*. Feb. 2023.
- [4] Bundesministerium für Bildung und Forschung, Hrsg. *Natürlich. Digital. Nachhaltig. Ein Aktionsplan des BMBF*. März 2020.
- [5] Bundesministerium für Digitales und Verkehr, Hrsg. *Digitalstrategie: Gemeinsam digitale Werte schöpfen*. Apr. 2023.
- [6] Bundesministerium für Bildung und Forschung, Hrsg. *BMBF-Aktionsplan Künstliche Intelligenz: Neue Herausforderungen chancenorientiert angehen*. Nov. 2023.
- [7] Promotorengruppe Kommunikation der Forschungsunion Wirtschaft – Wissenschaft, Hrsg. *Deutschlands Zukunft als Produktionsstandort sichern: Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 - Abschlussbericht des Arbeitskreises Industrie 4.0*. Apr. 2013.
- [8] Bundesministerium für Wirtschaft und Klimaschutz, Hrsg. *Industrie 4.0: Dossier*. abgerufen am: 01.12.2023. URL: <https://www.de.digital/Redaktion/DE/Dossier/industrie-40.html>.
- [9] Bundesministerium für Wirtschaft und Energie, Hrsg. *Leitbild 2030 für Industrie 4.0: Digitale Ökosysteme global gestalten*. Okt. 2019.
- [10] Bundesministerium für Wirtschaft und Energie, Hrsg. *Positionspapier: Interoperabilität – Unsere Vision für Industrie 4.0: Maschinen sprechen in vernetzten digitalen Ökosystemen interoperabel miteinander*. Nov. 2019.
- [11] IDTA, Hrsg. *Specification of the Asset Administration Shell: Part 1: Metamodel*. Apr. 2023.
- [12] IDTA, Hrsg. *Specification of the Asset Administration Shell: Part 2: Application Programming Interfaces*. Apr. 2023.

- 
- [13] VDI, Hrsg. *Sprache für I4.0-Komponenten: Struktur von Nachrichten*. VDI/VDE 2193 Blatt 1. Apr. 2020.
- [14] IDTA, Hrsg. *AAS Submodel Templates*. abgerufen am: 05.12.2023. URL: <https://industrialdigitaltwin.org/content-hub/teilmodelle>.
- [15] Google, Hrsg. *Google Übersetzer*. abgerufen am: 04.12.2023. URL: <https://translate.google.com/>.
- [16] DeepL, Hrsg. *DeepL Übersetzer*. abgerufen am: 04.12.2023. URL: <https://www.deepl.com/translator>.
- [17] Auhagen Ann Elisabeth. *Interaktion*. abgerufen am: 05.12.2023. URL: <https://www.spektrum.de/lexikon/psychologie/interaktion/7296>.
- [18] Rudolf Johannes Wilhelm Bergius. *soziale Interaktion*. abgerufen am: 05.12.2023. URL: <https://dorsch.hogrefe.com/stichwort/soziale-interaktion>.
- [19] Hans-Werner Bierhoff. *Kommunikation*. abgerufen am: 05.12.2023. URL: <https://dorsch.hogrefe.com/stichwort/kommunikation>.
- [20] Roland Posner. „Zur Systematik der Beschreibung verbaler und nonverbaler Kommunikation“. In: *Perspektiven auf Sprache*. Hrsg. von Hans G. Bosshardt. De Gruyter, 1986, S. 267–314. ISBN: 9783110886238. DOI: 10.1515/9783110886238-016.
- [21] C. E. Shannon. „A mathematical theory of communication“. In: *The Bell System Technical Journal* 27.3 (1948), S. 379–423. DOI: 10.1002/j.1538-7305.1948.tb01338.x.
- [22] Paul Watzlawick, Janet Beavin Bavelas und Don D. Jackson. *Menschliche Kommunikation: Formen, Störungen, Paradoxien*. 12., unveränd. Aufl. Programmbereich Psychologie. Bern: Huber, 2011. ISBN: 978-3456849706.
- [23] Axel Schmidt. „Interaktion und Kommunikation“. de. In: Hrsg. von Dagmar Hoffmann und Rainer Winter. *Mediensoziologie. Handbuch für Wissenschaft und Studium*. Baden-Baden: Nomos, 2018, S. 15–38. ISBN: 978-3-8329-7991-1. URL: <https://nbn-resolving.org/urn:nbn:de:bsz:mh39-70735>.
- [24] *Linguistik: Definition*. abgerufen am: 05.12.2023. URL: <https://dictionary.cambridge.org/de/worterbuch/deutsch-englisch/linguistik>.
- [25] Volker Harm. *Einführung in die Lexikologie*. Einführung Germanistik. Darmstadt: WBG, 2015. ISBN: 978-3-534-26384-4.
- [26] Kai-Uwe Carstensen u. a. *Computerlinguistik und Sprachtechnologie: Eine Einführung*. 3. Aufl. 2010. Heidelberg: Spektrum Akademischer Verlag, 2010. ISBN: 978-3-8274-2023-7. URL: <http://nbn-resolving.org/urn:nbn:de:bsz:31-epflicht-1610680>.

- [27] Renate Bartsch und Theo Vennemann. *Grundzüge der Sprachtheorie: Eine linguistische Einführung*. Unveränd. Studienausg. der Aufl. 1982. Bd. 32. Konzepte der Sprach- und Literaturwissenschaft. Tübingen: Niemeyer, 1983. ISBN: 3-484-22032-5.
- [28] Ferdinand de Saussure, Charles Bally und Albert Sechehaye. *Grundfragen der allgemeinen Sprachwissenschaft*. De Gruyter, 2001. ISBN: 978-3-11-017015-3. DOI: 10.1515/9783110870183.
- [29] Hans Peter Althaus, Helmut Henne und Herbert Ernst Wiegand, Hrsg. *Lexikon der Germanistischen Linguistik*. Max Niemeyer Verlag, 1980. ISBN: 9783110960846. DOI: 10.1515/9783110960846.
- [30] Andreas Gardt. *Geschichte der Sprachwissenschaft in Deutschland: Vom Mittelalter bis ins 20. Jahrhundert*. De-Gruyter-Studienbuch. Berlin: De Gruyter, 1999. ISBN: 3-11-015789-6. DOI: 10.1515/9783110803419.
- [31] Noam Chomsky. *Syntactic Structures*. Berlin, Boston: De Gruyter Mouton, 1957. ISBN: 9783112316009. DOI: 10.1515/9783112316009.
- [32] Noam Chomsky. *Aspects of the Theory of Syntax*. 50. Aufl. abgerufen am: 05.12.2023. The MIT Press, 1965. ISBN: 9780262527408. URL: <http://www.jstor.org/stable/j.ctt17kk81z>.
- [33] Dan Jurafsky und James H. Martin. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. 2. ed., Pearson internat. ed. Prentice Hall series in artificial intelligence. Upper Saddle River, NJ: Prentice Hall Pearson Education Internat, 2009. ISBN: 978-0135041963.
- [34] Matthias Janson. *Rechenleistung von Supercomputern wächst rasant*. Hrsg. von statista. abgerufen am: 05.12.2023. URL: <https://de.statista.com/infografik/26868/rechenleistung-des-jeweils-staerksten-supercomputers-der-welt/>.
- [35] Leo Gao u. a. *The Pile: An 800GB Dataset of Diverse Text for Language Modeling*. Dez. 2020. URL: <http://arxiv.org/pdf/2101.00027v1>.
- [36] Ashish Vaswani u. a. *Attention Is All You Need*. Juni 2017. URL: <http://arxiv.org/pdf/1706.03762v5>.
- [37] Alan M. Turing. „Computing Machinery and Intelligence“. In: *Mind* 59.October (1950), S. 433–460. DOI: 10.1093/mind/lix.236.433.
- [38] John McCarthy u. a. „A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence August 31 1955“. In: *AI Magazine* 27.4 (2006), S. 12. DOI: 10.1609/aimag.v27i4.1904. URL: <https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/1904>.

- 
- [39] Pedro Domingos. „A few useful things to know about machine learning“. In: *Communications of the ACM* 55.10 (2012), S. 78–87. ISSN: 0001-0782. DOI: 10.1145/2347736.2347755.
- [40] Ian Goodfellow, Yoshua Bengio und Aaron Courville. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016.
- [41] Yann LeCun, Yoshua Bengio und Geoffrey Hinton. „Deep learning“. In: *Nature* 521.7553 (2015), S. 436–444. DOI: 10.1038/nature14539.
- [42] Yoshua Bengio u. a. „A Neural Probabilistic Language Model“. In: *J. Mach. Learn. Res.* 3.null (2003), S. 1137–1155. ISSN: 1532-4435.
- [43] Tomas Mikolov u. a. *Efficient Estimation of Word Representations in Vector Space*. 2013. URL: <https://arxiv.org/abs/1301.3781v3>.
- [44] Jeffrey Pennington, Richard Socher und Christopher Manning. „GloVe: Global Vectors for Word Representation“. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Hrsg. von Alessandro Moschitti, Bo Pang und Walter Daelemans. Doha, Qatar: Association for Computational Linguistics, Okt. 2014, S. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: <https://aclanthology.org/D14-1162>.
- [45] Matthew E. Peters u. a. *Semi-supervised sequence tagging with bidirectional language models*. 2017. URL: <http://arxiv.org/pdf/1705.00108v1>.
- [46] Matthew E. Peters u. a. „Deep Contextualized Word Representations“. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Hrsg. von Marilyn Walker, Heng Ji und Amanda Stent. New Orleans, Louisiana: Association for Computational Linguistics, Juni 2018, S. 2227–2237. DOI: 10.18653/v1/N18-1202. URL: <https://aclanthology.org/N18-1202>.
- [47] Sepp Hochreiter und Jürgen Schmidhuber. „Long Short-Term Memory“. In: *Neural Computation* 9.8 (1997), S. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.
- [48] Matthew E. Peters, Sebastian Ruder und Noah A. Smith. „To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks“. In: *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*. Hrsg. von Isabelle Augenstein u. a. Florence, Italy: Association for Computational Linguistics, Aug. 2019, S. 7–14. DOI: 10.18653/v1/W19-4302. URL: <https://aclanthology.org/W19-4302>.
- [49] Sinno Jialin Pan und Qiang Yang. „A Survey on Transfer Learning“. In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), S. 1345–1359. ISSN: 1041-4347. DOI: 10.1109/TKDE.2009.191.

- [50] Dzmitry Bahdanau, Kyunghyun Cho und Yoshua Bengio. „Neural Machine Translation by Jointly Learning to Align and Translate“. In: *CoRR* abs/1409.0473 (2014). URL: <https://api.semanticscholar.org/CorpusID:11212020>.
- [51] Ilya Sutskever, Oriol Vinyals und Quoc V. Le. „Sequence to Sequence Learning with Neural Networks“. In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'14. Montreal, Canada: MIT Press, 2014, S. 3104–3112.
- [52] Kyunghyun Cho u. a. „Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation“. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Hrsg. von Alessandro Moschitti, Bo Pang und Walter Daelemans. Doha, Qatar: Association for Computational Linguistics, 2014, S. 1724–1734. DOI: 10.3115/v1/D14-1179. URL: <https://aclanthology.org/D14-1179>.
- [53] Pengcheng He, Jianfeng Gao und Weizhu Chen. *DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing*. Nov. 2021. URL: <http://arxiv.org/pdf/2111.09543v4>.
- [54] Jacob Devlin u. a. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2018. URL: <http://arxiv.org/pdf/1810.04805v2>.
- [55] Yinhan Liu u. a. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. Juli 2019. URL: <http://arxiv.org/pdf/1907.11692v1>.
- [56] Victor Sanh u. a. *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. Okt. 2019. URL: <http://arxiv.org/pdf/1910.01108v4>.
- [57] Alexis Conneau und Guillaume Lample. „Cross-Lingual Language Model Pretraining“. In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.
- [58] Kevin Clark u. a. *ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators*. März 2020. URL: <http://arxiv.org/pdf/2003.10555v1>.
- [59] Zhilin Yang u. a. „XLNet: Generalized Autoregressive Pretraining for Language Understanding“. In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.
- [60] Alec Radford u. a. *Improving language understanding by Generative Pre-Training*. 2018.
- [61] Alec Radford u. a. *Language Models are Unsupervised Multitask Learners*. 2019.

- 
- [62] Li Yang und Abdallah Shami. „On hyperparameter optimization of machine learning algorithms: Theory and practice“. In: *Neurocomputing* 415 (2020), S. 295–316. ISSN: 09252312. DOI: 10.1016/j.neucom.2020.07.061.
- [63] Joel Lamy-Poirier. *Layered gradient accumulation and modular pipeline parallelism: fast and efficient training of large language models*. Juni 2021. URL: <http://arxiv.org/pdf/2106.02679v1>.
- [64] Xue Ying. „An Overview of Overfitting and its Solutions“. In: *Journal of Physics: Conference Series* 1168.2 (2019), S. 022022. ISSN: 1742-6588. DOI: 10.1088/1742-6596/1168/2/022022.
- [65] Alice Zheng. *Evaluating Machine Learning Models*. 1st edition. O’Reilly Media, Inc, 2015. ISBN: 9781491932445. URL: <https://permalink.obvsg.at/>.
- [66] Richard Socher u. a. „Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection“. In: *Advances in Neural Information Processing Systems 24*. 2011.
- [67] Shankar Iyer, Nikhil Dandekar und Kornél Csernai. *First Quora Dataset Release: Question Pairs*. abgerufen am: 13.12.2023. URL: <https://www.quora.com/q/quoradata/First-Quora-Dataset-Release-Question-Pairs>.
- [68] Alfirna Rizqi Lahitani, Adhistya Erna Permanasari und Noor Akhmad Setiawan. „Cosine similarity to determine similarity measure: Study case in online essay assessment“. In: *2016 4th International Conference on Cyber and IT Service Management*. 2016, S. 1–6. DOI: 10.1109/CITSM.2016.7577578.
- [69] Per-Erik Danielsson. „Euclidean distance mapping“. In: *Computer Graphics and Image Processing* 14.3 (1980), S. 227–248. ISSN: 0146664X. DOI: 10.1016/0146-664X(80)90054-4.
- [70] Nils Reimers. *SentenceTransformers Documentation: Losses*. abgerufen am: 20.11.2023. URL: [https://www.sbert.net/docs/package\\_reference/losses.html#losses](https://www.sbert.net/docs/package_reference/losses.html#losses).
- [71] Lisa Ehrlinger und Wolfram Wöß. „Towards a Definition of Knowledge Graphs“. In: *International Conference on Semantic Systems*. 2016. URL: <https://api.semanticscholar.org/CorpusID:8536105>.
- [72] Thomas R. Gruber. „A translation approach to portable ontology specifications“. In: *Knowledge Acquisition* 5.2 (1993), S. 199–220. ISSN: 1042-8143. DOI: 10.1006/knac.1993.1008. URL: <https://www.sciencedirect.com/science/article/pii/S1042814383710083>.

- [73] Bharathan Balaji u. a. „Brick: Towards a Unified Metadata Schema For Buildings“. In: *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*. BuildSys '16. Palo Alto, CA, USA: Association for Computing Machinery, 2016, S. 41–50. ISBN: 9781450342643. DOI: 10.1145/2993422.2993577. URL: <https://doi.org/10.1145/2993422.2993577>.
- [74] Dan Praire u. a. *Project Haystack: A CABA Whitepaper*. Hrsg. von Continental Automated Buildings Association. 2016.
- [75] Marco Pritoni u. a. „Metadata Schemas and Ontologies for Building Energy Applications: A Critical Review and Use Case Analysis“. In: *Energies* 14.7 (2021). ISSN: 1996-1073. DOI: 10.3390/en14072024. URL: <https://www.mdpi.com/1996-1073/14/7/2024>.
- [76] Yuanfei Dai u. a. „A Survey on Knowledge Graph Embedding: Approaches, Applications and Benchmarks“. In: *Electronics* 9.5 (2020). ISSN: 2079-9292. DOI: 10.3390/electronics9050750. URL: <https://www.mdpi.com/2079-9292/9/5/750>.
- [77] Ivana Balazevic, Carl Allen und Timothy Hospedales. „TuckER: Tensor Factorization for Knowledge Graph Completion“. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hrsg. von Kentaro Inui u. a. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, S. 5185–5194. DOI: 10.18653/v1/D19-1522. URL: <https://aclanthology.org/D19-1522>.
- [78] Tianxiang Sun u. a. „CoLAKE: Contextualized Language and Knowledge Embedding“. In: *Proceedings of the 28th International Conference on Computational Linguistics*. Hrsg. von Donia Scott, Nuria Bel und Chengqing Zong. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dez. 2020, S. 3660–3670. DOI: 10.18653/v1/2020.coling-main.327. URL: <https://aclanthology.org/2020.coling-main.327>.
- [79] ISO/IEC, Hrsg. *Information technology — Cloud computing: Part 1: Vocabulary*. ISO/IEC 22123-1. 2023.
- [80] Peter Wegner. „Interoperability“. In: *ACM Computing Surveys* 28.1 (1996), S. 285–287. ISSN: 0360-0300. DOI: 10.1145/234313.234424.
- [81] Christian Diedrich, Arndt Lüder und Lorenz Hundt. „Bedeutung der Interoperabilität bei Entwurf und Nutzung von automatisierten Produktionssystemen“. In: *at - Automatisierungstechnik* 59.7 (2011), S. 426–438. ISSN: 0178-2312. DOI: 10.1524/auto.2011.0937.

- 
- [82] Amit P. Sheth. „Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics“. In: *Interoperating Geographic Information Systems*. Hrsg. von Michael Goodchild u. a. Boston, MA: Springer US, 1999, S. 5–29. ISBN: 978-1-4613-7363-6. DOI: 10.1007/978-1-4615-5189-8{\textunderscore}2.
- [83] ISO/ IEC, Hrsg. *Internet of things (IoT) – Interoperability for iot systems: Part 1: Framework*. ISO/IEC 21823-1. 2019.
- [84] Mahda Noura, Mohammed Atiquzzaman und Martin Gaedke. „Interoperability in Internet of Things: Taxonomies and Open Challenges“. In: *Mobile Networks and Applications* 24.3 (2019), S. 796–809. ISSN: 1383-469X. DOI: 10.1007/s11036-018-1089-9.
- [85] Andreas Tolk und James Muguira. „The Levels of Conceptual Interoperability Model“. In: *2003 Fall Simulation Interoperability Workshop*.
- [86] A. M. Ouksel und A. Sheth. „Semantic interoperability in global information systems“. In: *ACM SIGMOD Record* 28.1 (1999), S. 5–12. ISSN: 0163-5808. DOI: 10.1145/309844.309849.
- [87] Sandra Heiler. „Semantic interoperability“. In: *ACM Computing Surveys* 27.2 (1995), S. 271–273. ISSN: 0360-0300. DOI: 10.1145/210376.210392.
- [88] ISO/ IEC, Hrsg. *Information technology - Cloud computing - Interoperability and portability: Part 1: Framework*. ISO/IEC 19941. 2017.
- [89] International Electrotechnical Commission, Hrsg. *Semantic interoperability: challenges in the digital transformation age*. Schweiz, 2019.
- [90] ISO/ IEC, Hrsg. *Internet of Things (IoT) - Interoperability for IoT systems: Part 3: Semantic interoperability*. ISO/IEC 21823-3. 2021.
- [91] United Nations, Hrsg. *UNITED NATIONS FRAMEWORK CONVENTION ON CLIMATE CHANGE*. Juni 1992.
- [92] United Nations, Hrsg. *KYOTO PROTOCOL TO THE UNITED NATIONS FRAMEWORK CONVENTION ON CLIMATE CHANGE*. März 1998.
- [93] European Commission, Hrsg. *PARIS AGREEMENT: L 282*. Okt. 2016.
- [94] European Commission, Hrsg. *The European Green Deal: COM(2019) 640 final*. Dez. 2019.
- [95] Europäische Union, Hrsg. *REGULATION (EU) 2021/1119 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL: of 30 June 2021 establishing the framework for achieving climate neutrality and amending Regulations (EC) No 401/2009 and (EU) 2018/1999 (‘European Climate Law’)*. Juli 2021.

- [96] European Commission, Hrsg. *'Fit for 55': delivering the EU's 2030 Climate Target on the way to climate neutrality: COM(2021) 550 final*. Juli 2021.
- [97] European Commission, Hrsg. *In focus: Energy efficiency in buildings*. Feb. 2020.
- [98] Europäische Union, Hrsg. *REGULATION (EU) 2020/852 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 18 June 2020 on the establishment of a framework to facilitate sustainable investment, and amending Regulation (EU) 2019/2088*. Juni 2020.
- [99] *EU Taxonomy Compass*. abgerufen am: 13.12.2023. URL: <https://ec.europa.eu/sustainable-finance-taxonomy/taxonomy-compass/the-compass>.
- [100] Europäische Union, Hrsg. *DIRECTIVE 2010/31/EU OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 19 May 2010 on the energy performance of buildings*. Juni 2010.
- [101] European Commission, Hrsg. *Fragen und Antworten zur Überarbeitung der Richtlinie über die Gesamtenergieeffizienz von Gebäuden*. Dez. 2021.
- [102] ISO, Hrsg. *Energy management systems - Requirements with guidance for use*. ISO 50001. Aug. 2018.
- [103] Europäische Union, Hrsg. *DIRECTIVE (EU) 2023/1791 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 13 September 2023 on energy efficiency and amending Regulation (EU) 2023/955 (recast)*. Sep. 2023.
- [104] VDI, Hrsg. *Facility Management: Technisches Monitoring von Gebäuden*. VDI 6041. Juli 2017.
- [105] Arbeitskreis Maschinen- und Elektrotechnik staatlicher und kommunaler Verwaltungen, Hrsg. *Technisches Monitoring 2020: Technisches Monitoring als Instrument zur Qualitätssicherung*. Berlin, 2020.
- [106] ISO, Hrsg. *Internet of things (IoT) – Vocabulary*. ISO/IEC 20924. März 2021.
- [107] Michael Schluse u. a. „Experimentable Digital Twins—Streamlining Simulation-Based Systems Engineering for Industry 4.0“. In: *IEEE Transactions on Industrial Informatics* 14.4 (2018), S. 1722–1731. ISSN: 1551-3203. DOI: 10.1109/TII.2018.2804917.
- [108] Concetta Semeraro u. a. „Digital twin paradigm: A systematic literature review“. In: *Computers in Industry* 130 (2021), S. 103469. ISSN: 01663615. DOI: 10.1016/j.compind.2021.103469.
- [109] Mengnan Liu u. a. „Review of digital twin about concepts, technologies, and industrial applications“. In: *Journal of Manufacturing Systems* 58 (2021), S. 346–361. ISSN: 02786125. DOI: 10.1016/j.jmsy.2020.06.017.

- 
- [110] ISO/ IEC, Hrsg. *ISO/IEC JTC 1/SC 41: Internet of things and digital twin*. abgerufen am: 13.12.2023. URL: <https://www.iso.org/committee/6483279.html>.
- [111] Michael Grieves. *Digital Twin: Manufacturing Excellence through Virtual Factory Replication: A Whitepaper by Dr. Michael Grieves*. 2014.
- [112] Mike Shafto u. a. *DRAFT Modeling, Simulation, Information Technology & Processing Roadmap: Technology Area 11*. Hrsg. von National Aeronautics and Space Administration. 2010.
- [113] Sebastian Käbisch, Michael McCool und Ege Korkan. *Web of Things (WoT) Thing Description 1.1: W3C Proposed Recommendation*. abgerufen am: 06.12.2023. Juli 2023. URL: <https://www.w3.org/TR/wot-thing-description11/>.
- [114] Open Mobile Alliance Ltd, Hrsg. *Next Generation Service Interfaces Architecture: Approved Version 1.0*. Mai 2012. URL: [https://www.openmobilealliance.org/release/NGSI/V1\\_0-20120529-A/OMA-RD-NGSI-V1\\_0-20120529-A.pdf](https://www.openmobilealliance.org/release/NGSI/V1_0-20120529-A/OMA-RD-NGSI-V1_0-20120529-A.pdf).
- [115] Microsoft Azure, Hrsg. *Digital Twins Definition Language (DTDL): Version 3*. abgerufen am: 13.12.2023. URL: <https://azure.github.io/opensdigitaltwins-dtdl/DTDL/v3/DTDL.v3.html>.
- [116] Michael Jacoby und Thomas Usländer. „Digital Twin and Internet of Things—Current Standards Landscape“. In: *Applied Sciences* 10.18 (2020), S. 6519. DOI: 10.3390/app10186519.
- [117] Torben Miny u. a. „Overview and Comparison of Asset Information Model Standards“. In: *IEEE Access* 11 (2023), S. 99189–99221. DOI: 10.1109/ACCESS.2023.3312286.
- [118] Carlos Schmidt u. a. „Increasing Interoperability between Digital Twin Standards and Specifications: Transformation of DTDL to AAS“. In: *Sensors (Basel, Switzerland)* 23.18 (2023). DOI: 10.3390/s23187742.
- [119] DIN, Hrsg. *RM-SA - Referenzmodell für Industrie 4.0 Servicearchitekturen: Teil 1: Grundkonzepte einer interaktionsbasierten Architektur. Teil 1: Grundkonzepte einer interaktionsbasierten Architektur*. DIN SPEC 16593-1. 2018.
- [120] IEC, Hrsg. *Industrial-process measurement, control and automation - Life-cycle-management for systems and components*. IEC 62890. 2020.
- [121] IEC, Hrsg. *Enterprise-control system integration: Part 1: Models and terminology*. IEC 62264-1. 2013.
- [122] VDI/VDE, Hrsg. *Sprache für I4.0-Komponenten: Struktur von Nachrichten*. VDI 2193-1. 2020.

- [123] VDI/VDE-Gesellschaft, Hrsg. *Industrie 4.0 Begriffe/Terms: VDI Statusreport*. Apr. 2019.
- [124] ECLASS e.V., Hrsg. *Enable your global business and digitization*. abgerufen am: 13.12.2023. URL: <https://eclass.eu/en>.
- [125] DIN, Hrsg. *Genormte Datenelementtypen mit Klassifikationsschema für elektrische Betriebsmittel: Teil 1: Definitionen – Regeln und Methoden*. DIN EN 61360-1. Juli 2018.
- [126] Industrial Digital Twin Association e. V., Hrsg. *IDTA – working together to promote the Digital Twin*. abgerufen am: 13.12.2023. URL: <https://industrialdigitaltwin.org/en/>.
- [127] Steinbeis EU for You der Steinbeis Innovation GmbH, Hrsg. *Digitale Interoperabilität in kollaborativen Wertschöpfungsnetzwerken der Industrie 4.0*. abgerufen am: 13.12.2023. URL: <https://interopera.de/>.
- [128] IEC, Hrsg. *IEC Common Data Dictionary*. abgerufen am: 13.12.2023. URL: <https://cdd.iec.ch/cdd/iec61360/iec61360.nsf/TreeFrameset?OpenFrameSet>.
- [129] VDI/VDE-Gesellschaft, Hrsg. *Industrie 4.0 Begriffe/Terms: VDI Statusreport*. Apr. 2019.
- [130] Plattform Industrie 4.0, Hrsg. *Functional View of the Asset Administration Shell in an Industrie 4.0 System Environment*. Plattform Industrie 4.0.
- [131] Subash Kannoth u. a. „Enabling SMEs to Industry 4.0 Using the BaSyx Middleware: A Case Study“. In: *Software Architecture*. Hrsg. von Stefan Biffi u. a. Bd. 12857. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, S. 277–294. ISBN: 978-3-030-86043-1. DOI: 10.1007/978-3-030-86044-8{\textunderscore}19.
- [132] Pablo Oliveira Antonino u. a. „A Quality 4.0 Model for architecting industry 4.0 systems“. In: *Advanced Engineering Informatics* 54 (2022), S. 101801. ISSN: 14740346. DOI: 10.1016/j.aei.2022.101801.
- [133] Thomas Hansemann und Christof Hübner. *Gebäudeautomation: Kommunikationssysteme mit EIB/KNX, LON und BACnet*. 4., neu bearbeitete Auflage. München: Hanser, 2021. ISBN: 978-3-446-46843-6.
- [134] Frank Sokollik, Peter Helm und Ralph Seela. *KNX für die Gebäudesystemtechnik in Wohn- und Zweckbau*. Hrsg. von VDE Verlag. 6., neu bearbeitete und erweiterte Auflage. Berlin und Offenbach: VDE Verlag GmbH, 2017. ISBN: 978-3-8007-4055-0.
- [135] DIN EN, Hrsg. *Firmenneutrale Datenkommunikation für die Gebäudeautomation und Gebäudemanagement - Gebäude-Netzwerk-Protokoll: Teil 1: Datenprotokollschichtenmodell*. DIN EN 14908-1. Aug. 2014.

- 
- [136] ISO, Hrsg. *Building automation and control systems (BACS): Part 5: Data communication protocol. Part 5: Data communication protocol*. ISO 16484-5. Dez. 2017.
- [137] Jingkun Gao, Joern Ploennigs und Mario Berges. „A Data-driven Meta-data Inference Framework for Building Automation Systems“. In: *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. Hrsg. von David Culler, Yuvraj Agarwal und Rahul Mangharam. New York, NY, USA: ACM, 2015, S. 23–32. ISBN: 9781450339810. DOI: 10.1145/2821650.2821670.
- [138] Dezhi Hong u. a. „The Building Adapter“. In: *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. Hrsg. von David Culler, Yuvraj Agarwal und Rahul Mangharam. New York, NY, USA: ACM, 2015, S. 123–132. ISBN: 9781450339810. DOI: 10.1145/2821650.2821657.
- [139] Johannes Fütterer, Maksymilian Kochanski und Dirk Müller. „Application of selected supervised learning methods for time series classification in Building Automation and Control Systems“. In: *Energy Procedia* 122 (2017), S. 943–948. ISSN: 18766102. DOI: 10.1016/j.egypro.2017.07.428.
- [140] Gerrit Bode u. a. „A time series clustering approach for Building Automation and Control Systems“. In: *Applied Energy* 238 (2019), S. 1337–1345. ISSN: 03062619. DOI: 10.1016/j.apenergy.2019.01.196.
- [141] Jingkun Gao und Mario Bergés. „A large-scale evaluation of automated metadata inference approaches on sensors from air handling units“. In: *Advanced Engineering Informatics* 37 (2018), S. 14–30. ISSN: 14740346. DOI: 10.1016/j.aei.2018.04.010.
- [142] Zixiao Shi u. a. „Evaluation of Clustering and Time Series Features for Point Type Inference in Smart Building Retrofit“. In: *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. New York, NY, USA: ACM, 2019, S. 111–120. ISBN: 9781450370059. DOI: 10.1145/3360322.3360839.
- [143] Long Chen u. a. „A Metadata Inference Method for Building Automation Systems With Limited Semantic Information“. In: *IEEE Transactions on Automation Science and Engineering* 17.4 (2020), S. 2107–2119. ISSN: 1545-5955. DOI: 10.1109/TASE.2020.2990566.
- [144] Noah Mertens und Andreas Wilde. „Automated Classification of Datapoint Types in Building Automation Systems Using Time Series“. In: *Product Lifecycle Management. PLM in Transition Times: The Place of Humans and Transformative Technologies*. Hrsg. von Frédéric Noël u. a. Bd. 667. IFIP Advances in Informati-

- on and Communication Technology. Cham: Springer Nature Switzerland, 2023, S. 495–505. ISBN: 978-3-031-25181-8. DOI: [https://doi.org/10.1007/978-3-031-25182-5\\_48](https://doi.org/10.1007/978-3-031-25182-5_48).
- [145] Bharathan Balaji u. a. „Zodiac“. In: *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. Hrsg. von David Culler, Yuvraj Agarwal und Rahul Mangharam. New York, NY, USA: ACM, 2015, S. 13–22. ISBN: 9781450339810. DOI: [10.1145/2821650.2821674](https://doi.org/10.1145/2821650.2821674).
- [146] Arka A. Bhattacharya u. a. „Automated Metadata Construction to Support Portable Building Applications“. In: *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. Hrsg. von David Culler, Yuvraj Agarwal und Rahul Mangharam. New York, NY, USA: ACM, 2015, S. 3–12. ISBN: 9781450339810. DOI: [10.1145/2821650.2821667](https://doi.org/10.1145/2821650.2821667).
- [147] Dezhi Hong, Hongning Wang und Kamin Whitehouse. „Clustering-based Active Learning on Sensor Type Classification in Buildings“. In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. Hrsg. von James Bailey u. a. New York, NY, USA: ACM, 2015, S. 363–372. ISBN: 9781450337946. DOI: [10.1145/2806416.2806574](https://doi.org/10.1145/2806416.2806574).
- [148] Jason Koh u. a. „Scrabble“. In: *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments*. Hrsg. von Kamin Whitehouse, Prabal Dutta und Hae Young Noh. New York, NY, USA: ACM, 2017, S. 1–2. ISBN: 9781450355445. DOI: [10.1145/3137133.3141448](https://doi.org/10.1145/3137133.3141448).
- [149] Yang Jiao u. a. „SeNsER: Learning Cross-Building Sensor Metadata Tagger“. In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Hrsg. von Trevor Cohn, Yulan He und Yang Liu. Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, S. 950–960. DOI: [10.18653/v1/2020.findings-emnlp.85](https://doi.org/10.18653/v1/2020.findings-emnlp.85).
- [150] Jing Ma, Dezhi Hong und Hongning Wang. „Selective Sampling for Sensor Type Classification in Buildings“. In: *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2020, S. 241–252. ISBN: 978-1-7281-5497-8. DOI: [10.1109/IPSN48710.2020.00028](https://doi.org/10.1109/IPSN48710.2020.00028).
- [151] David Waterworth, Subbu Sethuvenkatraman und Quan Z. Sheng. „Advancing smart building readiness: Automated metadata extraction using neural language processing methods“. In: *Advances in Applied Energy* 3 (2021), S. 100041. ISSN: 26667924. DOI: [10.1016/j.adapen.2021.100041](https://doi.org/10.1016/j.adapen.2021.100041).

- 
- [152] Sicheng Zhan, Adrian Chong und Bertrand Lasternas. „Automated recognition and mapping of building management system (BMS) data points for building energy modeling (BEM)“. In: *Building Simulation* 14.1 (2021), S. 43–52. ISSN: 1996-3599. DOI: 10.1007/s12273-020-0612-7.
- [153] Fang He und Dan Wang. „Cloze: A Building Metadata Model Generation System Based on Information Extraction“. In: *Proceedings of the 9th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. BuildSys '22. New York, NY, USA: Association for Computing Machinery, 2022, S. 109–118. ISBN: 9781450398909. DOI: 10.1145/3563357.3564066.
- [154] Lu Lin u. a. „Sequential Learning with Active Partial Labeling for Building Metadata“. In: *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. New York, NY, USA: ACM, 2019, S. 189–192. ISBN: 9781450370059. DOI: 10.1145/3360322.3360866.
- [155] Burr Settles. *Active learning literature survey*. 2009.
- [156] Sebastian Ruder u. a. „Transfer Learning in Natural Language Processing“. In: *Proceedings of the 2019 Conference of the North*. Hrsg. von Anoop Sarkar und Michael Strube. Stroudsburg, PA, USA: Association for Computational Linguistics, S. 15–18. DOI: 10.18653/v1/N19-5004.
- [157] IDTA, Hrsg. *IDTA 02011-1-0: Hierarchical Structures enabling Bills of Material*. Apr. 2023.
- [158] IDTA, Hrsg. *Asset Interface Description - Submodel*. abgerufen am: 13.12.2023. URL: <https://github.com/admin-shell-io/submodel-templates/tree/main/development/Asset%20Interface%20Description/1/0>.
- [159] Björn Kämper und Jochen Müller. „Abbildung des BACnet-Kommunikationsstandards auf das Informationsmodell der Verwaltungsschale“. In: *AUTOMATION 2023, 23. Leitkongress der Mess- und Automatisierungstechnik*, VDI Verlag GmbH (2023).
- [160] Harish Kumar Pakala u. a. *Integration of asset administration shell and Web of Things*. 2021. DOI: 10.25673/39570.
- [161] IDTA, Hrsg. *Process description - Registration of AAS Submodel Templates for digital twins (IDTA Submodels) V1.0*. 2021.
- [162] Torben Miny. „Konzept für die semantische Interoperabilität zwischen Informationsmodellen“. Dissertation. RWTH Aachen, 2022.
- [163] VDI/VDE, Hrsg. *Sprache für I4.0-Komponenten: Interaktionsprotokoll für Ausschreibungsverfahren*. VDI 2193-2. 2020.

- [164] Jochen Müller und Maximilian Both. „Hersteller definieren Industrie-4.0-Verwaltungsschale für Pumpen: Pumpe wird zur Industrie-4.0-Komponente“. In: *Chemie Technik* (Apr. 2019).
- [165] OPC Foundation, Hrsg. *OPC 40223: Pumps and Vacuum pumps*. 2021.
- [166] ISO/IEC/IEEE, Hrsg. *Systems and software engineering - System life cycle processes*. ISO/IEC/IEEE 15288. 2023.
- [167] ISO/IEC/IEEE, Hrsg. *Systems and software engineering - Software life cycle processes*. ISO/IEC/IEEE 12207. 2017.
- [168] DIN, Hrsg. *Fertigungsverfahren - Begriffe, Einteilung*. DIN 8580. 2022.
- [169] DIN, Hrsg. *Instandhaltung - Begriffe der Instandhaltung*. DIN EN 13306. 2018.
- [170] Vereinte Nationen, Hrsg. *International Standard Industrial Classification of All Economic Activities (ISIC)*. Rev. 4. Bd. 4, Rev. 4. Economic & social affairs. New York, NY: United Nations, 2008. ISBN: 978-92-1-161518-0.
- [171] DIN, Hrsg. *Vakuumtechnik; Benennungen und Definitionen, Vakuumpumpen*. DIN 28400-2. 1980.
- [172] DIN, Hrsg. *Technische Anforderungen an Kreiselpumpen - Klasse II*. DIN EN ISO 5199. 2003.
- [173] ISO, Hrsg. *Vakuumtechnik; Verzeichnis von Fachausdrücken und Definitionen: Teil 1: Allgemeine Ausdrücke*. ISO 3529-1. 2019.
- [174] DIN, Hrsg. *Flüssigkeitspumpen und -installationen - Allgemeine Begriffe, Definitionen, Größen, Formelzeichen und Einheiten: Teil 1: Flüssigkeitspumpen*. DIN EN ISO 17769-1. 2012.
- [175] DIN, Hrsg. *Vakuumtechnik - Standardverfahren zur Messung der Leistungseffizienz von Vakuumpumpen: Teil 1: Grundlegende Beschreibung*. DIN ISO 21360-1. 2016.
- [176] ISO, Hrsg. *Zustandsüberwachung und -diagnostik von Maschinen - Begriffe*. ISO 13372. 2012.
- [177] DIN, Hrsg. *Instandhaltung - Dokumente für die Instandhaltung*. DIN EN 13460. 2009.
- [178] VDMA, Hrsg. *Geräteprofil für Flüssigkeits- und Vakuumpumpen: Teil 1: Geräteinformationsmodell - Allgemeine Profilelemente - Generische Pumpe. Teil 1: Geräteinformationsmodell - Allgemeine Profilelemente - Generische Pumpe*. VDMA 24223-1. 2008.
- [179] DIN, Hrsg. *OPC Unified Architecture: Teil 5: Informationsmodell. Teil 5: Informationsmodell*. DIN EN IEC 62541-5. 2021.

- 
- [180] DIN, Hrsg. *OPC Unified Architecture: Teil 3: Adressraummodell. Teil 3: Adressraummodell*. DIN EN IEC 62541-3. 2021.
- [181] DIN, Hrsg. *Internationales Elektrotechnisches Wörterbuch - Teil 351: Leittechnik*. DIN IEC 60050-351. 2014.
- [182] Maximilian Both, Jochen Müller und Christian Diedrich. „Automatisierte Abbildung semantisch heterogener I4.0-Verwaltungsschalen durch Methoden des Natural Language Processing“. In: *at - Automatisierungstechnik* 69.11 (2021), S. 940–951. ISSN: 0178-2312. DOI: 10.1515/auto-2021-0050.
- [183] TH Köln Labor für Gebäudeautomation und Regelungstechnik. *eclassCorpus (Revision 5603030)*. 2023. DOI: 10.57967/hf/0410. URL: <https://huggingface.co/datasets/gart-labor/eclassCorpus>.
- [184] TH Köln Labor für Gebäudeautomation und Regelungstechnik. *eclassQuery (Revision 4483d37)*. 2023. DOI: 10.57967/hf/0409. URL: <https://huggingface.co/datasets/gart-labor/eclassQuery>.
- [185] Nils Reimers und Iryna Gurevych. *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. Aug. 2019. URL: <http://arxiv.org/pdf/1908.10084v1>.
- [186] Yukun Zhu u. a. *Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books*. Juni 2015. URL: <http://arxiv.org/pdf/1506.06724v1>.
- [187] Suchin Gururangan u. a. „Don’t Stop Pretraining: Adapt Language Models to Domains and Tasks“. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Hrsg. von Dan Jurafsky u. a. Online: Association for Computational Linguistics, Juli 2020, S. 8342–8360. DOI: 10.18653/v1/2020.acl-main.740. URL: <https://aclanthology.org/2020.acl-main.740>.
- [188] Yi Luan u. a. „Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction“. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, S. 3219–3232. DOI: 10.18653/v1/D18-1360. URL: <https://www.aclweb.org/anthology/D18-1360>.
- [189] Kyle Lo u. a. „S2ORC: The Semantic Scholar Open Research Corpus“. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Hrsg. von Dan Jurafsky u. a. Stroudsburg, PA, USA: Association for Computational Linguistics, S. 4969–4983. DOI: 10.18653/v1/2020.acl-main.447.

- [190] Thomas Wolf u. a. „Transformers: State-of-the-Art Natural Language Processing“. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Hrsg. von Qun Liu und David Schlangen. Online: Association for Computational Linguistics, Okt. 2020, S. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6. URL: <https://aclanthology.org/2020.emnlp-demos.6>.
- [191] Yonghui Wu u. a. *Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*. Sep. 2016. URL: <http://arxiv.org/pdf/1609.08144v2>.
- [192] Nils Reimers. *Sentence Transformers Documentation*. abgerufen am: 13.12.2023. URL: <https://www.sbert.net/index.html>.
- [193] Adina Williams, Nikita Nangia und Samuel Bowman. „A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference“. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, 2018, S. 1112–1122. URL: <http://aclweb.org/anthology/N18-1101>.
- [194] Daniel Cer u. a. „SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation“. In: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, Canada: Association for Computational Linguistics, 2017, S. 1–14. DOI: 10.18653/v1/S17-2001. URL: <https://www.aclweb.org/anthology/S17-2001>.
- [195] Nils Reimers. *Sentence Transformers: Pretrained Models*. abgerufen am: 12.12.2023. URL: [https://www.sbert.net/docs/pretrained\\_models.html](https://www.sbert.net/docs/pretrained_models.html).
- [196] Kaitao Song u. a. *MPNet: Masked and Permuted Pre-training for Language Understanding*. 2020.
- [197] *Brick v1.3*. abgerufen am: 21.11.2023. URL: <https://brickschema.org/ontology/1.3/#v1.3>.
- [198] *Project Haystack*. abgerufen am: 21.11.2023. URL: <https://project-haystack.org/download>.
- [199] Jack W. Rae u. a. *Scaling Language Models: Methods, Analysis Insights from Training Gopher*. 2022. arXiv: 2112.11446 [cs.CL].
- [200] Victor Sanh u. a. *Multitask Prompted Training Enables Zero-Shot Task Generalization*. 2022. arXiv: 2110.08207 [cs.LG].
- [201] Colin Raffel u. a. „Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer“. In: *J. Mach. Learn. Res.* 21.1 (Jan. 2020). ISSN: 1532-4435.

- 
- [202] Pengfei Liu u. a. „Pre-Train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing“. In: *ACM Comput. Surv.* 55.9 (Jan. 2023). ISSN: 0360-0300. DOI: 10.1145/3560815. URL: <https://doi.org/10.1145/3560815>.
- [203] Rodrigo Nogueira u. a. „Document Ranking with a Pretrained Sequence-to-Sequence Model“. In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, 2020, S. 708–718. DOI: 10.18653/v1/2020.findings-emnlp.63. URL: <https://aclanthology.org/2020.findings-emnlp.63>.
- [204] Alina Cartus u. a. „Interoperability of semantically heterogeneous digital twins through Natural Language Processing methods: CLIMA 2022 conference, 2022: CLIMA 2022 The 14th REHVA HVAC World Congress“. In: (2022). DOI: 10.34641/clima.2022.143.
- [205] *Samples for Asset Administration Shell (AAS)*. abgerufen am: 13.12.2023. URL: <http://www.admin-shell-io.com/samples/>.
- [206] Jordan Hoffmann u. a. *Training Compute-Optimal Large Language Models*. 2022. arXiv: 2203.15556 [cs.CL].
- [207] Maximilian Both, Nicolai Maisch und Jochen Müller. „Semantische Interoperabilität durch Natural Language Processing als Basis für Self-X-Fähigkeiten von Verwaltungsschalen in semantisch heterogenen Asset-Netzwerken“. In: *Automation 2021*. Bd. 2392. 2021, S. 571–584. DOI: 10.51202/9783181023921.
- [208] Rebekka Benfer, Maximilian Both und Jochen Müller. „Verbesserung der Abbildungsgenauigkeit von heterogen ausgeprägten Industrie 4.0 Verwaltungsschalen durch die Integration von Knowledge Graphen“. In: *Automation 2023: transformation by automation, 24. Leitkongress der Mess- und Automatisierungstechnik : 27. und 28. Juni 2023, Baden-Baden*. Bd. 2419. 2021, S. 501–512. DOI: 10.51202/9783181024195.
- [209] Jo Beermann u. a. „Comparison of Different Natural Language Processing Models to Achieve Semantic Interoperability of Heterogeneous Asset Administration Shells“. In: *2023 IEEE 21st International Conference on Industrial Informatics (INDIN)*. 2023, S. 1–6. DOI: 10.1109/INDIN51400.2023.10218154.
- [210] RWTH IAT, Hrsg. *Semantic Matching Interface*. abgerufen am: 13.12.2023. URL: <https://github.com/rwth-iat/semantic-matching-interface>.
- [211] Tobias Meudt, Malte Pohl und Joachim Metternich. *Die Automatisierungspyramide - Ein Literaturüberblick*. de. Techn. Ber. Darmstadt, Juni 2017. URL: <http://tuprints.ulb.tu-darmstadt.de/6298/>.

- [212] DIN, Hrsg. *Integration von Unternehmensführungs- und Leitsystemen – Teil 1: Modelle und Terminologie*. DIN EN 62264-1. 2014.
- [213] Maximilian Both u. a. „Automated monitoring applications for existing buildings through natural language processing based semantic mapping of operational data and creation of digital twins“. In: *Energy and Buildings* 300 (2023), S. 113635. ISSN: 03787788. DOI: 10.1016/j.enbuild.2023.113635.
- [214] VDI, Hrsg. *VDI 3814 Blatt 4.1: Gebäudeautomation (GA) - Methoden und Arbeitsmittel für Planung, Ausführung und Übergabe - Kennzeichnung, Adressierung und Listen*. Jan. 2019.
- [215] Branden Chan, Stefan Schweter und Timo Möller. „German’s Next Language Model“. In: *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, 2020, S. 6788–6796. DOI: 10.18653/v1/2020.coling-main.598. URL: <https://aclanthology.org/2020.coling-main.598>.
- [216] DIN e. V. und DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE, Hrsg. *IDiS PILOTEN 2021/2022: Management Summary*. 2023.
- [217] Harsurinder Kaur, Husanbir Singh Pannu und Avleen Kaur Malhi. „A Systematic Review on Imbalanced Data Challenges in Machine Learning: Applications and Solutions“. In: *ACM Comput. Surv.* 52.4 (2019). ISSN: 0360-0300. DOI: 10.1145/3343440.
- [218] IDTA, Hrsg. *IDTA 02008-1-1: Time Series Data*. März 2023.

## Ehrenerklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit vollständig ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe.

Alle verwendeten fremden und eigenen Quellen sind als solche kenntlich gemacht und im Falle einer Ko-Autorenschaft, insbesondere im Rahmen einer kumulativen Dissertation, ist der Eigenanteil richtig und vollständig ausgewiesen. Insbesondere habe ich nicht die Hilfe einer kommerziellen Promotionsberaterin/eines kommerziellen Promotionsberaters in Anspruch genommen. Dritte haben von mir weder unmittelbar noch mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen.

Ich habe insbesondere nicht wissentlich:

- Ergebnisse erfunden oder widersprüchliche Ergebnisse verschwiegen,
- statistische Verfahren absichtlich missbraucht, um Daten in ungerechtfertigter Weise zu interpretieren,
- fremde Ergebnisse oder Veröffentlichungen plagiiert,
- fremde Forschungsergebnisse verzerrt wiedergegeben.

Mir ist bekannt, dass Verstöße gegen das Urheberrecht Unterlassungs- und Schadensersatzansprüche der Urheberin/des Urhebers sowie eine strafrechtliche Ahndung durch die Strafverfolgungsbehörden begründen können.

Ich erkläre mich damit einverstanden, dass die Dissertation ggf. mit Mitteln der elektronischen Datenverarbeitung auf Plagiate überprüft werden kann.

Die Arbeit wurde bisher weder im Inland noch im Ausland in gleicher oder ähnlicher Form als Dissertation eingereicht und ist als Ganzes auch noch nicht veröffentlicht.

Magdeburg, den 30. Januar 2024

Maximilian Both

## Eigene Veröffentlichungen

- **Automated monitoring applications for existing buildings through natural language processing based semantic mapping of operational data and creation of digital twins**  
Both, M.; Kämper, B.; Cartus, A.; Beermann, J.; Fessler, T.; Müller, J.; Diedrich, C., Energy and Buildings, Band 300, Artikel 113635, DOI: <https://doi.org/10.1016/j.enbuild.2023.113635>, *Peer Review*
- **Comparison of Different Natural Language Processing Models to Achieve Semantic Interoperability of Heterogeneous Asset Administration Shells**  
Beermann, J.; Benfer, R.; Both, M.; Müller, J.; Diedrich, C., IEEE 21st International Conference on Industrial Informatics (INDIN) (2023), DOI: 10.1109/INDIN51400.2023.10218154, *Peer Review*
- **Verbesserung der Abbildungsgenauigkeit von heterogen ausgeprägten Industrie 4.0 Verwaltungsschalen durch die Integration von Knowledge Graphen**  
Benfer, R.; Both, M.; Müller, J., Automation 2023: transformation by automation, 24. Leitkongress der Mess- und Automatisierungstechnik, Bandnummer: 2419, DOI: 10.51202/9783181024195
- **Konfigurationsfreies Energiemonitoring (Self-X) auf Basis einer automatisierten Abbildung von Kommunikationsnetzwerken durch Natural Language Processing**  
Both, M.; Müller, J.; Diedrich, C.; Automation 2023: transformation by automation, 24. Leitkongress der Mess- und Automatisierungstechnik, Bandnummer: 2419, DOI: 10.51202/9783181024195
- **Reducing configuration efforts in energy management systems based on natural language processing methods and asset administration shells**  
Both, M.; Müller, J.; Diedrich, C., IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA) (2022), DOI: 10.1109/ETFA52439.2022.9921479, *Peer Review*

- **Interoperability of semantically heterogeneous digital twins through Natural Language Processing methods**  
Cartus, A.; Both, M.; Maisch, N.; Müller, J.; Diedrich, C., CLIMA 2022, REHVA 14th HVAC World Congress (2022), DOI: <https://doi.org/10.34641/clima.2022.143>, *Peer Review*
- **Automated performance monitoring of HVAC components by artificial intelligence**  
Both, M.; Maisch, N.; Kämper, B.; Cartus, A.; Müller, J.; Diedrich, C., CLIMA 2022, REHVA 14th HVAC World Congress (2022), DOI: <https://doi.org/10.34641/clima.2022.144>, *Peer Review*
- **Automatisierte Abbildung semantisch heterogener I4.0-Verwaltungsschalen durch Methoden des Natural Language Processing**  
Both, M.; Müller, J.; Diedrich, C., at - Automatisierungstechnik, Band 69, Heft 11, 2021, DOI: <https://doi.org/10.1515/auto-2021-0050>, *Peer Review*
- **Semantische Interoperabilität durch Natural Language Processing als Basis für Self-X-Fähigkeiten von Verwaltungsschalen in semantisch heterogenen Asset-Netzwerken**  
Both, M.; Maisch, N.; Müller, J., Automation 2021: Navigating towards resilient Production, 22. Leitkongress der Mess- und Automatisierungstechnik, Bandnummer: 2392, DOI: [10.51202/9783181023921](https://doi.org/10.51202/9783181023921)
- **Digitization of pumps - Industrie 4.0 submodels for liquid and vacuum pumps**  
Both, M.; Müller, J., 4th International Rotating Equipment Conference (2019)
- **Entwicklung einer Industrie 4.0 Verwaltungsschale auf Basis des allgemeinen Geräteprofils für Pumpen**  
Both, M.; Müller, J., Automation 2018: Seamless Convergence of Automation & IT, 19. Leitkongress der Mess- und Automatisierungstechnik, Bandnummer: 2330, DOI: [10.51202/9783181023303](https://doi.org/10.51202/9783181023303)

## A Definitionen ChatGPT

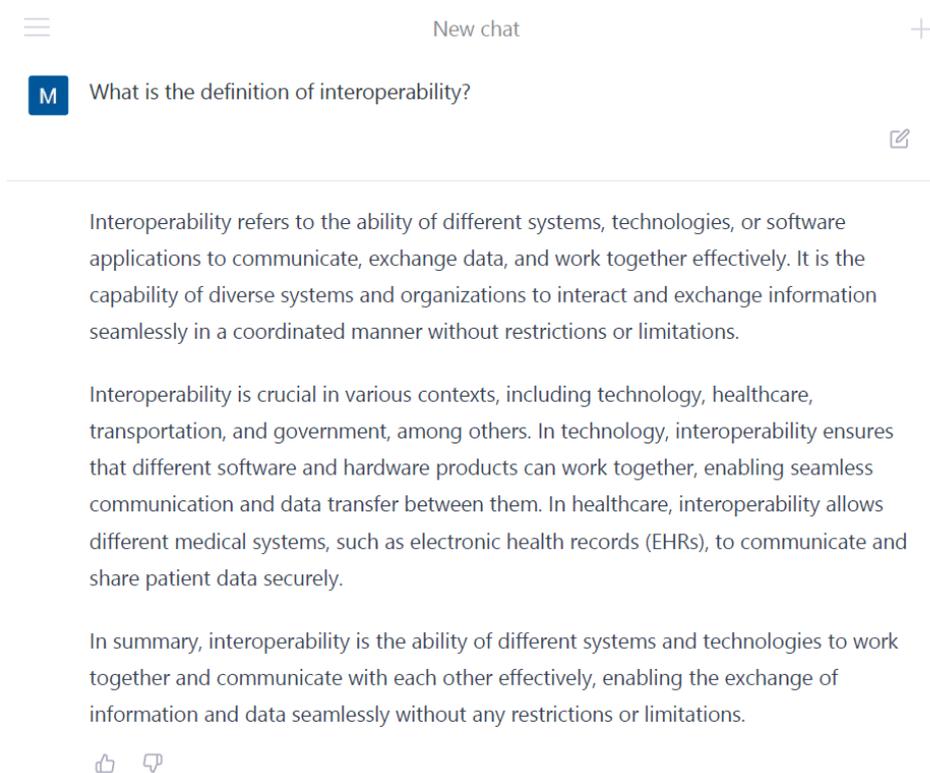


Abbildung A.1: Definition Interoperabilität des Chatbots ChatGPT

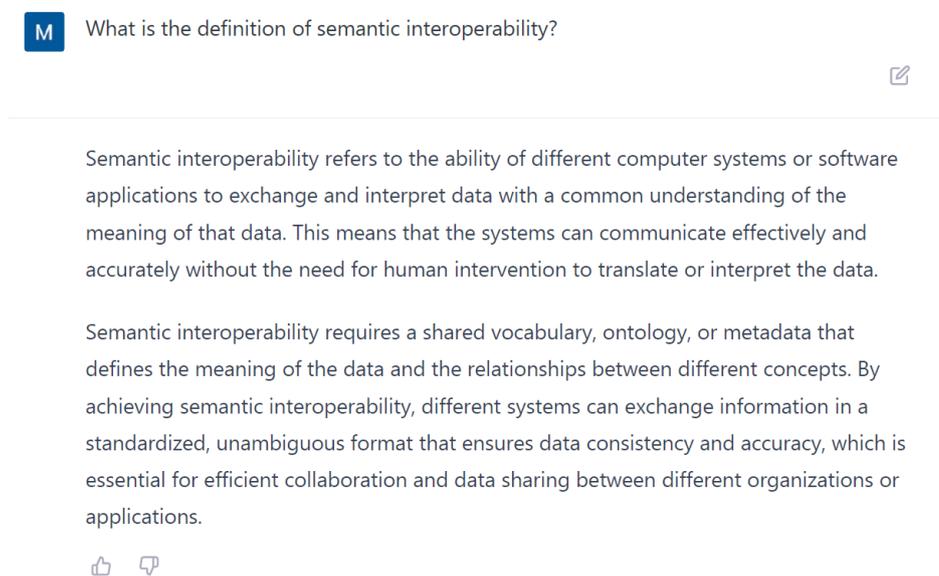
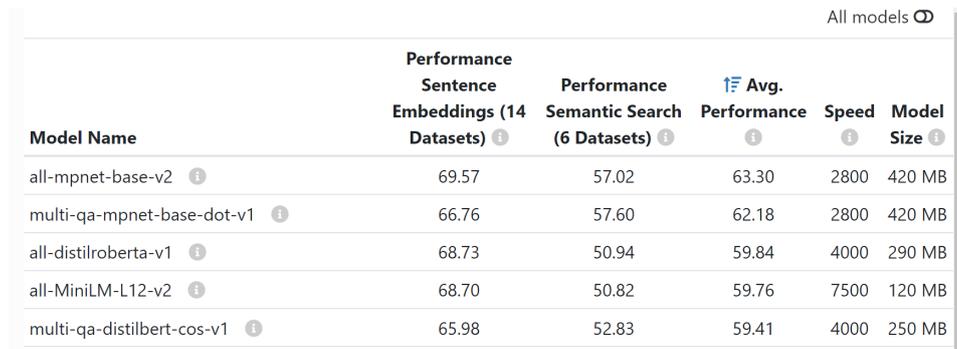


Abbildung A.2: Definition semantische Interoperabilität des Chatbots ChatGPT

## B Beste Modelle im Bereich Sentence Embeddings



All models 

Model Name	Performance Sentence Embeddings (14 Datasets) 	Performance Semantic Search (6 Datasets) 	Avg. Performance 	Speed 	Model Size 
all-mpnet-base-v2 	69.57	57.02	63.30	2800	420 MB
multi-qa-mpnet-base-dot-v1 	66.76	57.60	62.18	2800	420 MB
all-distilroberta-v1 	68.73	50.94	59.84	4000	290 MB
all-MiniLM-L12-v2 	68.70	50.82	59.76	7500	120 MB
multi-qa-distilbert-cos-v1 	65.98	52.83	59.41	4000	250 MB

Abbildung B.1: Screenshot der aktuell besten Modelle im Bereich SEms und Semantic Search

## C Übersicht Datensatz auf Datenpunktebene

Tabelle C.1: Übersicht der Komponenten mit möglichen Datenpunkten

Unterfunktion	Komponente	Datenpunkte
Wärmeerzeugen	BHKW	Blockierschutz, Laufzeit Nächste Wartung, Messwert Abgastemperatur, Messwert Drehzahl, Messwert Druck, Messwert Frequenz, Messwert Gas, Messwert Klappe, Messwert Lambda, Messwert Leistung, Messwert Spannung, Messwert Strom, Messwert Temperatur, Messwert Temperatur Generator, Messwert Wärmemenge, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Netzparallel, Rückmeldung Start, Rückmeldung Ventil, Rückmeldung Ölnachspeisung Aktiv, Schaltbefehl Anlage, Seriennummer, Sollwert Leistung, Stellbefehl, Störmeldung, Warnmeldung, Wartungsmeldung
	Pelletkessel	Messwert Abgastemperatur, Messwert Außentemperatur, Messwert Primärluft, Messwert Sauerstoff, Messwert Sekundärluft, Messwert Temperatur, Messwert Temperatur Einschubrohr, Messwert Drehzahl, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Schaltbefehl, Sollwert Temperatur, Störmeldung
	Wärmepumpe	Blockierschutz, Messwert Wasserspiegel Förderbrunnen, Rückmeldung Betrieb, Rückmeldung Betriebsbereit, Rückmeldung Betriebsstunden, Rückmeldung Blockierschutz Brunnenpumpe, Rückmeldung Blockierschutz Umwälzpumpe, Rückmeldung Handschaltung, Rückmeldung Umschaltventil Zu, Rückmeldung Umschaltventil Auf, Schaltbefehl, Stellbefehl, Störmeldung, Zähler Strom, Zähler Wärmemenge

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Wärme erzeugen	Kessel	Alarmmeldung, Anforderung, Anzahl Schaltungen, Ausschalt Differenz, Blockierschutz, Freigabe, Messwert Druck, Messwert Gas, Messwert Leistung, Messwert Temperatur, Reglerparameter, Reset Betriebsstunden, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Heizperiode, Rückmeldung Klappe, Rückmeldung Managementbedieneinrichtung, Rückmeldung Modulationsgrad, Rückmeldung Not Aus, Rückmeldung Ventil, Rückmeldung Zeitprogramm, Schaltbefehl Anlage, Schaltbefehl Heizperiode Nach Datum, Schaltbefehl Heizperiode Nach Temperatur, Schaltbefehl Klappe, Schaltbefehl Sommer Monat Ein, Schaltbefehl Winter Monat Ein, Schalthysterese, Seriennummer, Sollwert Abschaltung, Sollwert Einschalten Sommer, Sollwert Kesselanzahl, Sollwert Laufzeit, Sollwert Leistung, Sollwert Modulation, Sollwert Sommer Tag, Sollwert Temperatur, Stellbefehl Anlage, Stellbefehl Klappe, Stellbefehl Ventil, Störmeldung, Warnmeldung, Wartungsmeldung, Zähler Wärmemenge, Übersteuerung
	Wärmeversorger allgemein	Alarmmeldung, Freigabe Betrieb Comfort Economy, Freigabe Betrieb Winter, Freigabe Filter, Freigabe Sondertageprogramm, Freigabe Wärmeerzeuger, Messwert Außentemperatur, Mindestausschaltzeit, Mindesteinschaltzeit, Optimierung, Quittierung, Regelparameter, Rückmeldung Betrieb, Rückmeldung Erzeuger, Rückmeldung Handschaltung, Rückmeldung Klappe, Rückmeldung Sommer Winter, Rückmeldung Start Stopp, Rückmeldung Wärmeanforderung, Rückmeldung Zeitprogramm, Schaltbefehl, Schaltbefehl Klappe, Schaltbefehl Sommer Winter, Schaltbefehl Wärmeanforderung, Sollwert Außentemperatur Abschaltung Nacht, Sollwert Außentemperatur Abschaltung Tag, Sollwert Temperatur, Sollwert Wärmeanforderung, Sollwert Wärmebedarf, Sollwert Zeit Berechnung, Sollwert glt, Störmeldung, Vorrangschaltung, Zeitangabe, Zähler Gas, Übersteuerung

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Wärme verteilen	Heizkreis allgemein (1)	Alarmmeldung, Anforderung, Ausschaltzeit, Device Description, Einschaltzeit, Externe Vorrangschaltung, Grenzwert Frost, Grenzwert Temperatur, Heizkurve, Hysterese Frostschutz, Hysterese Heizkreis, Messwert Außentemperatur, Messwert Feuchte, Messwert Leistung, Messwert Temperatur, Messwert Volumenstrom, Messwert Windgeschwindigkeit, Messwert Wärmeleistung, Nachlaufdauer Klappe Wärmetauscher, Nachtabsenkung Tage, Quittierung, Reglerparameter, Restlaufzeit Nutzzeitverlängerung, Rückmeldung Aufheizbetrieb, Rückmeldung Betrieb, Rückmeldung Betriebsart, Rückmeldung Betriebsstunden, Rückmeldung Frostschutz, Rückmeldung Frostschutz Handschalter, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Nachtabsenkung, Rückmeldung Nutzzeitverlängerung, Rückmeldung Regelabweichung, Rückmeldung Restwärmeoptimierung, Rückmeldung Tagbetrieb, Rückmeldung Zeitprogramm, Schaltbefehl Anlage, Schaltbefehl Betriebsart, Schaltbefehl Gleitendes Schalten, Schaltbefehl Nachtabsenkung, Schaltbefehl Nutzzeitverlängerung, Schaltbefehl Optimierung, Schaltbefehl Start Stop Optimierung, Schaltbefehl Zeitprogramm, Sollwert Abschalten Stützbetrieb, Sollwert Aufheizzeit, Sollwert Außentemperatur, Sollwert Außentemperatur Abschalten Heizung, Sollwert Außentemperatur Einschalten Heizung, Sollwert Außentemperatur Start Stopp Aktiv, Sollwert Delta T Vorlauf Rücklauf, Sollwert Economy, Sollwert Ferien, Sollwert Frostschutz, Sollwert Grenztemperatur Start Stopp, Sollwert Hysterese Aufheizbetrieb, Sollwert Hysterese Nachtabsenkung, Sollwert Komfort, Sollwert Laufzeit Blockierschutz, Sollwert Maximale Aufheizzeit, Sollwert Minimale Außentemperatur, Sollwert Nacht, Sollwert Nachtabsenkung, Sollwert Nutzzeitverlängerung, Sollwert Schutzbetrieb, Sollwert Speicherfähigkeit, Sollwert StandBy Nacht, Sollwert StandBy Tag

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Wärme verteilen	Heizkreis allgemein (2)	Sollwert Stützbetrieb Nacht, Sollwert Stützbetrieb Nacht Hysterese, Sollwert Stützbetrieb Tag Hysterese, Sollwert Stützbetrieb Tag, Sollwert Tag, Sollwert Temperatur, Sollwert Temperatur Start Aufheizen, Sollwert Zeitkonstante, Sollwert Überhöhung Hydraulische Weiche, Sollwert Überhöhung Wärmeanforderung, Sollwertverschiebung, Stellbefehl, Stunden Sollwert Außentemperatur, Störmeldung, Stützbetrieb Nacht Erreicht, Stützbetrieb Tag Erreicht
	Druckhaltestation	Messwert, Störmeldung
	Heizkurve	Aufheizbetrieb, Außentemperatur X1, Außentemperatur X2, Außentemperatur X3, Außentemperatur X4, Messwert Außentemperatur, Rückmeldung Managementbedieneinrichtung, Rückmeldung Verschiebung, Sollwert, Sollwert Fußpunkt, Sollwert Krümmung, Sollwert X1, Sollwert X2, Sollwert X3, Sollwert X4, Sollwert Y1, Sollwert Y2, Sollwert Y3, Sollwert Y4, Sollwertverschiebung, Steilheit, Stützbetrieb X1, Stützbetrieb X2, Stützbetrieb X3, Stützbetrieb X4, Stützbetrieb Y1, Stützbetrieb Y2, Stützbetrieb Y3, Stützbetrieb Y4, Stützpunkt X1, Stützpunkt X2, Stützpunkt X3, Stützpunkt X4, Stützpunkt Y1, Stützpunkt Y2, Stützpunkt Y3, Stützpunkt Y4
	Kältemengenzähler	Messwert Kälteleistung, Messwert Kältemenge, Messwert Rücklauftemperatur, Messwert Volumen, Messwert Volumenstrom, Messwert Vorlauftemperatur

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Wärme verteilen	Pumpe	Alarmmeldung, Anforderung, Anzahl Schaltungen, Blockierschutz, Freigabe, Hysterese Frostschutz, Messwert Drehzahl, Messwert Druck, Messwert Durchfluss, Messwert Energieverbrauch, Messwert Förderhöhe, Messwert Leistung, Messwert Leistungsaufnahme, Messwert Stromaufnahme, Messwert Temperatur, Messwert Volumenstrom, Reset Betriebsstunden, Reset Wartungsintervall, Rückmeldung Auto, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Handschaltung Pumpe, Rückmeldung Managementbedieneinrichtung, Rückmeldung Zeitprogramm, Schaltbefehl, Schaltbefehl Blockierschutz, Schaltbefehl Wechsel Doppelpumpe, Schalthysterese, Sollwert, Sollwert Drehzahl, Sollwert Frostschutz, Sollwert Intervall Blockierschutz, Sollwert Laufzeit Blockierschutz, Sollwert Nachlaufzeit, Sollwert Nacht, Sollwert Pumpe Ein Außentemperatur, Sollwert Reset Betriebsstunden, Sollwert Tag, Stellbefehl, Störmeldung, Wartungsintervall, Wartungsmeldung, Übersteuerung
	Raum	Fensteranteil, Freigabe Externer Sollwert, Freigabe Heizung, Freigabe Optimum Stopp, Freigabe Raumkorrektur, Freigabe Stellantrieb, Freigabe Stützbetrieb Nacht, Freigabe Stützbetrieb Nacht Ventil, Freigabe Stützbetrieb Tag, Freigabe Stützbetrieb Tag Ventil, Freigabe Zeitprogramm, Messwert CO <sub>2</sub> , Messwert Raumtemperatur, Präsenzmelder, Rückmeldung Absenkbetrieb, Rückmeldung Aufheizbetrieb, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Normalbetrieb, Schaltbefehl Raumkorrektur, Schaltbefehl Ventil, Sollwert Absenkbetrieb, Sollwert CO <sub>3</sub> , Sollwert Raumkorrektur, Sollwert Raumtemperatur Nacht, Sollwert Raumtemperatur Normalbetrieb, Sollwert Raumtemperatur Schutzbetrieb, Sollwert Raumtemperatur Tag, Sollwert Stützbetrieb Nacht, Sollwert Stützbetrieb Tag, Sollwert minimale Raumtemperatur, Sollwertsteller, Sollwertverschiebung, Stellbefehl, Störmeldung

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Wärme verteilen	Regler	Abtastzeit, Außentemperatur Errechnet, D-Anteil, I-Anteil, P-Anteil, Proportionalband, Reglerausgang, Reglereingang, Reset, Rückmeldung Betrieb, Rückmeldung Handschaltung, Rückmeldung Regelabweichung, Schaltbefehl, Sollwert Nachstellzeit, Sollwert Regler, Sollwert Regler Max, Sollwert Regler Min, Stellbefehl, Störmeldung, Verschiebung
	Rücklauf	Arbeitspunkt, Grenzwert Winteranfahen, Messwert Rücklauf-temperatur, Rücklauf Begrenzung, Rücklauf-temperatur Errechnet, Schaltbefehl Begrenzung, Schaltbefehl Frostschutz, Sollwert Frostschutz, Sollwert Maximale Rücklauf-temperatur, Sollwert Minimale Rücklauf-temperatur, Sollwert Rücklauf-temperatur Errechnet, Störmeldung
	Übertrager	Heizkörperexponent, Rückmeldung Heizdecke, Rückmeldung Ventil Zone, Schaltbefehl Heizdecke, Schaltbefehl Ventil Zone, Vorregelung Deckenstrahlplatten Messwert Temperatur, Vorregelung Deckenstrahlplatten Sollwert, Vorregelung Deckenstrahlplatten Status, Vorregelung Deckenstrahlplatten Stellbefehl, Vorregelung Deckenstrahlplatten Temperatur Max, Vorregelung Deckenstrahlplatten Temperatur Min, Vorregelung Deckenstrahlplatten Temperaturdifferenz, Vorregelung Deckenstrahlplatten Zeit
	Ventil	Laufzeit Ventil, Laufzeit 3 Punkt Antrieb, Rückmeldung Betrieb, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Stellsignal, Schaltbefehl, Sicherheit, Sollwert, Sollwert Handschaltung, Sollwert Mischventil, Sollwert Stellsignal Max, Sollwert Stellsignal Min, Stellbefehl, Stellbefehl Max, Stellbefehl Min, Stellbefehl Mischventil, Störmeldung

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Wärme verteilen	Vorlauf	Anhebung Vorlauftemperatur, Arbeitspunkt, Frostschutz Vorlauftemperatur, Messwert Vorlauftemperatur, Messwert Vorlauftemperatur Errechnet, Rückmeldung Managementbedieneinrichtung, Rückmeldung Regelabweichung, Schaltbefehl Dämpfung, Schaltbefehl Optimierung, Sollwert Maximale Vorlauftemperatur, Sollwert Minimale Vorlauftemperatur, Sollwert Nachtabsenkung Vorlauftemperatur, Sollwert Vorlauftemperatur, Sollwert Vorlauftemperatur Errechnet, Sollwertkorrektur Vorlauftemperatur, Störmeldung
	Warmwasserbereitung	Alarmmeldung, Anzahl Stufen, Ausschalttemperatur, Ausschaltzeit, Blockierschutz, Einschalttemperatur, Einschaltzeit, Freigabe, Handschaltung Legionellen, Hysterese Sollwert Warmwasserbereitung, Legionellenschaltung Tage, Messwert Wassertemperatur, Messwert Zapftemperatur, Messwert Zirkulationstemperatur, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Handschaltung Auto, Rückmeldung Legionellenschaltung, Rückmeldung Legionellenschaltung Handschaltung, Rückmeldung Legionellenschaltung Handschaltung Auto, Schaltbefehl, Schaltbefehl Legionellenschaltung, Sollwert Maximale Laufzeit Legionellen, Sollwert Temperatur Legionellen, Sollwert Warmwasserbereitung, Sollwert Zirkulationstemperatur, Sollwert Zirkulationstemperatur Legionellen, Stellbefehl, Störmeldung, Warnmeldung Legionellen, Wartungsmeldung, Wochenprogramm
	Wärmemengenzähler	Messwert Durchfluss, Messwert Kältemenge, Messwert Leistung, Messwert Rücklauftemperatur, Messwert Temperaturdifferenz, Messwert Volumenstrom, Messwert Vorlauftemperatur, Messwert Wärmeleistung, Messwert Wärmemenge, Messwert Wärmemenge Energie, Messwert Wärmemenge Volumen, Störmeldung

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Wärme speichern	Speicher	Ausschaltgrenzwert, Einschaltgrenzwert, Messwert Außentemperatur, Messwert Speichertemperatur, Messwert Speichertemperatur Mitte, Messwert Speichertemperatur Oben, Messwert Speichertemperatur Unten, Rückmeldung Handschaltung, Schaltbefehl, Schaltdifferenz, Sollwert Speichertemperatur, Sollwert Speichertemperatur Oben, Sollwert Speichertemperatur Unten, Störmeldung, Vorrangschaltung
Wärme beziehen	Fernwärme	Alarmmeldung, Anforderung, Blockierschutz, Freigabe, Heizkurve, Messwert Außentemperatur, Messwert Drehzahl, Messwert Druck, Messwert Leistung, Messwert Rücklauftemperatur, Messwert Rücklauftemperatur Primär, Messwert Rücklauftemperatur Sekundär, Messwert Temperatur, Messwert Vorlauftemperatur, Messwert Vorlauftemperatur Primär, Messwert Vorlauftemperatur Sekundär, Messwert Durchfluss, Offset Vorlauftemperatur, Reglerparameter, Reset Betriebsstunden, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Klappe, Rückmeldung Pumpe, Rückmeldung Regelabweichung, Rückmeldung Ventil, Schaltbefehl Anlage, Schaltbefehl Klappe, Schaltbefehl Pumpe, Schaltbefehl Ventil, Sollwert Außentemperatur Ein Aus, Sollwert Nachlaufzeit Pumpe, Sollwert Nacht, Sollwert Rücklauftemperatur, Sollwert Rücklauftemperaturbegrenzung, Sollwert Temperatur, Sollwert Ventil, Sollwert Vorlauftemperatur, Sollwert Vorlauftemperaturbegrenzung, Sollwertverschiebung, Stellbefehl Pumpe, Stellbefehl Ventil, Störmeldung, Warnmeldung, Zähler

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Luft bereitstellen	Abluft allgemein	Alarmmeldung, Befehlsausführkontrolle, Messwert Druck, Messwert Feuchte, Messwert Luftqualität, Messwert Temperatur, Messwert Volumenstrom, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Nutzzeitverlängerung, Rückmeldung Stellsignal, Rückmeldung Zeitprogramm, Schaltbefehl Anlage, Sollwert CO2 Konzentration, Sollwert Druck, Sollwert Feuchte, Sollwert Temperatur, Sollwert Volumenstrom, Sollwert Zeitverzögerung, Stellbefehl, Störmeldung, Warnmeldung Feuchte, Wartungsmeldung, Werkseinstellungen
	Abluftfilter, Außenluftfilter, Filter, Zuluftfilter	Alarmmeldung, Messwert Druck, Störmeldung, Wartungsmeldung
	Abluftklappe, Außenluftklappe, Fortluftklappe, Klappe, Zuluftklappe	Alarmmeldung, Befehlsausführkontrolle, Rückmeldung Stellsignal, Rückmeldung Betrieb, Rückmeldung Handschaltung, Rückmeldung Klappe Auf, Rückmeldung Klappe Zu, Rückmeldung Managementbedieneinrichtung, Schaltbefehl, Sollwert Stellsignal, Stellbefehl, Störmeldung
	Abluftventilator	Alarmmeldung, Anzahl Schaltungen, Befehlsausführkontrolle, Laufüberwachung, Messwert Druck, Reset Betriebsstunden, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Nutzzeitverlängerung, Rückmeldung Reperaturschalter, Rückmeldung Reset Betriebsstunden, Rückmeldung Stellsignal, Schaltbefehl, Sollwert Laufzeit, Sollwert Stellsignal, Stellbefehl, Störmeldung, Wartungsmeldung

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Luft bereitstellen	Befeuchter	Reset Betriebsstunden, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Stellsignal, Schaltbefehl, Sollwert Befeuchten, Stellbefehl, Störmeldung
	Erhitzer	Alarmmeldung Frostschutz, Befehlsausführkontrolle, Freigabe, Messwert Energieverbrauch, Messwert Leistung, Messwert Temperatur, Messwert Volumen, Messwert Volumenstrom, Reset Betriebsstunden, Rückmeldung Betrieb, Rückmeldung Handschaltung, Rückmeldung Stellsignal, Schaltbefehl, Schaltbefehl Blockierschutz, Schaltbefehl Frostschutz, Sollwert Dauerfreigabe, Sollwert Frostschutz, Sollwert Laufzeit Blockierschutz, Sollwert Nachstellzeit, Sollwert Stellsignal, Sollwert Temperatur, Spülzeit, Stellbefehl, Stellbefehl Ventil, Störmeldung
	Gerät allgemein	Alarmmeldung, Anforderung Tableau, Anlagenuhr, Ausschaltzeit, Einschaltzeit, Messwert Außenfeuchte, Messwert Außentemperatur, Quittierung, Rückmeldung Anfahrbetrieb, Rückmeldung Anlage Fern, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Entfeuchtung, Rückmeldung Freie Nachtkühlung, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Nutzzeitverlängerung, Rückmeldung Präsenzmelder, Rückmeldung Restlaufzeit Nutzzeitverlängerung, Rückmeldung Spülen, Rückmeldung Zeitprogramm, Schaltbefehl, Schaltbefehl Anlage, Schaltbefehl Anlage Fern, Schaltbefehl Nachtkühlung, Schaltbefehl Nutzzeitverlängerung, Schaltbefehl Quittierung, Schaltbefehl Tagesprogramm, Schaltbefehl Zeitprogramm, Schalthysterese, Sollwert Auskühlschutz, Sollwert Feuchte, Sollwert Freie Nachtkühlung, Sollwert Kühlbedarf, Sollwert Maximale Einschaltverzögerung, Sollwert Minimale Einschaltverzögerung, Sollwert Nutzzeitverlängerung, Sollwert Spülzeit, Sollwert Wärmebedarf, Stellbefehl, Störmeldung, Zurücksetzen, Übersteuert

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Luft bereitstellen	Kältemengenzähler	Messwert Kälteleistung, Messwert Kältemenge, Messwert Volumen, Messwert Volumenstrom
	Kühler	Alarmmeldung Frostschutz, Befehlsausführkontrolle, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Stellsignal, Schaltbefehl, Sollwert Kühlbedarf, Stellbefehl, Störmeldung, Zählwert Kältemenge
	Regler	D-Anteil, Grenzwert Regler, Hysterese, I-Anteil, P-Anteil, Proportionalband, Reglerausgang, Reglereingang, Reset, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Regelabweichung, Schaltbefehl, Sollwert Nachstellzeit, Sollwert Regler, Sollwert Regler Max, Sollwert Regler Min, Stellbefehl, Störmeldung, Stützwert, Totzone, Wirksinn
	Umluft	Messwert Temperatur, Rückmeldung Betrieb, Rückmeldung Handschaltung, Rückmeldung Stellsignal, Stellbefehl, Störmeldung
	Ventilator	Alarmmeldung, Anzahl Schaltungen, Befehlsausführkontrolle, Rückmeldung Betrieb, Rückmeldung Drehzahl, Rückmeldung Handschaltung, Schaltbefehl, Sollwert Laufzeit, Sollwert Stellsignal, Stellbefehl, Störmeldung, Wartungsmeldung
	Wärmemengenzähler	Messwert Leistung, Messwert Temperaturdifferenz, Messwert Wärmemenge
	Wärmrückgewinnung	Alarmmeldung, Alarmmeldung Frostschutz, Messwert Temperatur, Messwert Temperatur Austritt Abluft, Messwert Temperatur Austritt Zuluft, Messwert Temperatur Eintritt Abluft, Messwert Temperatur Eintritt Zuluft, Messwert Vorlauftemperatur, Reset Betriebsstunden, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Stellsignal, Schaltbefehl, Sollwert Frostschutz, Sollwert Stellsignal, Sollwert Stellsignal Max, Sollwert Stellsignal Min, Stellbefehl, Störmeldung

Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Luft bereitstellen	Zuluft allgemein	Alarmmeldung, Alarmmeldung Frostschutz, Messwert Druck, Messwert Enthalpie, Messwert Feuchte, Messwert Temperatur, Messwert Volumenstrom, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Ventil, Schaltbefehl Anlage, Sollwert CO2 Konzentration, Sollwert CO2 Konzentration Max, Sollwert Druck, Sollwert Enthalpie, Sollwert Feuchte, Sollwert Feuchte Max, Sollwert Feuchte Min, Sollwert Frostschutz, Sollwert Grenzwert Soll Ist Abweichung Temperatur, Sollwert Temperatur, Sollwert Temperatur Max, Sollwert Temperatur Min, Sollwert Volumenstrom, Sollwert Volumenstrom Max, Sollwert Volumenstrom Min, Sollwert Zeitverzögerung, Störmeldung, Warnmeldung Temperatur Niedrig, Warnmeldung Feuchte, Warnmeldung Temperatur Hoch, Wartungsmeldung, Werkseinstellungen
	Zuluft-ventilator	Alarmmeldung, Anzahl Schaltungen, Befehlsausführkontrolle, Laufüberwachung, Messwert Differenzdruck, Messwert Volumenstrom, Reset Betriebsstunden, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Drehzahl, Rückmeldung Handschaltung, Rückmeldung Laufüberwachung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Reperaturschalter, Rückmeldung Stellsignal, Schaltbefehl, Sollwert Laufzeit, Sollwert Stellsignal, Stellbefehl, Störmeldung, Wartungsmeldung

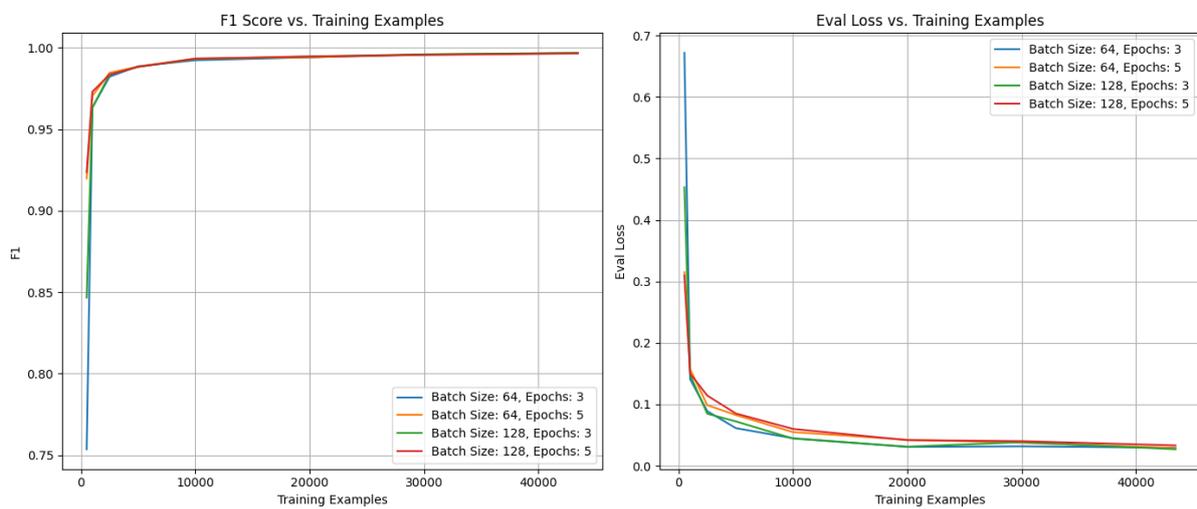
Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Luft verteilen	Auslass	Rückmeldung Handschaltung, Rückmeldung Stellsignal, Stellbefehl, Zeitverzögerung
	Raum	Alarmerückmeldung Zurück Gestellt, Alarmmeldung, Messwert CO <sub>2</sub> , Messwert Feuchte, Messwert Raumtemperatur, Rückmeldung Betrieb, Rückmeldung Fensterkontakt, Rückmeldung Handschaltung, Rückmeldung Kommunikation, Rückmeldung Managementbedieneinrichtung, Rückmeldung Präsenzmelder, Rückmeldung Trennwandkontakt, Rückmeldung Ventil, Schalthysterese, Sollwert Ausschaltverzögerung, Sollwert CO <sub>2</sub> Max, Sollwert CO <sub>2</sub> Min, Sollwert CO <sub>3</sub> , Sollwert Feuchte, Sollwert Raumtemperatur, Störmeldung, Warnmeldung Temperatur Niedrig, Warnmeldung CO <sub>2</sub> Hoch, Warnmeldung Feuchte, Warnmeldung Temperatur Hoch
	Volumenstromregler (Abluft, Zuluft)	Messwert Volumenstrom, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Rückmeldung Stellsignal, Schaltbefehl, Sollwert Volumenstrom, Stellbefehl
	Volumenstromregler Raum	Rückmeldung Handschaltung, Rückmeldung Stellsignal, Schaltbefehl, Stellbefehl
Kälte erzeugen	Kälteanlage	Anforderung, Messwert Energieverbrauch, Messwert Leistung, Rückmeldung Betrieb, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Schaltbefehl, Störmeldung
	Kältekreis allgemein	Grenzwert Außentemperatur, Messwert Außentemperatur, Messwert Druck

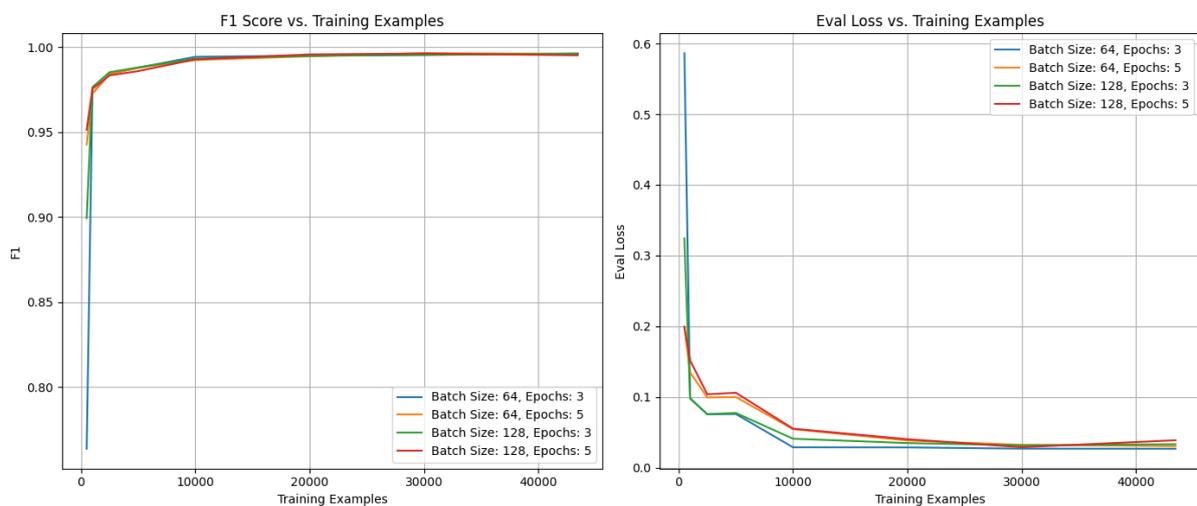
Tabelle C.1 - fortgesetzt

Unterfunktion	Komponente	Datenpunkte
Kälte erzeugen	Kältemaschine	Anforderung, Hysterese, Leistungszahl, Messwert Druck Verdampfer, Messwert Druck Verflüssiger, Messwert Kälteleistung, Messwert Stromaufnahme, Messwert Temperatur Kaltwassereintritt, Messwert Temperatur Kühlwasseraustritt, Messwert Temperatur Kühlwassereintritt, Messwert Verdampfer Austritt, Messwert Verdampfer Eintritt, Messwert Verflüssiger Austritt, Messwert Verflüssiger Eintritt, Messwert Verflüssiger Temperaturniveau, Messwert Volumenstrom, Rückmeldung Aktueller Sollwert, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Kälteleistung, Rückmeldung Managementbedieneinrichtung, Schaltbefehl, Sollwert Stellsignal, Störmeldung
	Kältemengenzähler	Messwert Kälteleistung, Messwert Kältemenge, Messwert Rücklauftemperatur, Messwert Vorlauftemperatur
	Klappe	Rückmeldung Handschaltung, Rückmeldung Klappe Auf, Rückmeldung Klappe Zu, Schaltbefehl
	Pumpe	Anforderung, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Schaltbefehl, Störmeldung
	Regler	D-Anteil, I-Anteil, P-Anteil, Sollwert, Sollwert Regler Max, Sollwert Regler Min, Stellbefehl, Totzeit
	Rückkühlwerk	Grenzwert Druck, Messwert Drehzahl, Messwert Druck, Messwert Leistung, Rückmeldung Betrieb, Rückmeldung Betriebsstunden, Rückmeldung Handschaltung, Rückmeldung Managementbedieneinrichtung, Schaltbefehl, Sollwert Stellsignal, Störmeldung
	Rücklauf	Messwert Rücklauftemperatur, Sollwert Rücklauftemperatur
	Ventil	Freigabe Sollwert, Rückmeldung Handschaltung, Rückmeldung Stellsignal, Stellbefehl
	Vorlauf	Messwert Vorlauftemperatur, Sollwert Vorlauftemperatur
	Wärmemengenzähler	Messwert Wärmemenge, Störmeldung

## D Entwicklung des F1-Score und Evaluation-Loss der Modelle

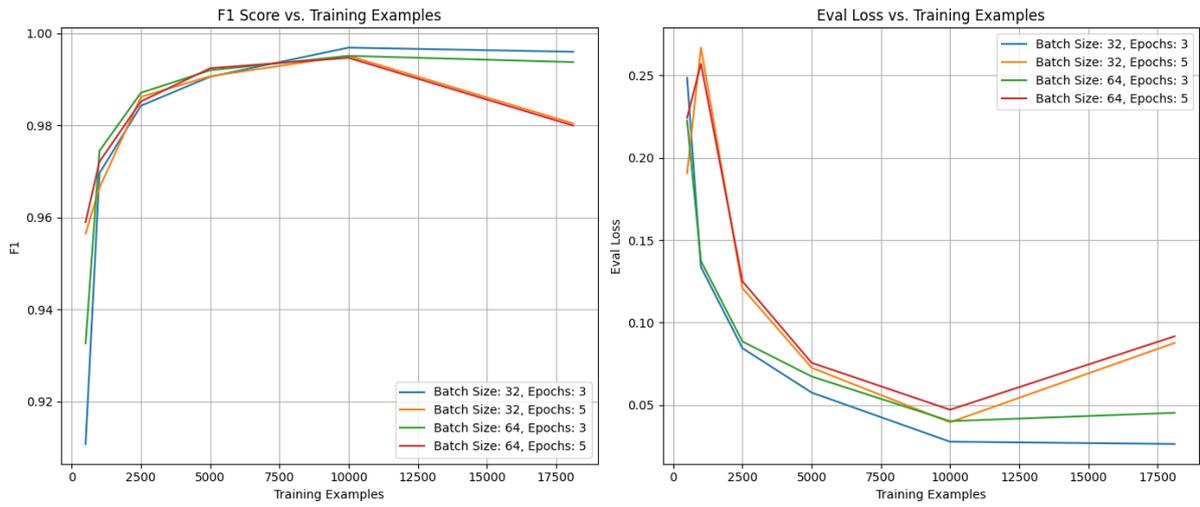


(a) F1-Score und Evaluation-Loss Grundfunktion - gBERT

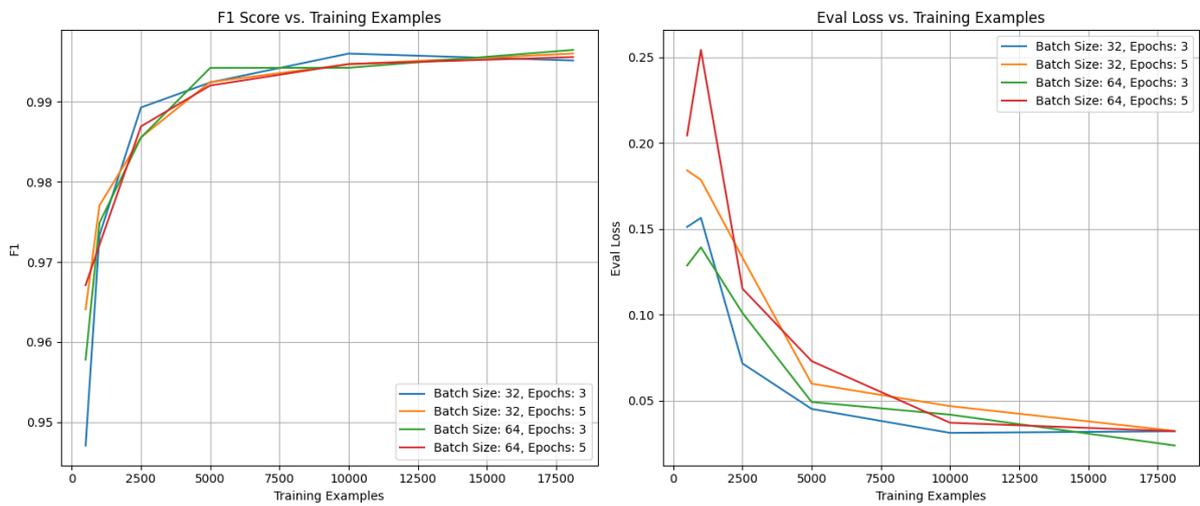


(b) F1-Score und Evaluation-Loss Grundfunktion - IDiS

Abbildung D.1: F1-Score und Evaluation-Loss der Ebene Grundfunktion

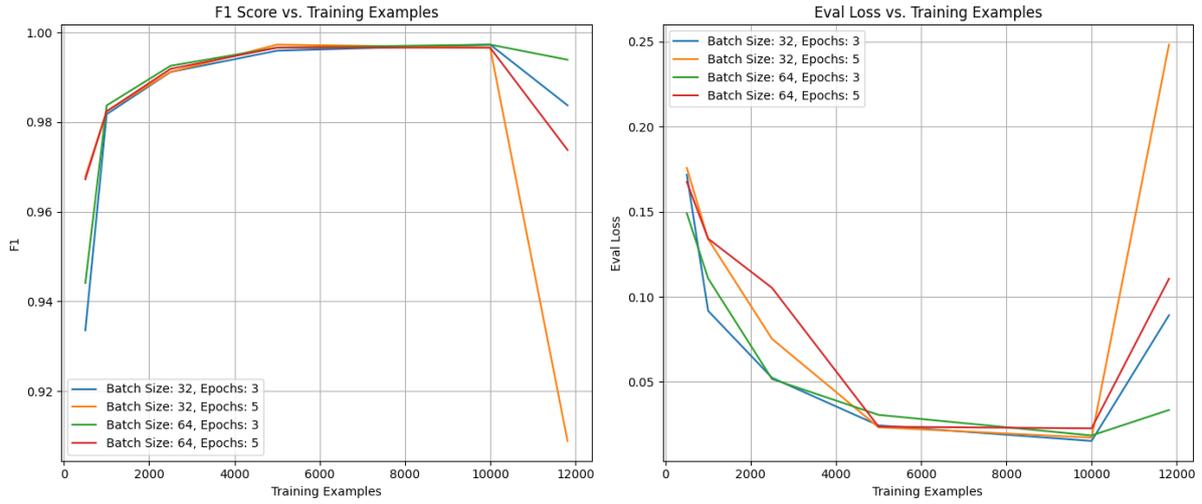


(a) F1-Score und Evaluation-Loss Wärme versorgen - gBERT

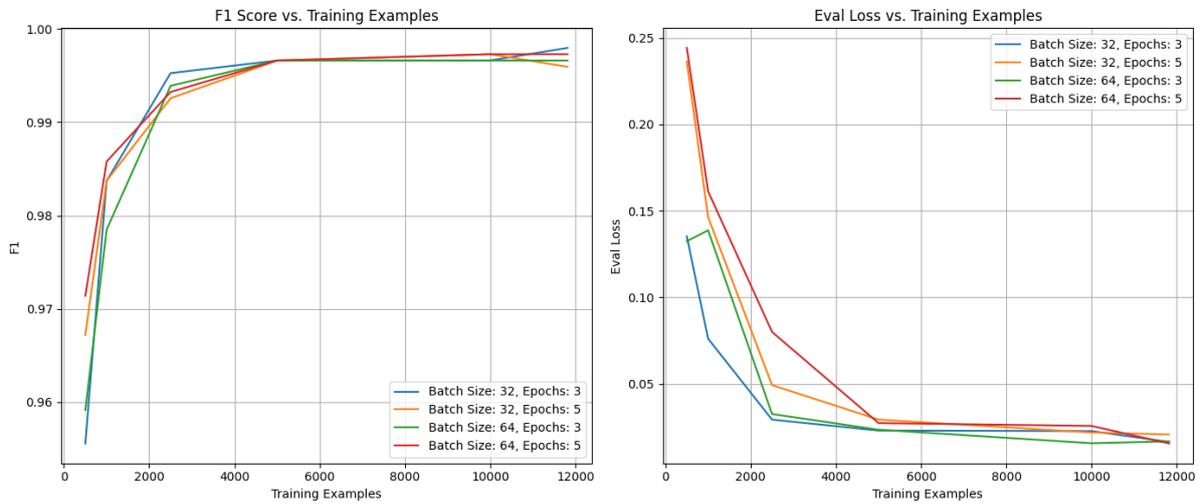


(b) F1-Score und Evaluation-Loss Wärme versorgen - IDiS

Abbildung D.2: F1-Score und Evaluation-Loss der Ebene Wärme versorgen

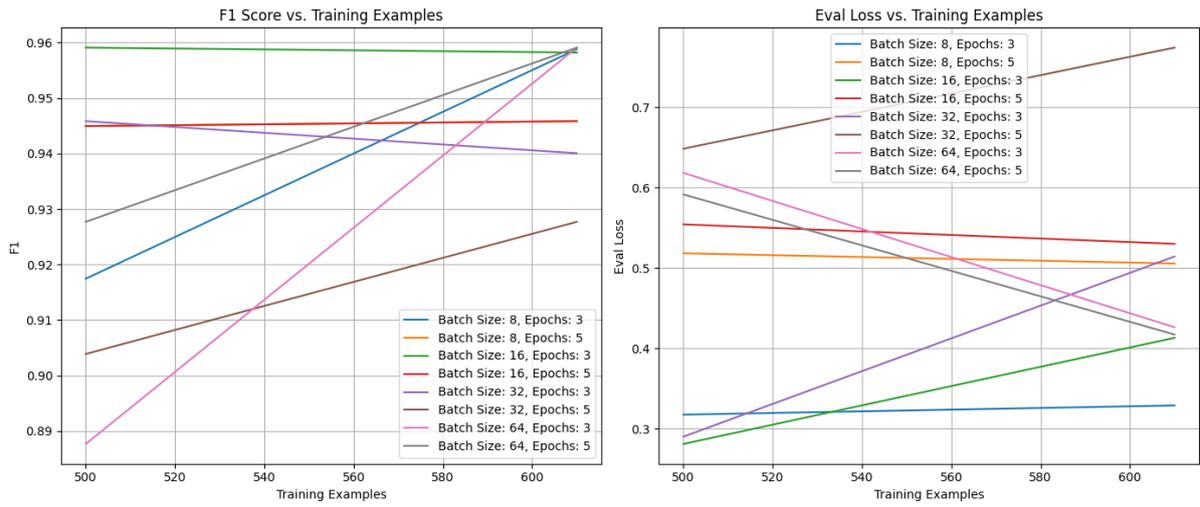


(a) F1-Score und Evaluation-Loss Luft versorgen - gBERT

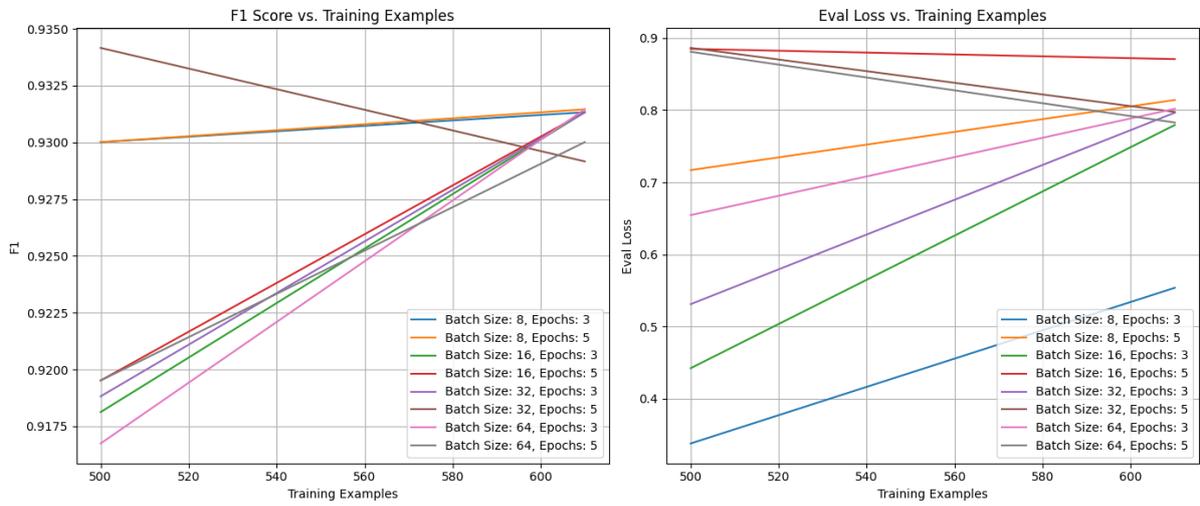


(b) F1-Score und Evaluation-Loss Luft versorgen - IDiS

Abbildung D.3: F1-Score und Evaluation-Loss der Ebene Luft versorgen

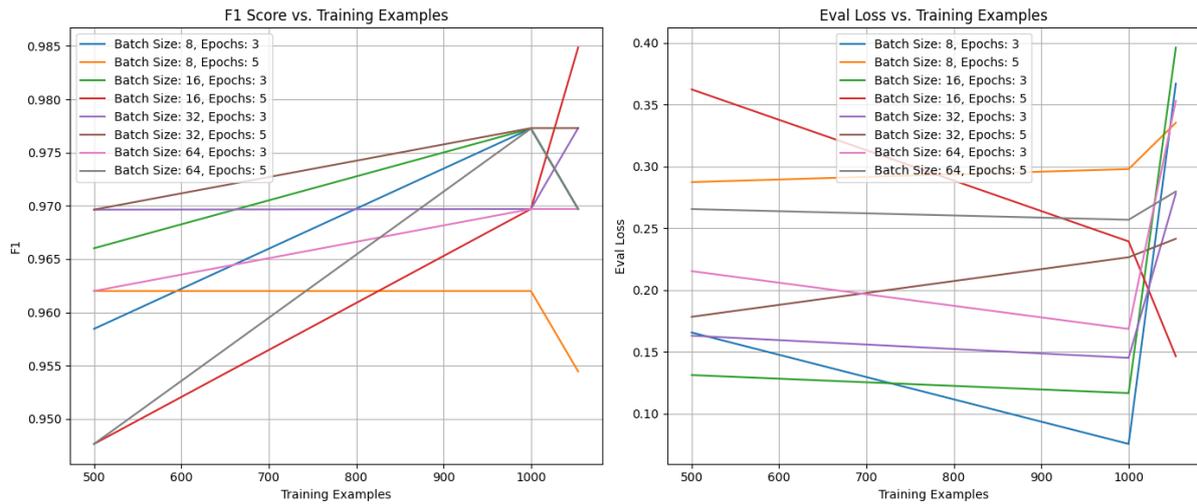


(a) F1-Score und Evaluation-Loss Medien versorgen - gBERT

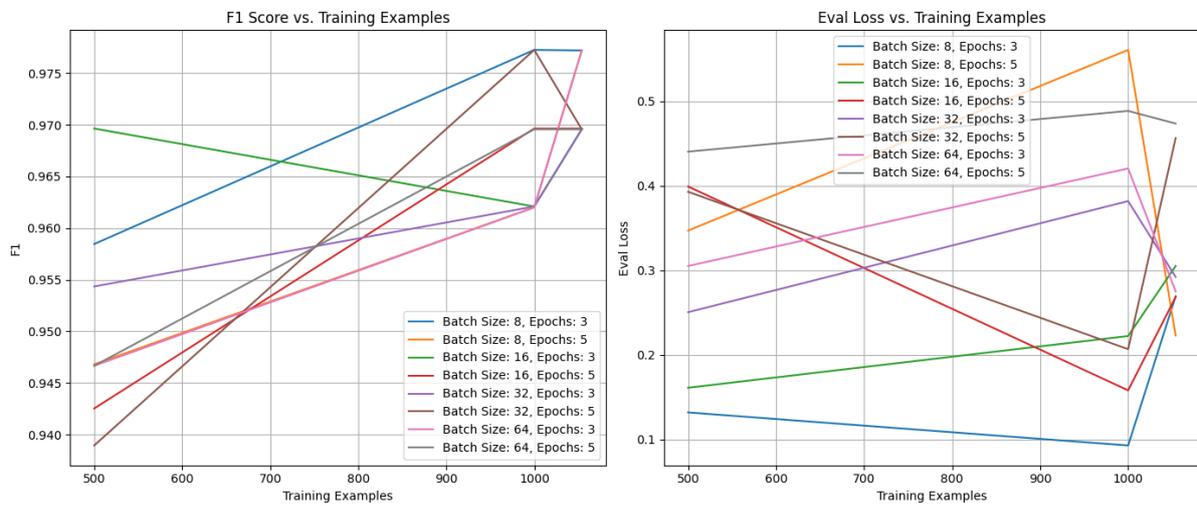


(b) F1-Score und Evaluation-Loss Medien versorgen - IDiS

Abbildung D.4: F1-Score und Evaluation-Loss der Ebene Medien versorgen

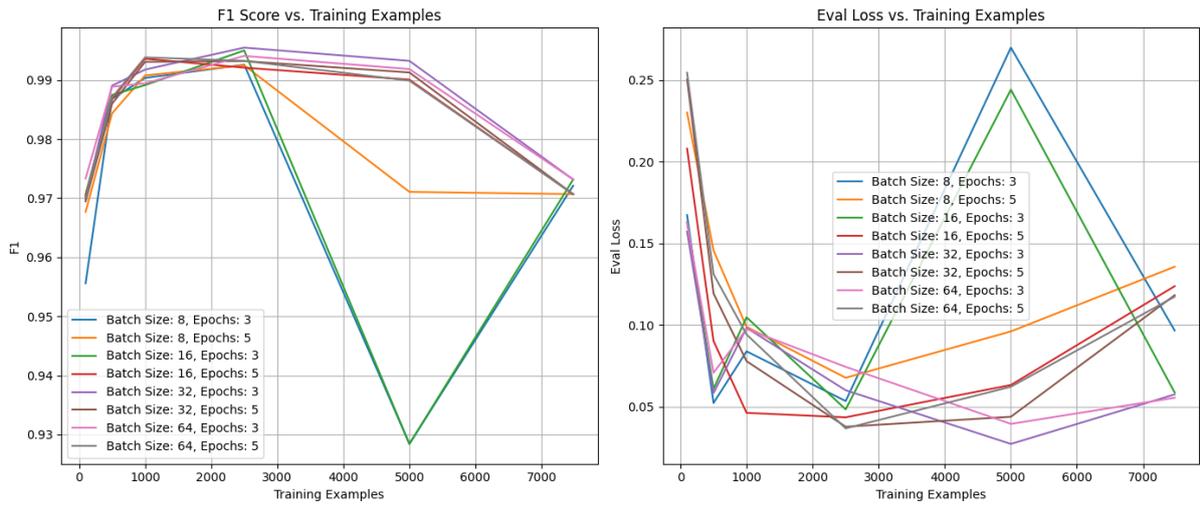


(a) F1-Score und Evaluation-Loss Kälte versorgen - gBERT

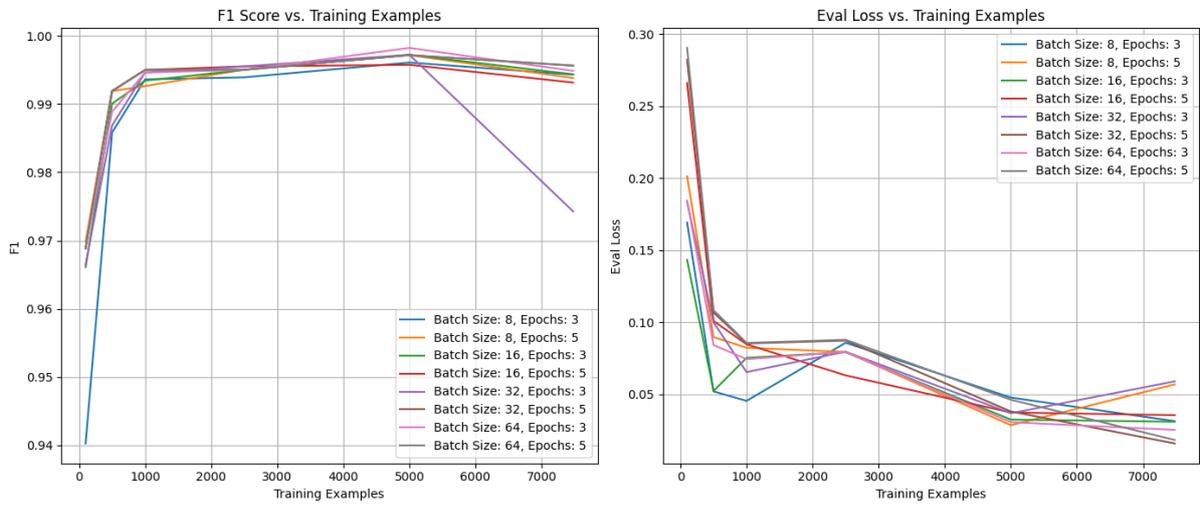


(b) F1-Score und Evaluation-Loss Kälte versorgen - IDiS

Abbildung D.5: F1-Score und Evaluation-Loss der Ebene Kälte versorgen

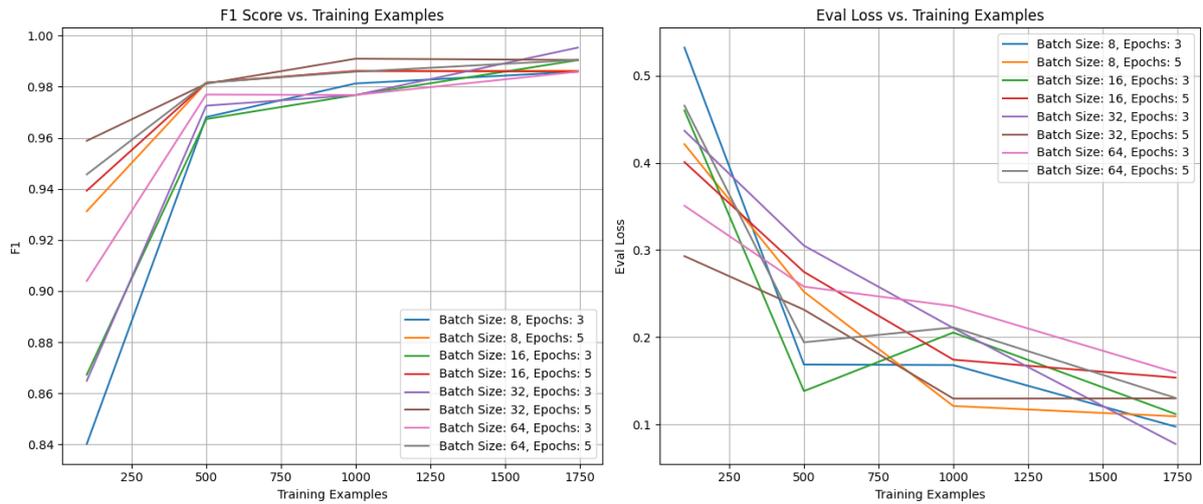


(a) F1-Score und Evaluation-Loss Sichern - gBERT

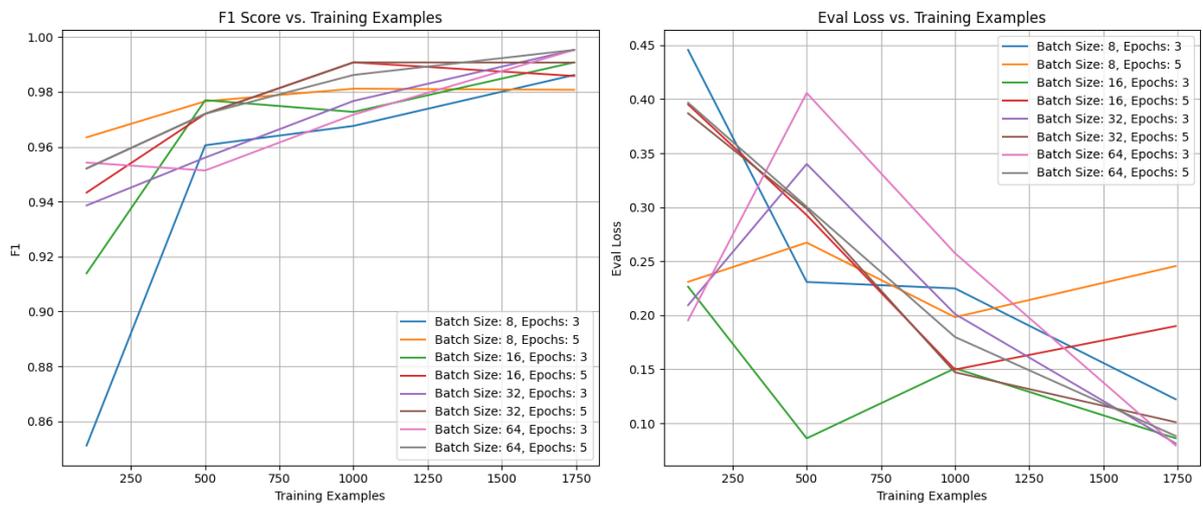


(b) F1-Score und Evaluation-Loss Sichern - IDiS

Abbildung D.6: F1-Score und Evaluation-Loss der Ebene Sichern

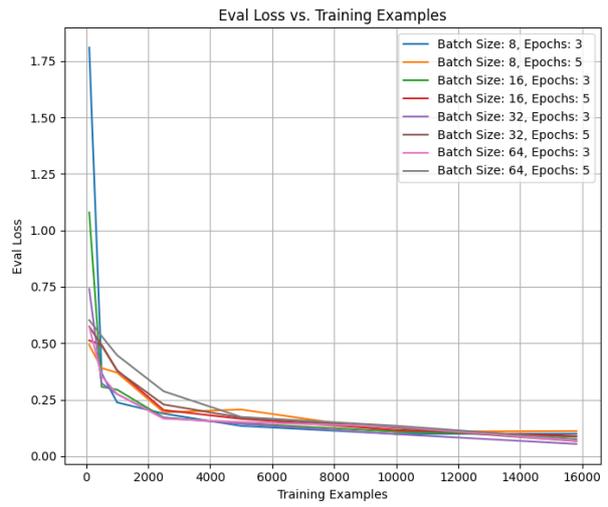
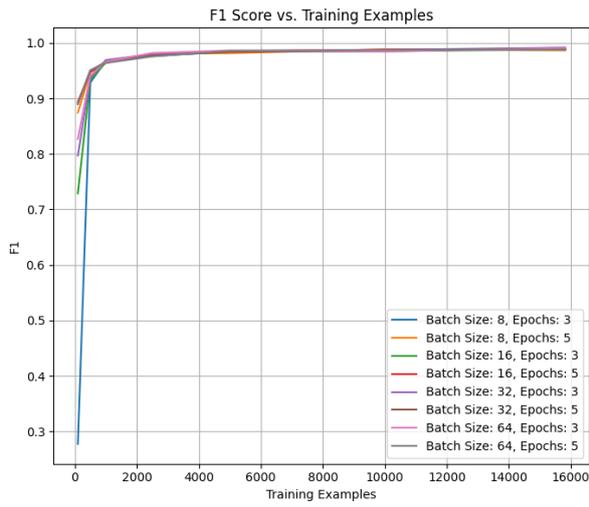


(a) F1-Score und Evaluation-Loss Wärme erzeugen - gBERT

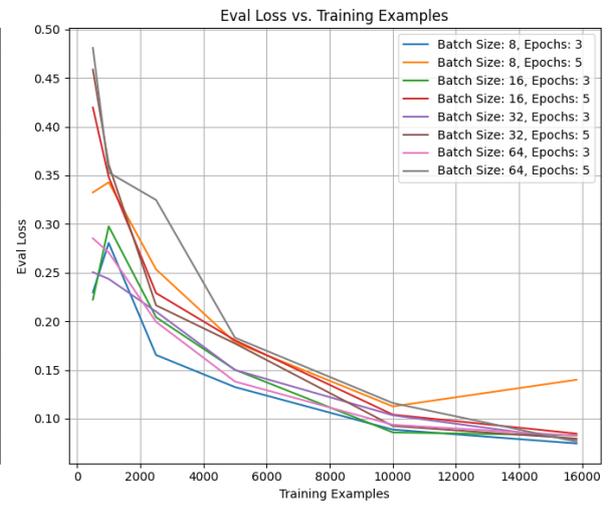
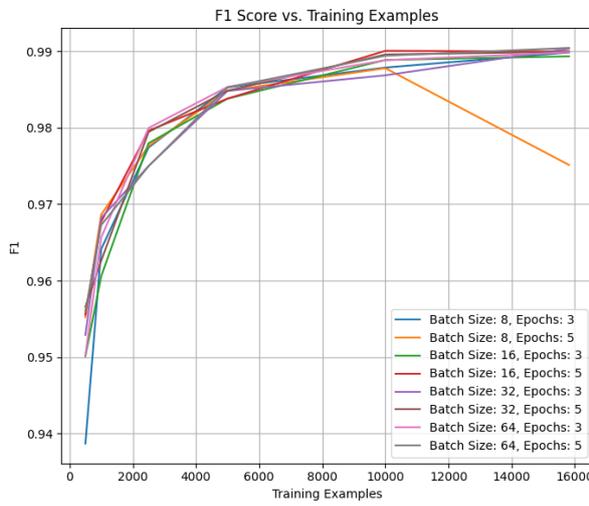


(b) F1-Score und Evaluation-Loss Wärme erzeugen - IDiS

Abbildung D.7: F1-Score und Evaluation-Loss der Ebene Wärme erzeugen

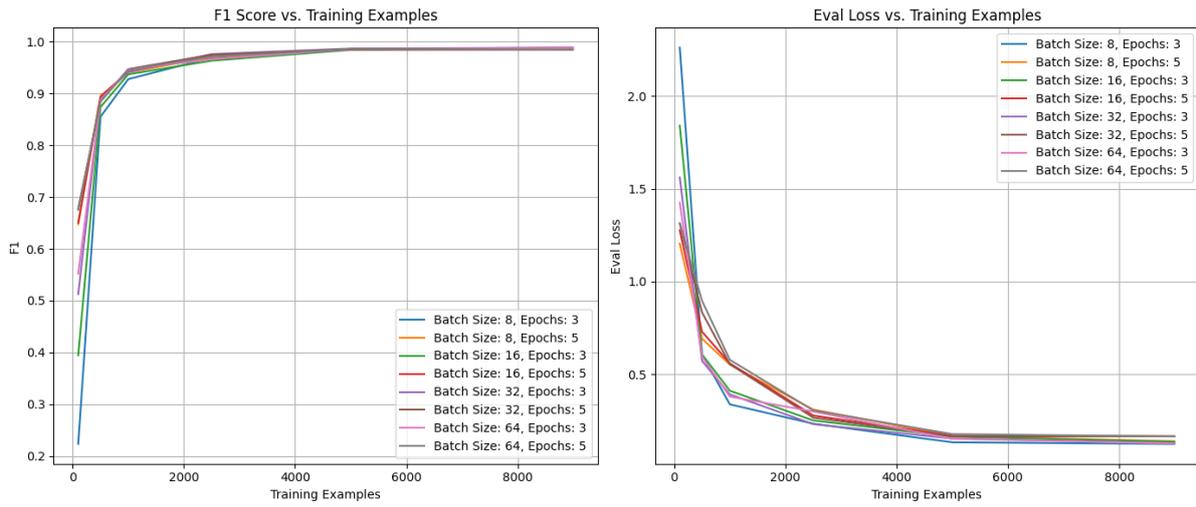


(a) F1-Score und Evaluation-Loss Wärme verteilen - gBERT

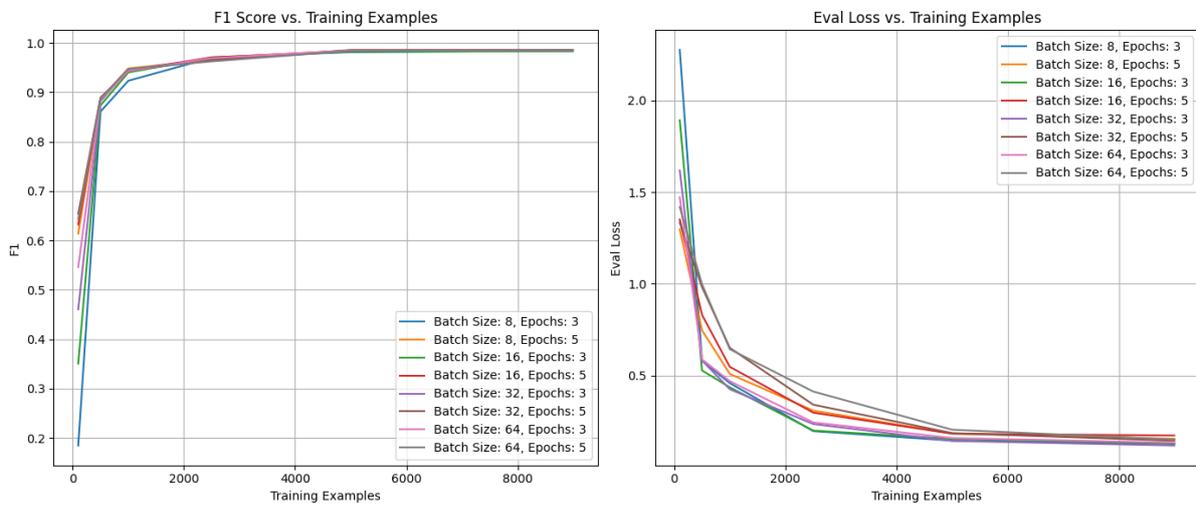


(b) F1-Score und Evaluation-Loss Wärme verteilen - IDiS

Abbildung D.8: F1-Score und Evaluation-Loss der Ebene Wärme verteilen

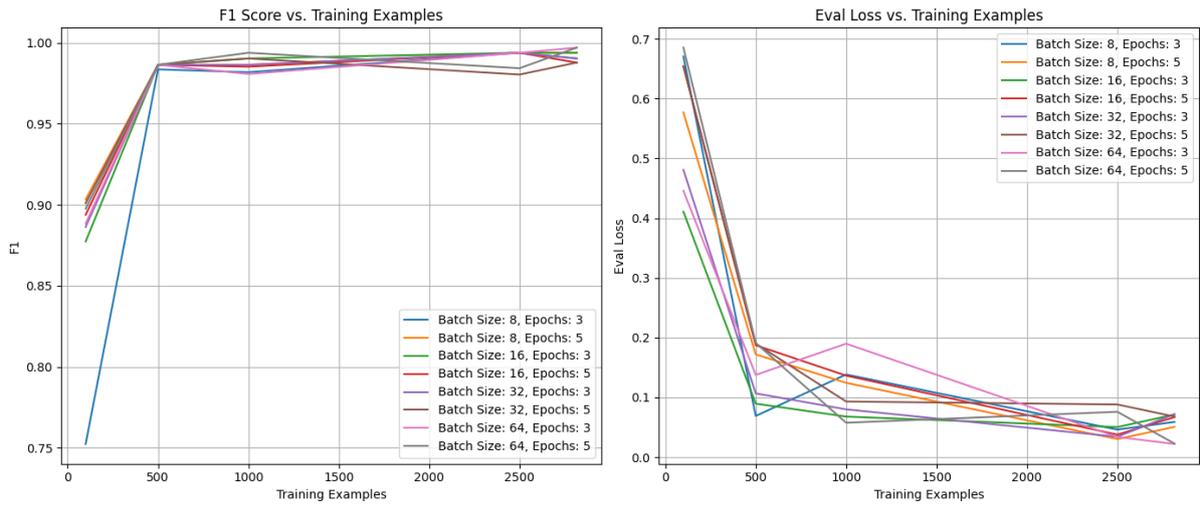


(a) F1-Score und Evaluation-Loss Luft bereitstellen - gBERT

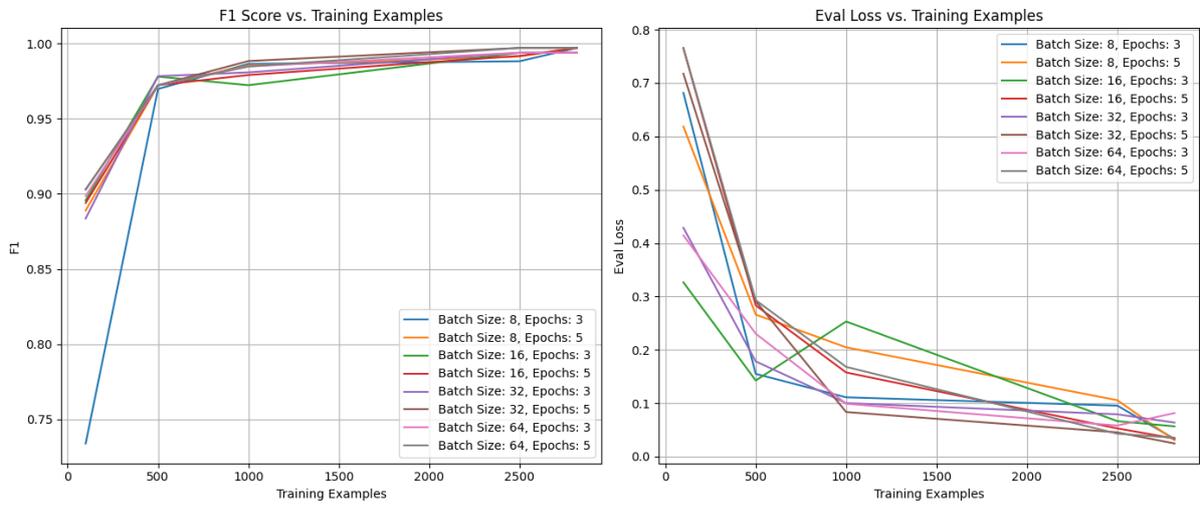


(b) F1-Score und Evaluation-Loss Luft bereitstellen - IDiS

Abbildung D.9: F1-Score und Evaluation-Loss der Ebene Luft bereitstellen

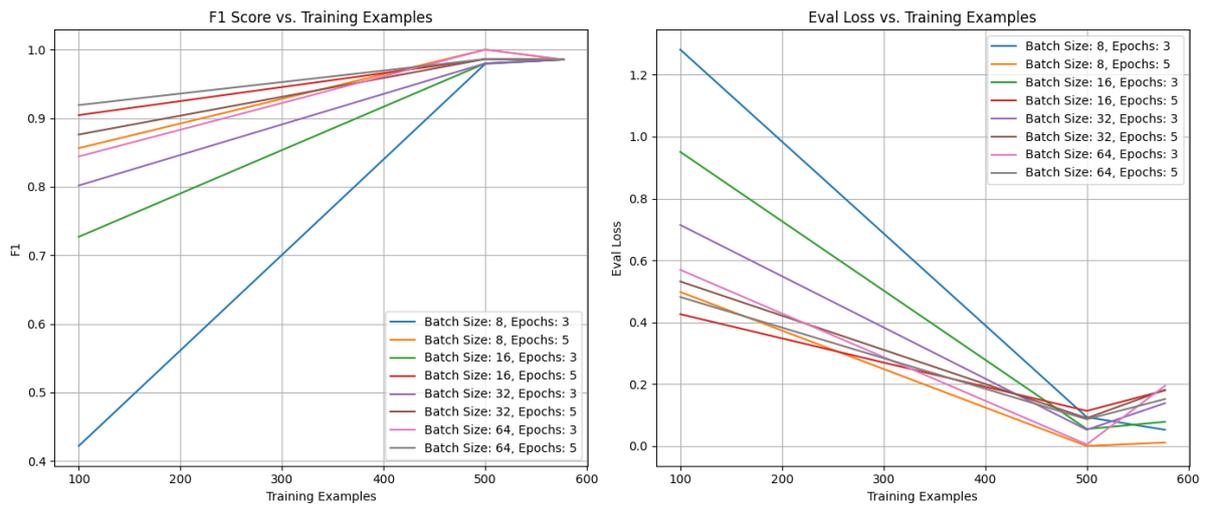


(a) F1-Score und Evaluation-Loss Luft verteilen - gBERT

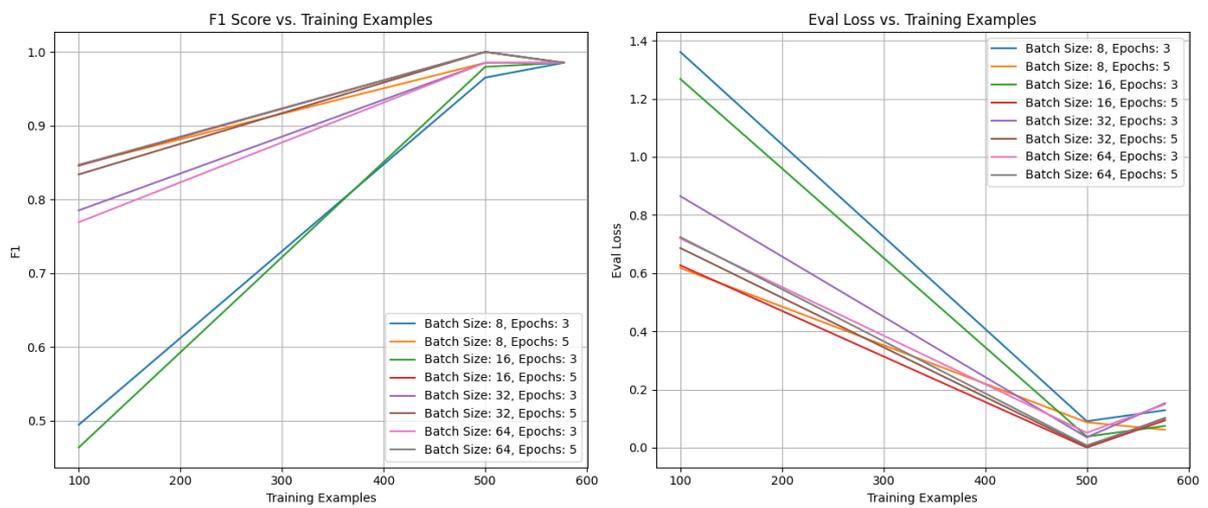


(b) F1-Score und Evaluation-Loss Luft verteilen - IDiS

Abbildung D.10: F1-Score und Evaluation-Loss der Ebene Luft verteilen

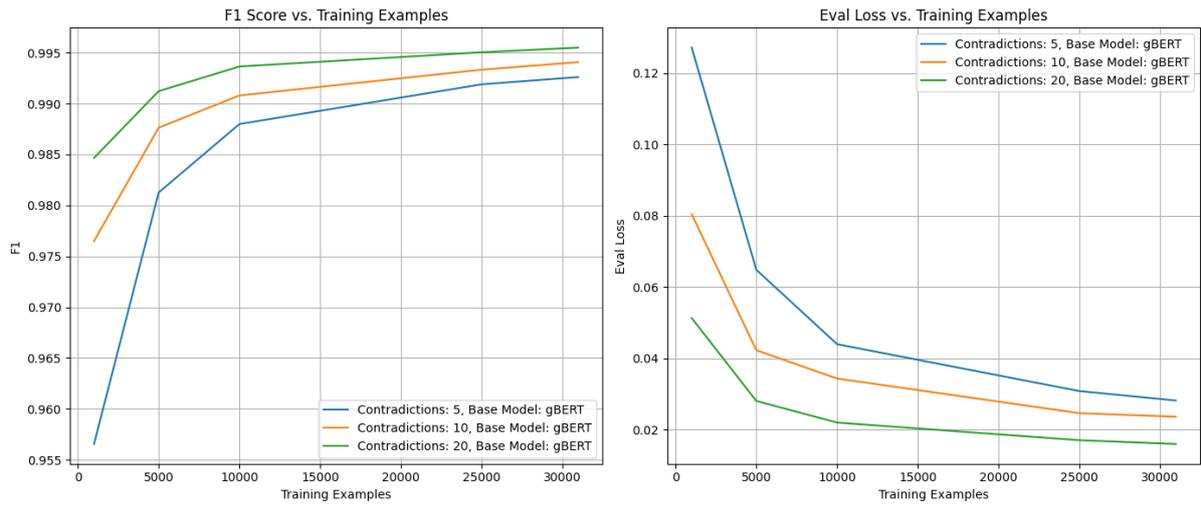


(a) F1-Score und Evaluation-Loss Kälte erzeugen - gBERT

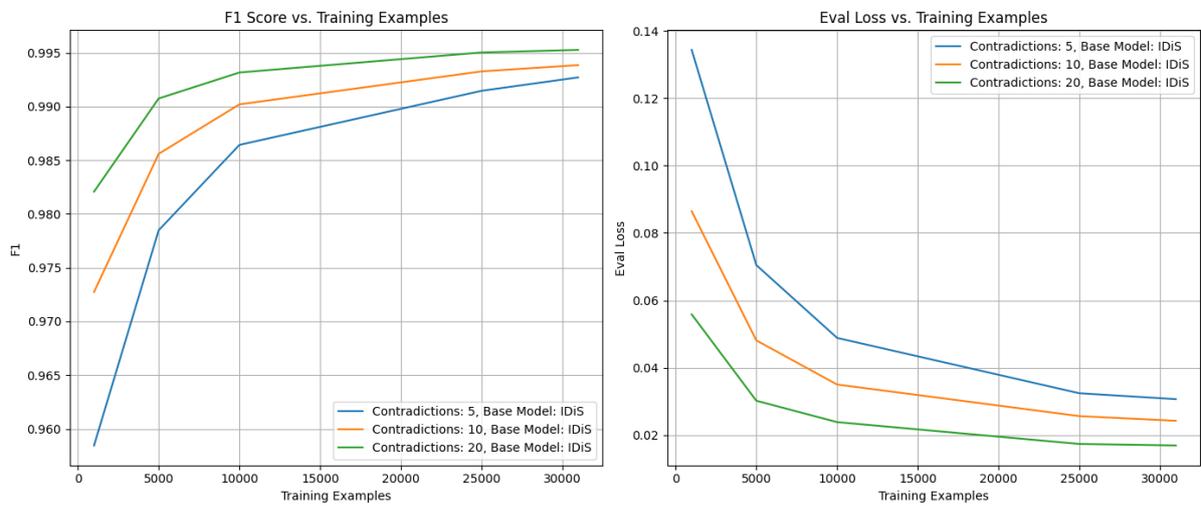


(b) F1-Score und Evaluation-Loss Kälte erzeugen - IDiS

Abbildung D.11: F1-Score und Evaluation-Loss der Ebene Kälte erzeugen



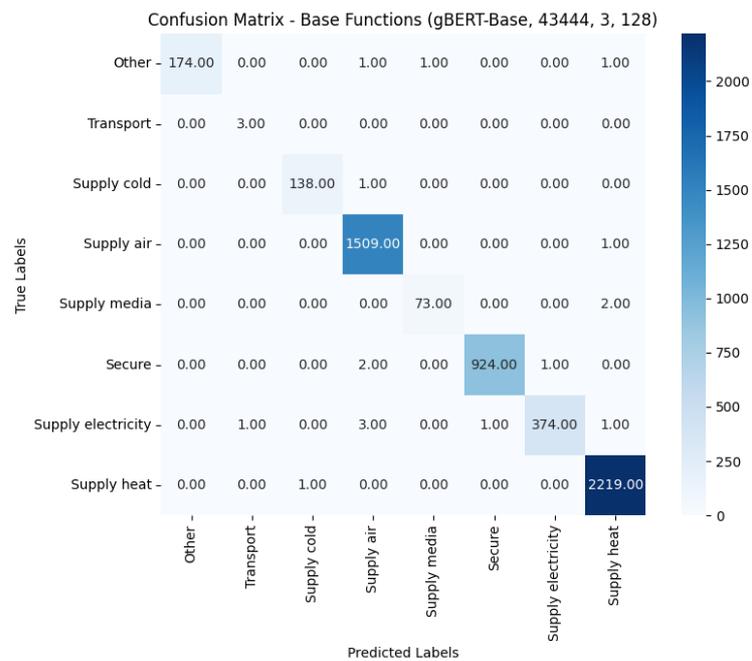
(a) F1-Score und Evaluation-Loss Datenpunkte - gBERT



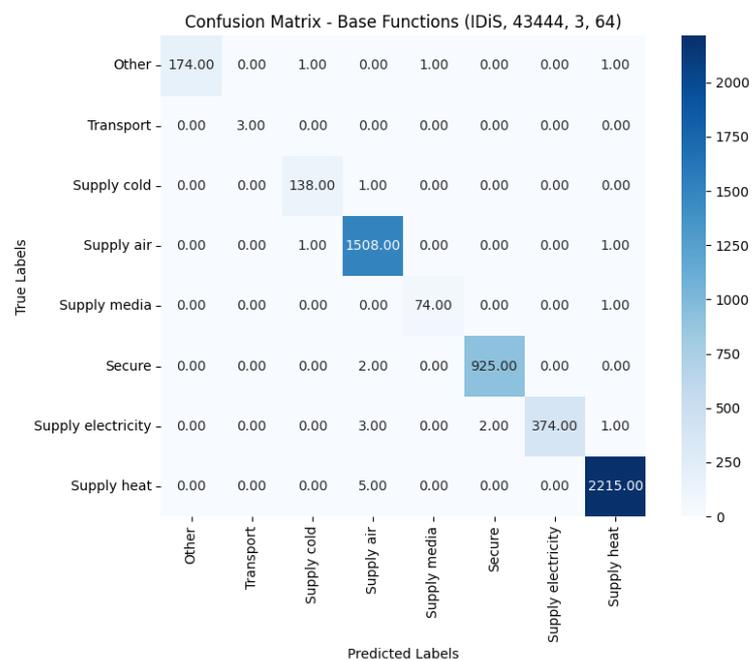
(b) F1-Score und Evaluation-Loss Datenpunkte - IDiS

Abbildung D.12: F1-Score und Evaluation-Loss der Ebene Datenpunkte

## E Wahrheitsmatrizen der besten Modelle



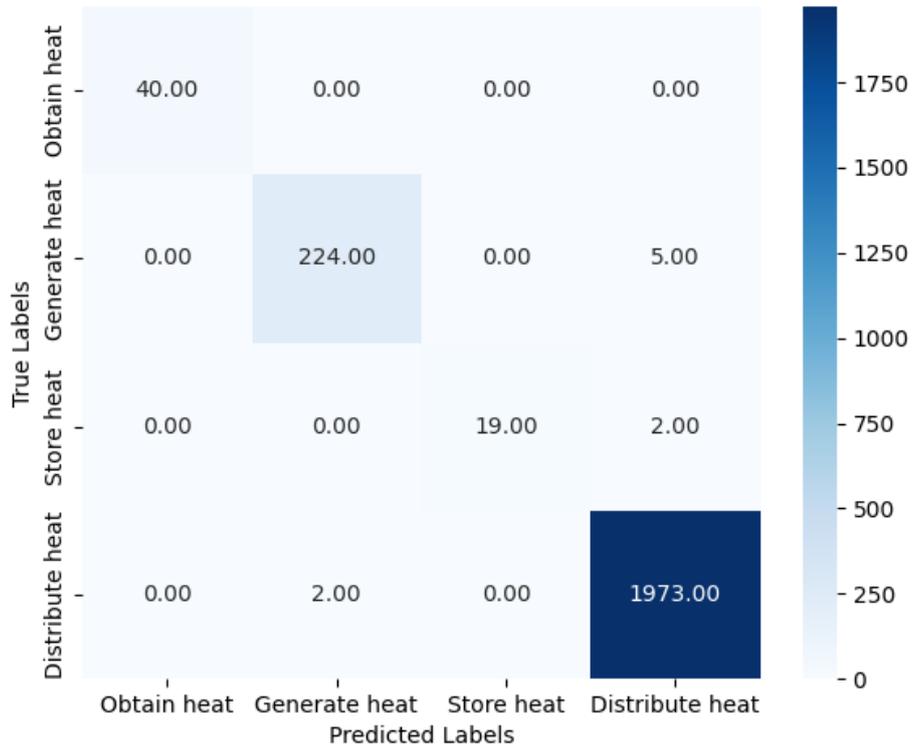
(a) Wahrheitsmatrix Grundfunktion - gBERT



(b) Wahrheitsmatrix Grundfunktion - IDiS

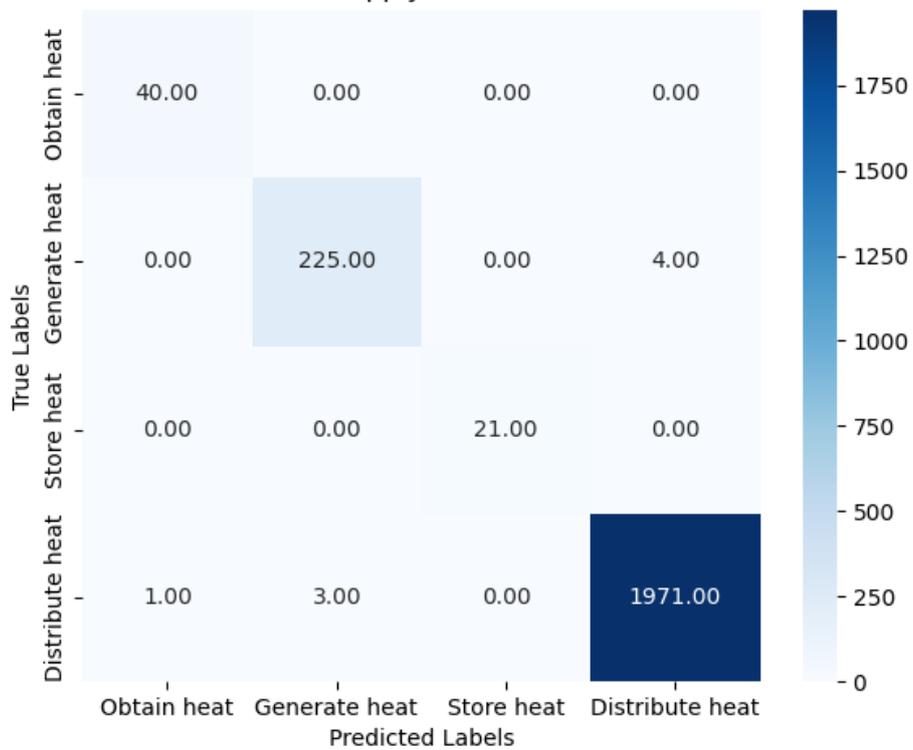
Abbildung E.1: Wahrheitsmatrizen der Ebene Grundfunktion

Confusion Matrix - Supply heat (gBERT-Base, 18116, 3, 32)



(a) Wahrheitsmatrix Wärme versorgen - gBERT

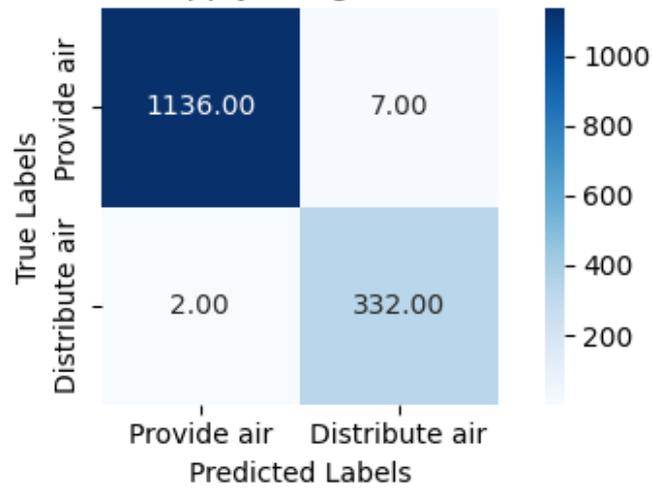
Confusion Matrix - Supply heat (IDiS, 18116, 3, 64)



(b) Wahrheitsmatrix Wärme versorgen - IDiS

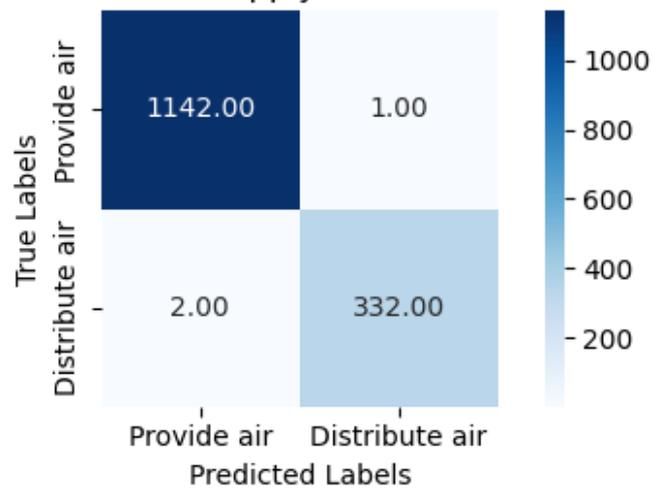
Abbildung E.2: Wahrheitsmatrizen der Ebene Wärme versorgen

Confusion Matrix - Supply air (gBERT-Base, 11816, 3, 1)



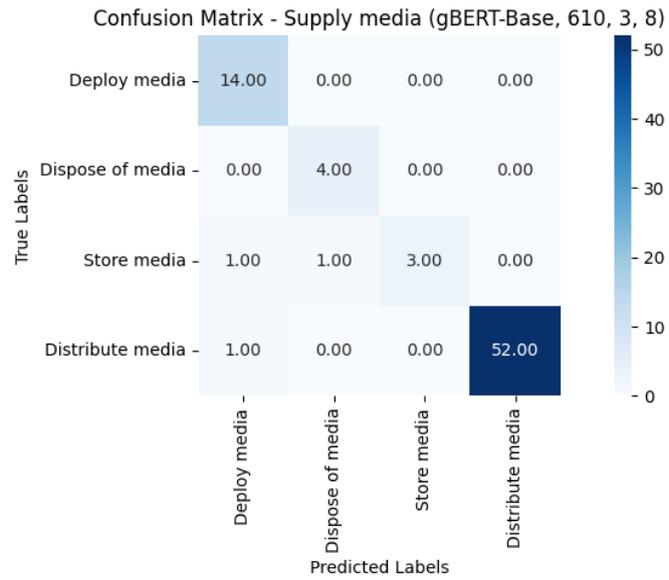
(a) Wahrheitsmatrix Luft versorgen - gBERT

Confusion Matrix - Supply air (IDiS, 11816, 3, 32)

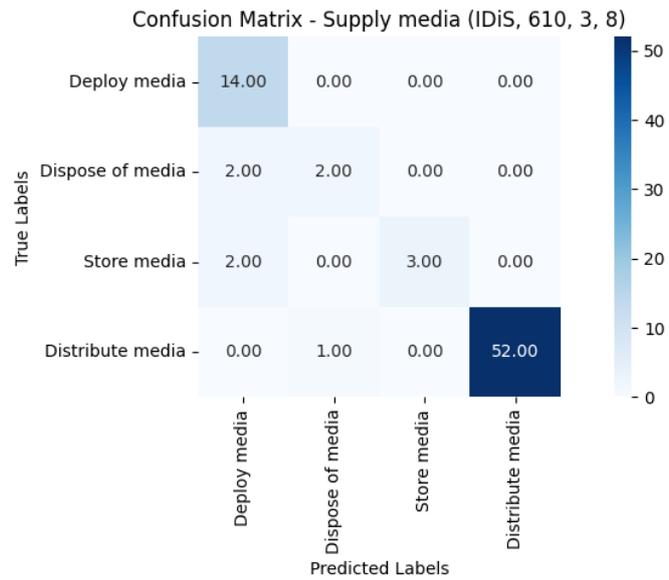


(b) Wahrheitsmatrix Luft versorgen - IDiS

Abbildung E.3: Wahrheitsmatrizen der Ebene Luft versorgen

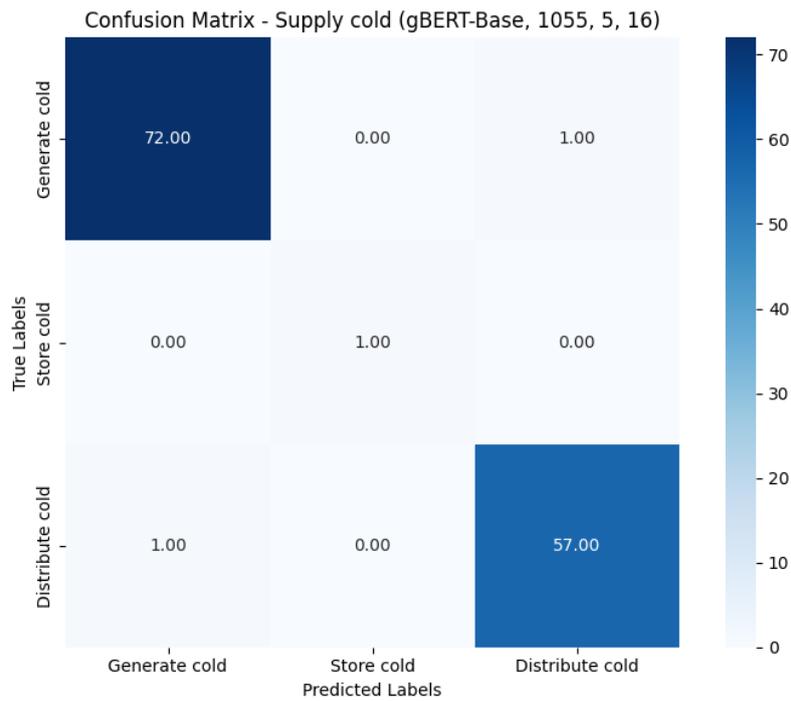


(a) Wahrheitsmatrix Medien versorgen - gBERT

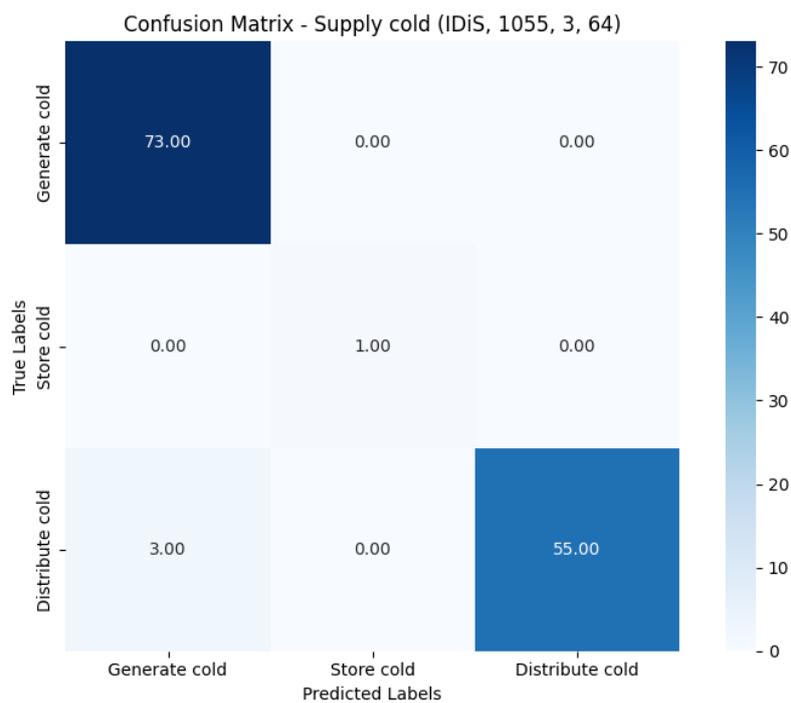


(b) Wahrheitsmatrix Medien versorgen - IDiS

Abbildung E.4: Wahrheitsmatrizen der Ebene Medien versorgen

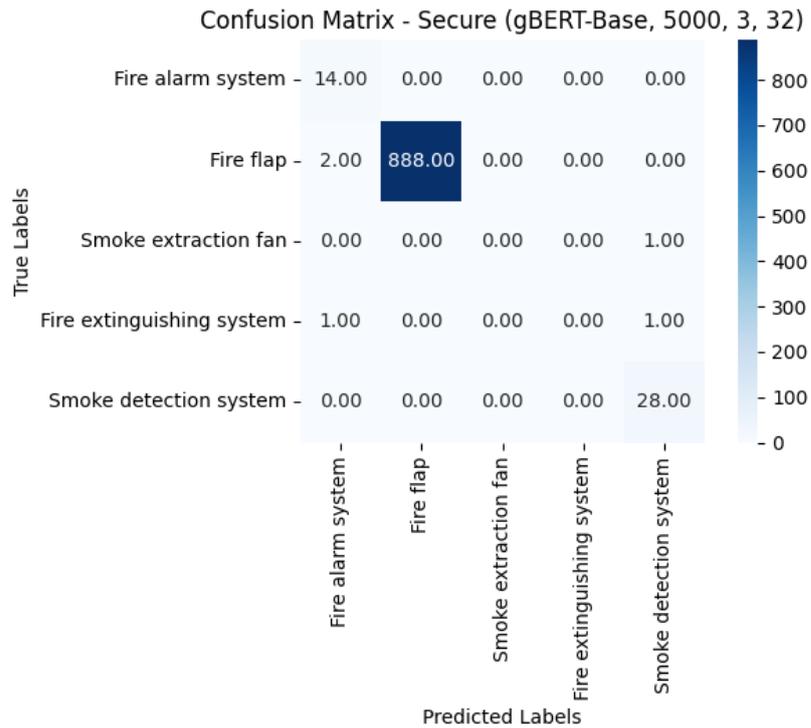


(a) Wahrheitsmatrix Kälte versorgen - gBERT

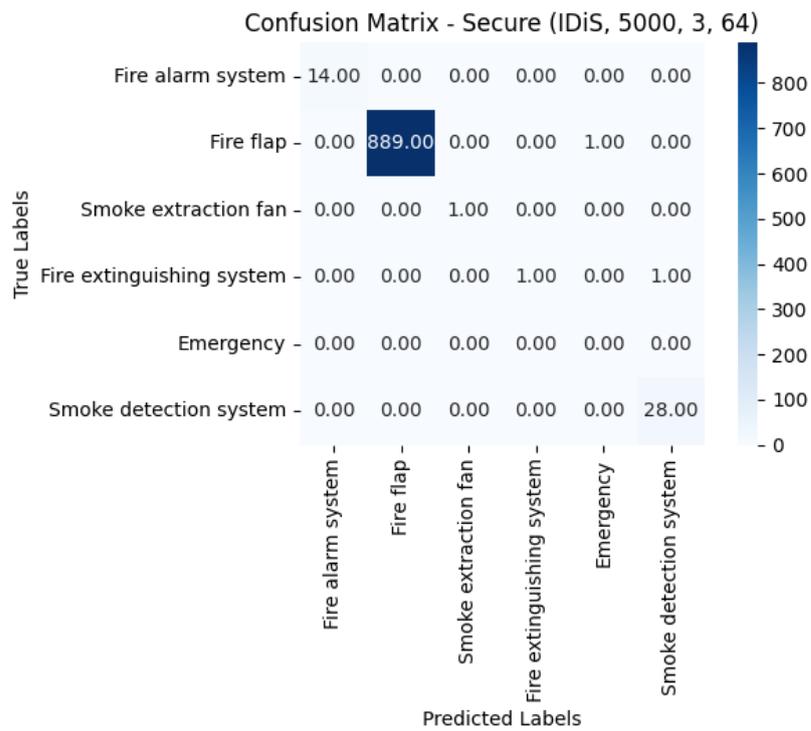


(b) Wahrheitsmatrix Kälte versorgen - IDiS

Abbildung E.5: Wahrheitsmatrizen der Ebene Kälte versorgen

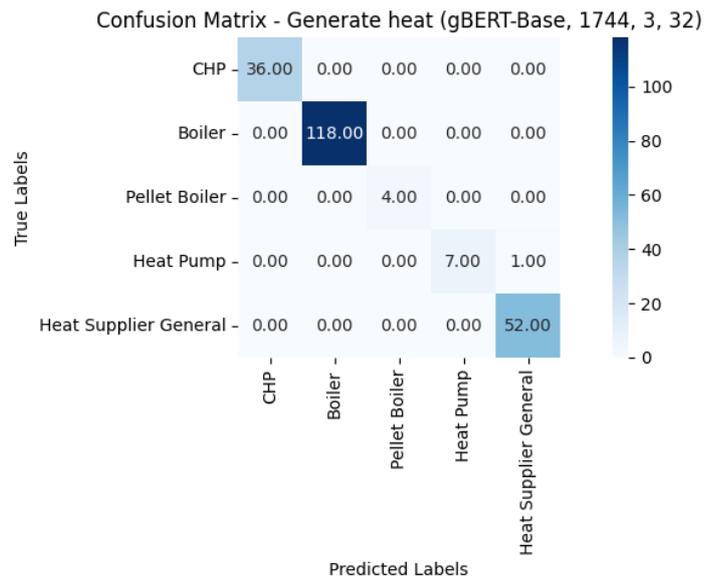


(a) Wahrheitsmatrix Sichern - gBERT

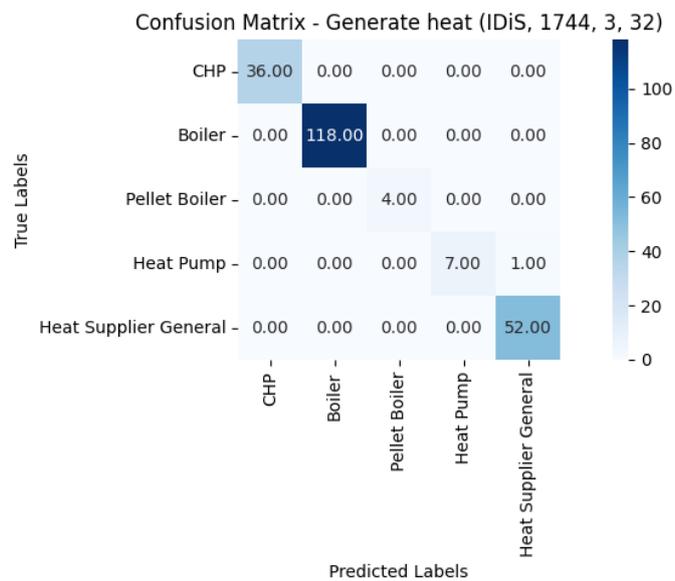


(b) Wahrheitsmatrix Sichern - IDiS

Abbildung E.6: Wahrheitsmatrizen der Ebene Sichern

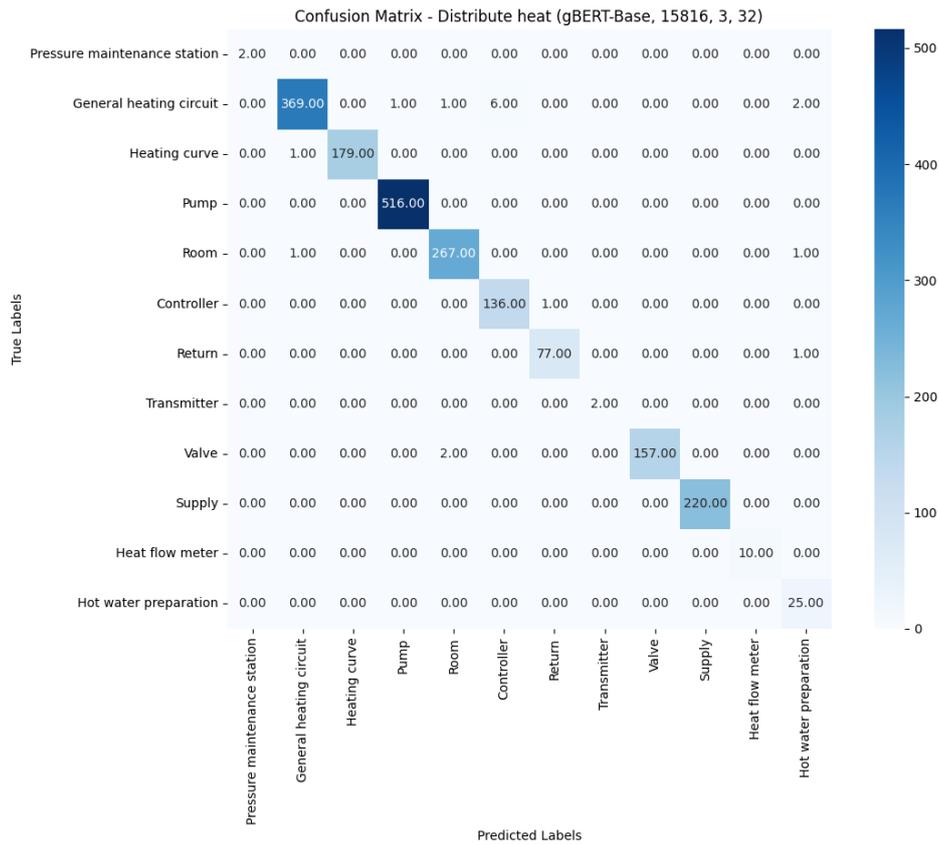


(a) Wahrheitsmatrix Wärme erzeugen - gBERT

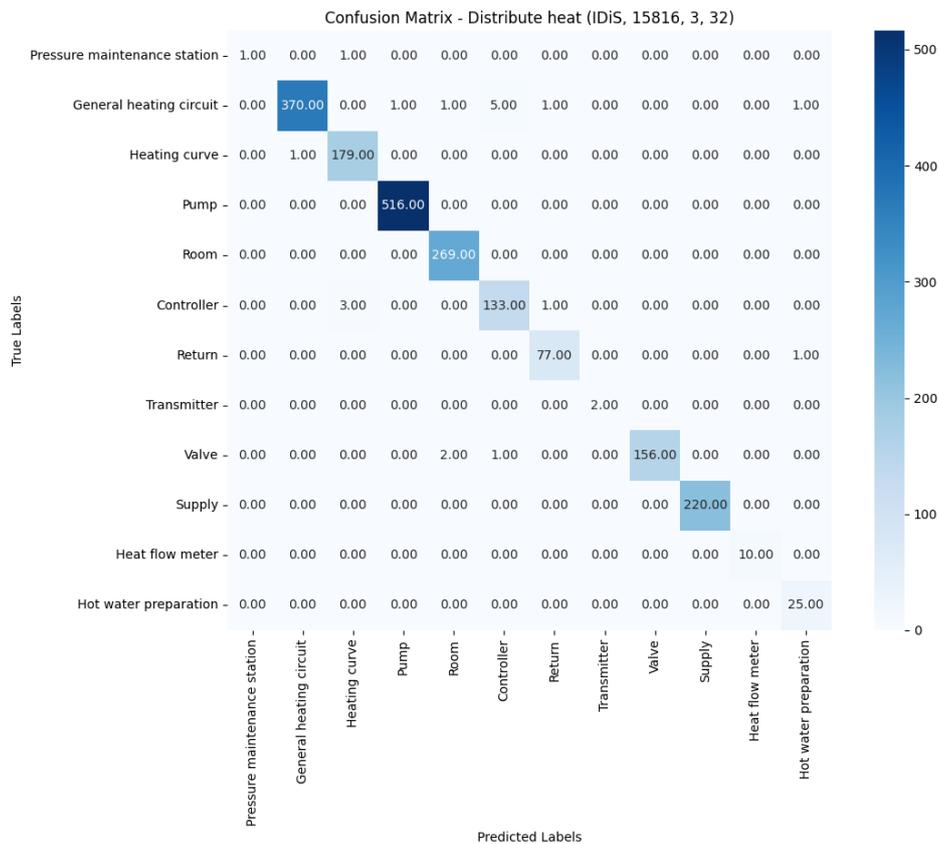


(b) Wahrheitsmatrix Wärme erzeugen - IDiS

Abbildung E.7: Wahrheitsmatrizen der Ebene Wärme erzeugen



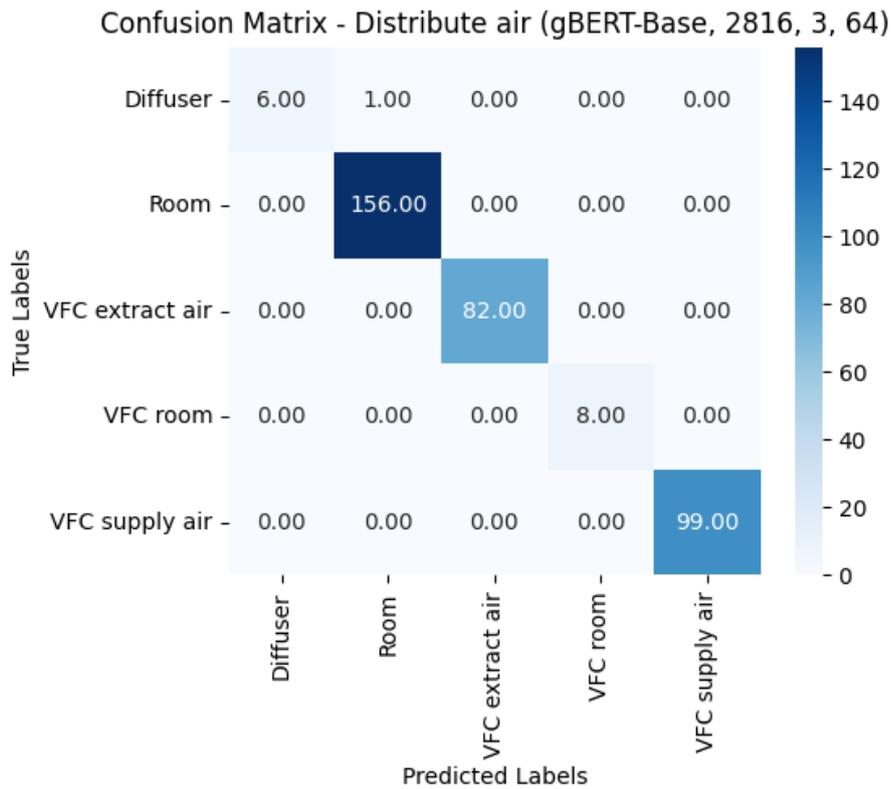
(a) Wahrheitsmatrix Wärme verteilen - gBERT



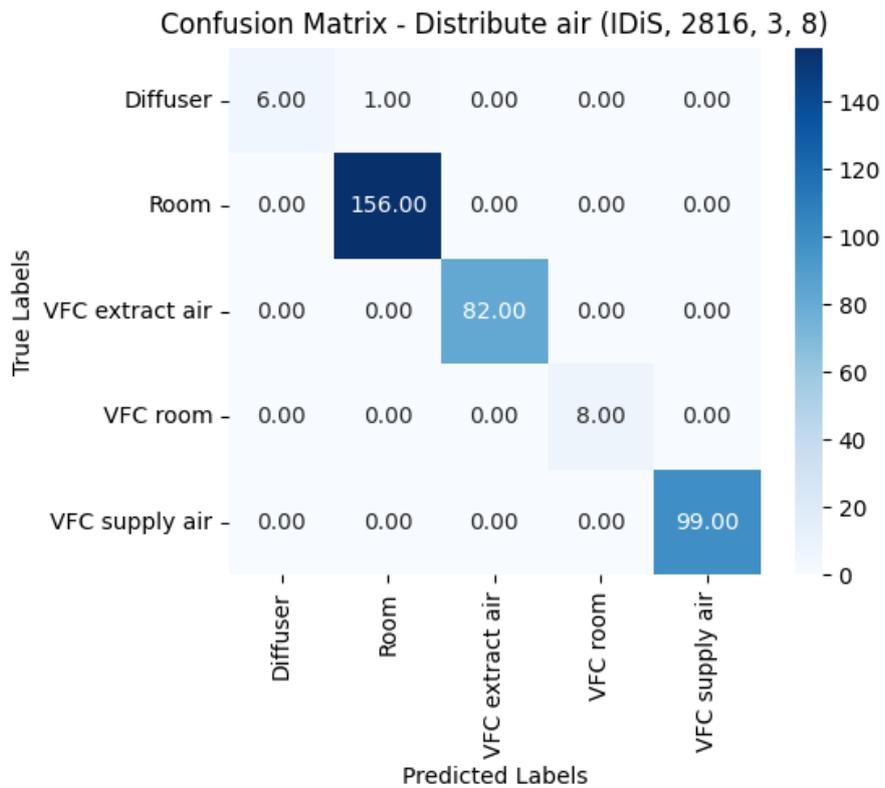
(b) Wahrheitsmatrix Wärme verteilen - IDiS

Abbildung E.8: Wahrheitsmatrizen der Ebene Wärme verteilen



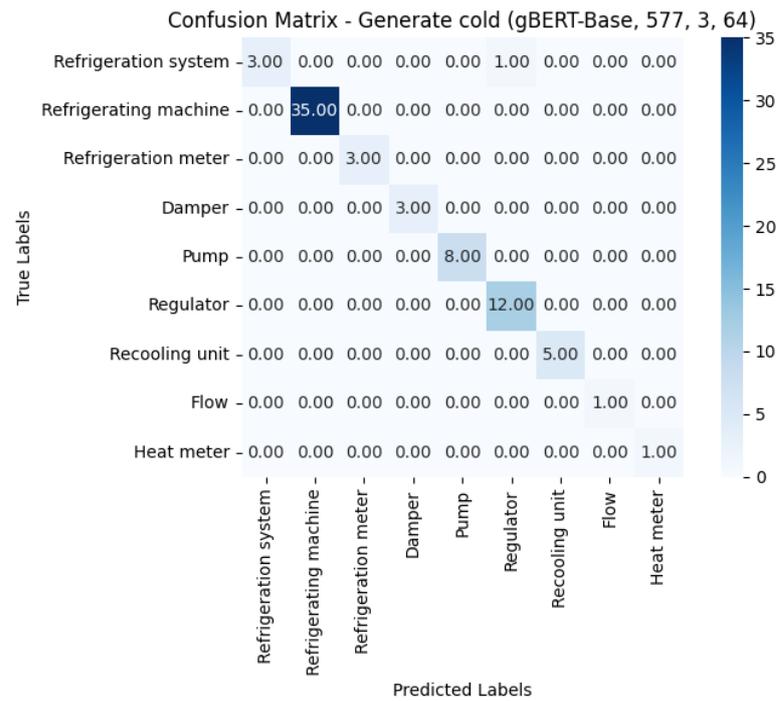


(a) Wahrheitsmatrix Luft verteilen - gBERT

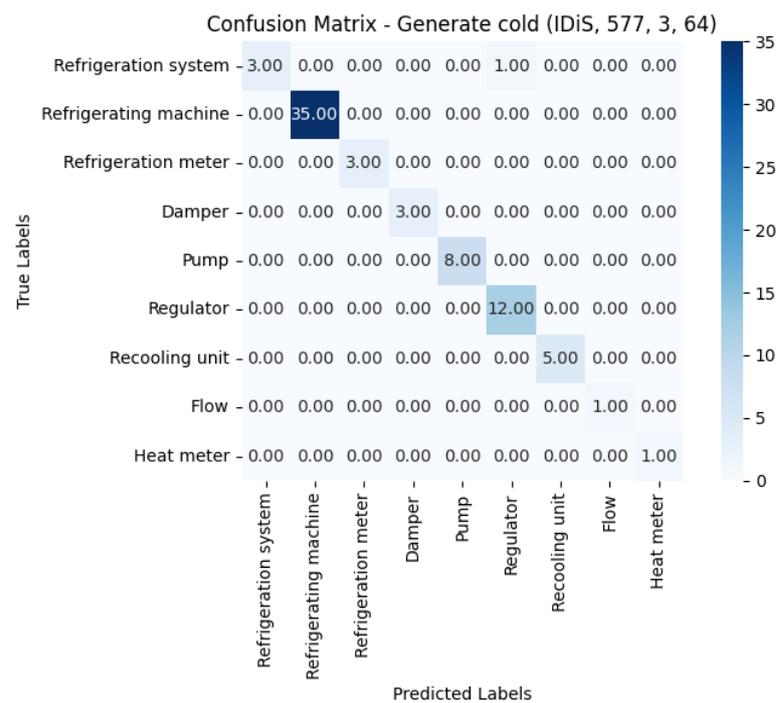


(b) Wahrheitsmatrix Luft verteilen - IDiS

Abbildung E.10: Wahrheitsmatrizen der Ebene Luft verteilen



(a) Wahrheitsmatrix Kälte erzeugen - gBERT



(b) Wahrheitsmatrix Kälte erzeugen - IDiS

Abbildung E.11: Wahrheitsmatrizen der Ebene Kälte erzeugen