

Tools des Maschinellen Lernens: Marktstudie, Anwendungsbereiche & Lösungen der Künstlichen Intelligenz

Marcus Grum, Eldar Sultanow, Daniel Friedmann, André Ullrich, Norbert Gronau

Vorwort

Künstliche Intelligenz ist in aller Munde. Immer mehr Anwendungsbereiche werden durch die Auswertung von vorliegenden Daten mit Algorithmen und Frameworks z.B. des Maschinellen Lernens erschlossen. Unter Maschinellern Lernen (ML) verstehen die Autoren die künstliche Generierung von Wissen durch Lernen auf Basis von Beispielen in Daten, Informationen und Erfahrung. Dabei werden Algorithmen – also mathematische Vorgehensweisen zur Lösung von Problemen – angewandt, um statische Modelle zur Diagnose, Analyse, Klassifikation oder Spracherkennung aufzubauen.

Das besondere beim ML ist, dass die ML-Werkzeuge ihr Verhalten flexibel an Umgebungsänderungen und Eingabedaten anpassen können. Diese Eigenschaft macht die Anwendung des Maschinellen Lernens für viele praktische Aufgaben äußerst wertvoll.

Dieses Buch hat das Ziel, einen Überblick über gegenwärtig vorhandene Lösungen zu geben und darüber hinaus konkrete Hilfestellung bei der Auswahl von Algorithmen oder Tools bei spezifischen Problemstellungen zu bieten. Um diesem Anspruch gerecht zu werden, wurden 89 Lösungen mittels einer systematischen Literaturrecherche und Praxissuche identifiziert sowie anschließend klassifiziert.

Dieses Werk gliedert und erläutert die verschiedenen Anwendungsbereiche des Maschinellen Lernens. Hervorzuheben ist, dass die Inhalte dieses Buches gleichermaßen für Praxis und Forschung tauglich sind. Besonderen Wert erlangt die vorliegende Schrift durch ihre gründliche und aktuelle Marktübersicht vorhandener Werkzeuge für das Maschinelle Lernen, die einen großen Teil des Inhalts ausmacht.

Mit Hilfe dieses Buches gelingt es, schnell die notwendigen Grundlagen zu verstehen, gängige Anwendungsgebiete zu verstehen und den Prozess zur Auswahl eines passenden ML-Tools für das eigene Projekt systematisch zu meistern.

Univ.-Prof. Dr.-Ing. habil. Norbert Gronau

Marcus Grum



M. Sc. mult. Marcus Grum studierte Wirtschaftsinformatik an der Hochschule für Wirtschaft und Recht Berlin. Seinen M. Sc. erwarb er 2014 an der Technischen Universität Berlin in Informatik mit dem Schwerpunkt auf der Wissenschaft der Intelligenz. Einen weiteren M. Sc. erwarb er 2016 in Betriebswirtschaftslehre an der Universität Potsdam. Derzeit promoviert er am Lehrstuhl für Wirtschaftsinformatik, Prozesse und Systeme. Sein Hauptforschungsinteresse gilt neuronalen Netzen und der Wissensverarbeitung. Dies schließt die Integration von künstlicher Intelligenz in wirtschaftliche Zusammenhänge mit ein.

Eldar Sultanow



Dr. Eldar Sultanow ist Architekt bei Capgemini. Er hat 2005 am Hasso-Plattner-Institut in Potsdam seinen B. Sc. in Software Engineering erlangt. Nach seinem Diplom-Abschluss in Informatik an der Uni Potsdam promovierte er am Lehrstuhl für Wirtschaftsinformatik. Seine Schwerpunkte sind Künstliche Intelligenz, moderne Softwarearchitekturen und Unternehmensarchitekturmanagement.

Daniel Friedmann



Daniel Friedmann erlangte Ende 2019 an der Technischen Hochschule Nürnberg Georg Simon Ohm seinen B. Sc. in Medieninformatik. Derzeit studiert er an dieser Hochschule im Masterstudien-gang Informatik und arbeitete bis zuletzt als Werkstudent bei Capgemini. Im Zuge seiner Masterarbeit ist er als Masterand bei Capgemini tätig. Seine Interessen liegen in den Bereichen Cloud, Container, Microservices und Maschinelles Lernen.

André Ullrich



André Ullrich studierte Betriebswirtschaftslehre mit den Schwerpunkten Wirtschaftsinformatik und Finanzierung & Banken an der Universität Potsdam und der Finanzakademie Moskau. 2018 promovierte er zu „Eigenschaften wandlungsfähiger Systeme“ an der Universität Potsdam, wo er von 2011-2017 als wissenschaftlicher Mitarbeiter arbeitete. Von 2017-2018 war er am Institut für Wirtschaftsinformatik und Digitale Gesellschaft e.V. beschäftigt. Seit 2018 arbeitet er im Rahmen der Nachwuchsforschergruppe ProMUT als Post-Doktorand am Lehrstuhl für Wirtschaftsinformatik, Prozesse und Systeme an der Universität Potsdam. Aktuelle Foki seiner wissenschaftlichen Arbeit sind nachhaltige Digitalisierung, deren Wirkung auf Unternehmensarchitekturen, organisationales Wissensmanagement sowie Lernfabriken.

Norbert Gronau



Norbert Gronau studierte Maschinenbau und Betriebswirtschaftslehre an der Technischen Universität Berlin (TU). 1994 promovierte er im Fachbereich Informatik (TU). Bis einschließlich März 2000 leitete er an der TU Berlin die Lehr- und Forschungsgruppe Produktionsorientierte Wirtschaftsinformatik, im Oktober 2000 habilitierte er zum Thema „Nachhaltige Architekturen industrieller Informationssysteme bei organisatorischem Wandel“. Durch seine Berufung nach Oldenburg leitete Prof. Gronau von 2000 bis 2004 die Abteilung der Wirtschaftsinformatik an der Universität Oldenburg. Im Jahr 2004 folgte er dem Ruf auf eine C4-Professur an der Universität Potsdam und ist seither Lehrstuhlinhaber für Wirtschaftsinformatik, Prozesse und Systeme, einschließlich Electronic Government. Univ.-Prof. Dr.-Ing habil. Gronau ist Research Fellow und Gastdozent an der Stellenbosch University in Südafrika für Knowledge Management. Er initiierte 2008 die Potsdam Consulting Group (PCG), die seit 2016 unter Potsdam Consulting Advisory GmbH firmiert. Prof. Gronau war darüber hinaus an mehreren Internet-Startups zu den Themen Wissensmanagement und Sales Acceleration beteiligt.

Inhaltsverzeichnis

Vorwort	3
Inhaltsverzeichnis	6
Verzeichnis der Abkürzungen und Akronyme	11
Abbildungsverzeichnis	13
Tabellenverzeichnis	15
1 Einleitung	17
2 Grundlagen	19
2.1 Maschinelles Lernen	19
2.1.1 Überwachtes Lernen	20
2.1.2 Unüberwachtes Lernen	21
2.1.3 Teilüberwachtes Lernen	21
2.1.4 Verstärkendes Lernen	21
2.1.5 Passives Lernen	22
2.1.6 Aktives Lernen	22
2.1.7 Tutorielles Lernen	22
2.1.8 Träges Lernen	23
2.1.9 Eifriges Lernen	23
2.1.10 ML-Methoden und -Ansätze	23
2.2 Künstliche Intelligenz	26
2.2.1 Symbolische KI	27
2.2.2 Neuronale KI	27
2.2.3 Simulationsmethodenbasierte KI	28
2.2.4 Phänomenologiemethodenbasierte KI	28
2.2.5 Schwache KI	28
2.2.6 Starke KI	29
2.2.7 Heuristikbasierte KI	29
2.2.8 KI-Methoden und -Ansätze	29

3	Anwendungsbereiche für das Maschinelle Lernen.....	33
3.1	Energie	36
3.1.1	Smart Grid.....	36
3.1.2	Ausbau und Maintenance.....	39
3.1.3	Verwaltung.....	40
3.1.4	Prognose und Analyse	42
3.2	Gesundheit.....	44
3.2.1	Prognosen und Erkennung.....	44
3.2.2	Pharma.....	45
3.2.3	Psychologie.....	46
3.3	Öffentlicher Sektor.....	46
3.3.1	Recht und Ordnung.....	46
3.3.2	Organisation und Management	48
3.3.3	Bildung.....	48
3.4	Mobilität.....	49
3.4.1	Luft- und Raumfahrt.....	49
3.4.2	Fahrzeug	57
3.4.3	Automotive.....	61
3.4.4	Logistik und Transport	66
3.5	Alltag und Entertainment	69
3.5.1	Alltag.....	70
3.5.2	Gaming.....	70
3.5.3	E-Commerce.....	71
3.6	Produktfertigung	71
3.6.1	Betrieb.....	72
3.6.2	Wartung	73
3.7	Finanzen, IT und Software-Engineering.....	73
3.7.1	Prämien.....	73
3.7.2	Entwicklung.....	74

4	Auswahlmethodik von ML-Tools	77
4.1	Unternehmensorientierte ML-Tool-Auswahl	78
4.2	Projektorientierte ML-Systementwicklung	80
4.3	Algorithmische Verfügbarkeit	84
4.4	Ebenen einer ML-Auswahlmethodik	86
4.5	Vorgehensmodell zum Einsatz des ML-Auswahltools.....	87
5	Marktübersicht.....	89
5.1	Vorgehen zur Identifikation von Tools	89
5.2	Toolübersicht	92
5.3	Tools des Maschinellen Lernens im Detail	102
5.4	Heuristische Tools im Detail.....	133
5.5	Genetische Tools im Detail	135
6	Resümee.....	139
	Literaturverzeichnis.....	142

Verzeichnis der Abkürzungen und Akronyme

AI.....	<i>Artificial Intelligence</i>
ANN.....	<i>Artificial Neural Networks</i>
CC.....	<i>Cognitive Computing</i>
DL.....	<i>Deep Learning</i>
ITS.....	<i>Intelligent Transportation Systems</i>
KI.....	<i>Künstliche Intelligenz</i>
KNN.....	<i>Künstliche Neuronale Netzwerke</i>
ML.....	<i>Machine Learning</i>
NKI.....	<i>Neuronale Künstliche Intelligenz</i>
NLP.....	<i>Natural Language Processing</i>
OCR.....	<i>Optical Character Recognition</i>
PMKI.....	<i>Phänomenologiemethodenbasierte Künstliche Intelligenz</i>
SKI.....	<i>Symbolische Künstliche Intelligenz</i>
SMKI.....	<i>Simulationsmethodenbasierte Künstliche Intelligenz</i>
SVM.....	<i>Support Vector Machines</i>

Abbildungsverzeichnis

Abbildung 2-1: Heuristik-Algorithmenbaum (Eigendarstellung).	30
Abbildung 2-2: Genetischer Algorithmenbaum (Eigendarstellung).....	31
Abbildung 3-1: Anwendungsbereiche für ML (Eigendarstellung).	34
Abbildung 3-2: Übersicht ML-Anwendungsbereiche in der Luftfahrt (Eigendarstellung auf Basis von Mousa (Mousa, 2019, Folie 46)). .	49
Abbildung 3-3: Übersicht ML in der Logistik (Eigendarstellung).	67
Abbildung 3-4: Codeanalyse – Zusammenhang statischer und dynamischer Code- analysemethoden (Eigendarstellung).	75
Abbildung 4-1: Beispiel eines Entscheidungsbaums für die Auswahl von ML- Systemen (Eigendarstellung).....	79
Abbildung 4-2: Projektorientierte ML-Systementwicklung (Eigendarstellung).....	81
Abbildung 4-3: Beispiel eines taxonomischen ML-Algorithmus (Eigendarstellung). 85	
Abbildung 4-4: Ebenen-orientierte ML-Darstellung (Eigendarstellung).....	86
Abbildung 5-1: Übersicht ML-Frameworks Teil 1 (Eigendarstellung).	94
Abbildung 5-2: Übersicht ML-Frameworks Teil 2 (Eigendarstellung).	96
Abbildung 5-3: Übersicht ML-Frameworks Teil 3 (Eigendarstellung).	98
Abbildung 5-4: Übersicht ML-Frameworks Teil 4 (Eigendarstellung).	100

Tabellenverzeichnis

Tabelle 1-1: Ergebnisdarstellung - Auftrittshäufigkeit der Begriffe.....	18
Tabelle 3-1: Anwendung von Algorithmen für autonome Fahrzeuge (eigene Darstellung auf Basis von Johanning & Mildner (2015, S. 64, 66)).	64
Tabelle 3-2: Vergleich verschiedener „Platooning“-Projekte (eigene Darstellung auf Basis von Bergenhem, Shladover, Coelingh, Englund und Tsugawa (2012, S. 6) und Dadras, Gerdes und Sharma (2015)).	69
Tabelle 4-1: Unternehmensstrategische Einflüsse auf die KI-Toolauswahl.....	80
Tabelle 5-1: Abweichungen und Ergänzungen gegenüber KDNuggets-Umfrage (Piatetsky, 2019).	91

1 Einleitung

Im Zeitalter der Digitalisierung werden immer mehr Daten generiert, die gegenwärtig teilweise auch für Auswertungen herangezogen werden. Jedoch liegt in diesen Daten viel Potenzial zur Auswertung und Nutzung. Die Daten- und Informationsflut übersteigt bei weitem die für menschliche Fähigkeiten handhabbare Menge an zu analysierenden Daten, um profunde und auf vollständiger Informationsbasis basierende Auswertungen erstellen und Zusammenhänge erkennen zu können. Maschinen, Computer im engeren Sinne, sind in der Lage, die Grenzen dieser menschlichen Verarbeitungskapazität zu überwinden. Mit deren Hilfe können somit sehr große Datenmengen schneller analysiert, daraus gelernt sowie darauf aufbauend Informationen und Wissen generiert werden. Angesichts einer Fülle algorithmischer Ansätze sind Auswahl und Ausgestaltung einer konkreten Künstlichen Intelligenz (KI)-basierten Lösung für ein spezifisches Problem nicht trivial. Um diese Herausforderung zu überwinden, ist es ein Ziel dieses Buches, einen Marktüberblick über vorhandene Ansätze zu bieten.

Allgemein wird mit Maschinellen Lernen (ML) die künstliche Generierung von Wissen durch Lernen auf Basis von Beispielen in Daten, Informationen und Erfahrung verstanden. Dabei werden Algorithmen – das sind mathematische Vorgehensweisen zur Lösung von Problemen – angewendet, um statische Modelle zur Diagnose, Analyse, Klassifikation oder Spracherkennung aufzubauen. Dies geht weit über reines Auswendiglernen hinaus. Das „Verarbeiten“ beinhaltet dabei einerseits die Verwendung von Daten und Informationen entsprechend vordefinierter Regeln und einem vorgegebenen Raum alternativer Lösungspfade, um Wissen im Sinne einer Inferenz abzuleiten. Andererseits beinhaltet es das (kreative) Erschließen von Sachverhalten und Lösungen über vorgegebene Strukturen hinweg; mit vorab unbestimmtem Ergebnis. Letzteres kann als Intelligenz verstanden werden. Diese künstliche Intelligenz (KI) zielt darauf ab, Muster in den Daten zu erkennen und menschliche Entscheidungsstrukturen nachzubilden. Der Bedarf danach ist immens und die Anwendungsfelder breit und heterogen gestaltet. Einen Überblick über und einen Einblick in Anwendungsfelder zu bieten, ist somit ein weiteres Ziel dieses Buches.

Der Markt an ML-Algorithmen, -Ansätzen und Tools ist nicht nur für Fachfremde und Novizen schwer zu überschauen. Sowohl in der Forschung als auch in der Praxis sind unzählige Frameworks und Bibliotheken entwickelt worden. Einige sind leicht zugänglich, übersichtlich aufbereitet und ohne hohen Aufwand anwendbar. Andere hingegen sind sehr komplex ihrer Mächtigkeit, benötigen spezielle Umwelten zum Einsatz und

sind auch für Domänenexperten anspruchsvoll in der Anwendung. Weiterhin existieren Lösungen, die sehr branchenspezifisch sind. Aufgrund der Heterogenität in diesem sich ständig weiterentwickelnden Themenbereich können die unterschiedlichen Ansätze aus verschiedenen Blickwinkeln betrachtet werden. Zur Illustration der aktuellen Relevanz der Thematik werden in Tabelle 1-1: Ergebnisdarstellung - Auftrittshäufigkeit der Begriffe die Häufigkeiten des Auftretens der Suchbegriffe „Machine Learning“ und „Software“ oder „Tools“ oder „Frameworks“ oder „Packages“ für bekannte Literaturdatenbanken dargestellt. Diese Masse an Informationen strukturiert zusammenzuführen und zu kondensieren, sodass für den Leser die Inhalte auf das Relevante reduziert aufbereitet werden, ist ein weiteres Ziel dieses Buches.

Tabelle 1-1: Ergebnisdarstellung - Auftrittshäufigkeit der Begriffe

	Machine Learning und Software	Machine Learning und Frameworks	Machine Learning und Packages
Sciencedirect	88.784	43	40.285
emeraldinsight	> 13.000	> 16.000	> 9.000
springerlink	37.074	26.939	25.270
wiley	50.297	63.453	48.194
Scholar.google	3.370.000	1.620.000	767.000

Dieses Buch soll es dem interessierten Leser einerseits ermöglichen, einen Überblick über gegenwärtig vorhandene Lösungen zu erhalten sowie andererseits konkrete Hilfestellung bei der Auswahl von Algorithmen oder Tools bei spezifischen Problemstellungen bieten. Um diesem Anspruch gerecht zu werden, wurden Lösungen mittels einer systematischen Literatur- und Praxissuche identifiziert sowie anschließend klassifiziert.

Kapitel 2 umfasst theoretische Grundlagen hinsichtlich ML und KI. In Kapitel 3 werden Anwendungsbereiche von ML eingeführt. Darüber hinaus illustriert eine Vielzahl unterschiedlicher Beispiele aus verschiedenen Domänen die Anwendungsmöglichkeiten und Einsatzgebiete von ML und KI. Als Beispiel sei der Einsatz von ML in der (Quell-)Code-Analyse erwähnt. Kapitel 4 beinhaltet ein Vorgehen für die Auswahl von ML-Techniken aller Art, welches eine Grundlage für die spezifische Toolauswahl darstellt. In Kapitel 5 wird der Marktüberblick über vorhandene Lösungen gegeben. Kapitel 6 schließt mit einer Zusammenfassung sowie einem Ausblick.

2 Grundlagen

Es folgt zunächst die Definition grundlegender Begriffe sowie deren Abgrenzung, um ein einheitliches Verständnis für die in dieser Marktstudie verwendeten Begriffe zu erlangen. Weiterhin wird dadurch das Ziel verfolgt, die Möglichkeiten von KI und ML aufzuzeigen. Darüber hinaus fungieren die Verfügbarkeiten von nachfolgend beschriebenen algorithmischen Ansätzen in den betrachteten KI- und ML-Tools als Auswahlkriterien. Sie dienen damit als Grundlage für die Gestaltung einer Auswahlmethode.

2.1 Maschinelles Lernen

Maschinelles Lernen (ML) ist eine Unterkategorie der KI, welche inhaltliche Überschneidungen mit weiteren wissenschaftlichen Feldern, wie Statistik, Informationstheorie, Spieltheorie und Data Science aufweist. Obwohl ML als Teil der KI eingestuft wird, versucht diese - anders als traditionelle KI-Methoden (Turing, 1950) - keineswegs menschliches Verhalten zu imitieren. ML verwendet spezielle computerbasierte Verfahren, wie bspw. die Verarbeitung großer Datenbanken, um menschliche Fähigkeiten zu unterstützen. Sie eint somit sämtliche Ansätze, die Informationen und Wissen auf algorithmischer Basis künstlich generieren. Dies erfolgt, indem ein ML-System aus Beispielen lernt, nach Beendigung der Lernphase Zusammenhänge erfolgreich verallgemeinert sowie versucht, Wissen und Erfahrungen künstlich herzuleiten und anwendbar zu machen.

ML ist ein Feld der Informatik, das es ermöglicht, Computerprogramme so zu konstruieren, dass diese von Daten lernen, deren Inhalte und Struktur verstehen sowie daraus Modelle bilden können. Dabei passen ML-Werkzeuge ihr Verhalten flexibel auf Umgebungsänderungen und Eingabedaten an. Ihre Anpassungsfähigkeit kann damit als eine Haupteigenschaft gewertet werden (Shalev-Shwartz & Ben-David, 2014). Bekannte maschinelle Verfahren sind Neuronale Netze, Deep Learning, Support Vector Maschinen, Entscheidungsbäume, Bayessche Netze, Nächste-Nachbarn-Klassifikationen, Selbstorganisierende Karten, Fallbasiertes Schließen, Instanz-basiertes Lernen, das Hidden-Markov-Modell oder die Regressionsanalyse (Larrañaga et al., 2018). Einzelne Formen des ML finden in nachfolgenden Unterkapiteln eine weiterführende Betrachtung.

2.1.1 Überwachtes Lernen

Ein ML-Algorithmus ist eine Handlungsvorschrift, welche ein mathematisches Modell in die Lage versetzt, Muster aus gegebenen Eingabeinformationen zu erkennen. Eingabedaten werden gewöhnlich in Trainings- und Testdaten aufgeteilt. Im Trainingsdatensatz sind alle Eingabeobjekte mit ihren korrespondierenden Ausgabewerten zugeordnet (sogenannte „labelled data“). Entscheidend ist zudem, dass die Trainingsdatensätze repräsentativ sind und zu lernende Zusammenhänge entsprechend der Zielstellung des Lernens berücksichtigt werden. Daher werden diese in der Regel von Experten gestellt. Ein Data Scientist oder ML-Ingenieur versieht den Datensatz mit Markierungen, sodass wichtige Datensätze hervorgehoben werden können, um besondere Aufmerksamkeit zu erhalten. Anschließend werden die Trainingsdaten einem ML-Algorithmus übergeben, sodass dieser die Verbindungen zwischen Eingaben und Ausgaben erlernen kann. Insofern ein Lernen auf Basis der Abweichung zwischen einem Zielwert und einem durch die KI generierten Wert, eines sogenannten Fehlers, realisiert wird, wird in diesem Zusammenhang auch vom korrigierenden Lernen gesprochen (Rojas, 1996). Sobald der ML-Algorithmus die zugrundeliegenden Muster der Trainingsdaten erlernt hat, wird dieser mit neuen, noch nicht verwendeten Testdaten getestet, sodass die korrekte Funktionsweise des trainierten Systems sichergestellt wird. Der Trainingsprozess dauert so lange an, bis das Modell im Rahmen des Arbeitens mit den Testdaten eine akzeptable Leistung aufweisen kann (Shalev-Shwartz & Ben-David, 2014). Hierbei ist jedoch darauf zu achten, dass eine Überanpassung (sogenanntes „Overfitting“) auf die Testdaten vermieden werden muss, da sonst das Modell nicht korrekt generalisiert.

Zentrale Probleme des überwachten Lernens beziehen sich auf die Klassifikation und Regression, welche beide auf der Vorhersage von Mustern beruhen. Während die *Klassifikation* eine Gruppe überwachter ML-Algorithmen beschreibt, die auf Basis von Eingangsdaten die entsprechenden Kategorien vorherzusagen vermögen, beschreibt die Gruppe der *Regression* eine Gruppe überwachter ML-Algorithmen, welche auf Basis von Eingangsdaten Werte eines kontinuierlichen Wertebereichs vorhersagen können. So werden Klassifikationssysteme hauptsächlich verwendet, wenn sich Kategorien einzeln erkennen lassen, sodass diese mittels „ja“- oder „nein“-Fragen beantwortet werden können. Sofern jedoch ein kontinuierlicher Wert, wie bspw. ein Aktienkurs, vorhergesagt werden soll, kommen zumeist Regressionssysteme zum Einsatz. Hier prognostiziert der ML-Algorithmus die Entwicklung des realen Aktienkurses, da dieser Zusammenhänge auf Basis alter Eingabeposten der Trainingsdaten bereits erlernt hat (Brink, Richards & Fetherolf, 2016).

Werden Regeln einer Klassifizierung als geordneter und gerichteter Baum visualisiert, so wird von einem *Entscheidungsbaum* gesprochen. Dieser dient der Darstellung hierarchisch aufeinanderfolgender Entscheidungen und ermöglicht die formalisierte Darstellung von Erfahrungswissen. ML-Algorithmen helfen hier, Regeln auf Basis von Daten induktiv zu identifizieren.

2.1.2 Unüberwachtes Lernen

Im Unterschied zum überwachten Lernen wird beim unüberwachten Lernen kein Zielwert vorgegeben. Trotz der Abwesenheit von vordefinierten Ergebnissen gilt es, mithilfe eines konkreten ML-Algorithmus Gemeinsamkeiten in den Eingabedaten bzw. Muster zu finden und von einem strukturlosen Rauschen zu unterscheiden (Shalev-Shwartz & Ben-David, 2014). Beispielsweise wird intendiert Eingangsdaten in kompakter Form zu repräsentieren (Komprimierung) oder in ähnliche Segmente zu unterteilen (Segmentierung).

2.1.3 Teilüberwachtes Lernen

Das teilüberwachte Lernen bezieht sich auf eine Mischform aus überwachtem und unüberwachtem Lernen, in der ML-Algorithmen mit Daten operieren können müssen, die nur teilweise mit einem Label versehen sind. Es werden also sowohl gekennzeichnete als auch ungekennzeichnete Datensätze für den Trainingsprozess des Modells verwendet. Wie auch beim überwachten Lernen werden die Markierungen dabei von Experten vorgenommen. Weil neben hohen Kosten und Zeitaufwänden hierbei auch Fehler entstehen können, werden meist nur kleine gekennzeichnete Datensätze zu Trainingszwecken des Algorithmus verwendet (Brink et al., 2016).

2.1.4 Verstärkendes Lernen

Sofern das Lernen die Maximierung eines Nutzens oder eines zu erwartenden Gewinns fördert, indem eine Belohnung oder Bestrafung ausgesprochen wird, wird von verstärkendem Lernen gesprochen. Erstrebenswert ist für das Modell somit die Einnahme eines wünschenswerten Zustands in der Zukunft oder die Wahl einer attraktiven Aktion zur Erreichung dieses Zustands. Da erst durch die Interaktion mit der Umwelt erfahren wird, ob eine Belohnung oder Bestrafung vorliegt, hierfür aber zuvor das Berechnungsergebnis eines Agenten zu einem Input ermittelt werden muss, wird das verstärkende Lernen von einigen Autoren als Spezialform des überwachten Lernens verstanden

(Rojas, 1996). Dies geht einher mit der Vermeidung von nicht wünschenswerten Zuständen in der Zukunft oder der Aussortierung von unattraktiven Aktionen zur Erreichung jenes Zustands.

2.1.5 Passives Lernen

Ein weiteres Unterscheidungsmerkmal von Lernverfahren bezieht sich auf die Art der Herkunft von Trainingsdaten. Hat der Lernende keinen Einfluss auf die Auswahl der ihm präsentierten Trainingsdaten, und somit aus der Fülle von der Beobachtung entstandenen Daten die relevanten Daten eigenständig erkannt werden müssen, so wird von passivem Lernen gesprochen (Fischer, 2013). Da passive Lernverfahren nicht die Möglichkeit haben durch ihre Aktivität (z.B. durch Fragen) Hypothesen experimentell zu überprüfen, kann die Güte von Hypothesen somit nur durch statistische Tests eingeschätzt werden.

2.1.6 Aktives Lernen

Kennzeichnend für ein aktives Lernen ist, dass der ML-Algorithmus auf die Eingangsdaten reagiert, und spezifizierte Fragen stellt, um passende oder korrekte Ergebnisse zu erhalten. Ähnlich eines Experimentierens können Lernvorgänge somit auf ein eigenständiges, aktives Untersuchen durch den Lernenden selbst auf die relevanten Beobachtungen fokussieren. Um die Fragenanzahl möglichst klein zu halten, muss der algorithmische Ansatz die Fragen bestimmen, die einen hohen Informationsgewinn versprechen (Fischer, 2013). Hierbei werden Fragen mittels einer Ergebnisrelevanz bewertet und die attraktivsten Fragen ausgewählt.

2.1.7 Tutorielles Lernen

Wird aus dem Datensatz sämtlicher Beobachtungen eine Vorauswahl getroffen, die aus Sicht eines Dritten, dieser wird hier als Tutor bezeichnet, getroffen, kann ein Lernen effizient gestaltet werden (Fischer, 2013). Da dieser natürlich nur diejenigen Trainingsdaten auswählt, welche einen hohen Informationsgewinn versprechen, muss ein besonderes Augenmerk darauf gelegt werden, dass der Lernende nicht nur die Beschreibung dekodiert und einlernt, die ein Tutor unabsichtlicher Weise in die Trainingsdatensätze kodiert hat, sondern allgemeingültige Zusammenhänge gelernt wurden.

2.1.8 Träges Lernen

Das Träge Lernen (Engl. „Lazy Learning“) bezieht sich auf algorithmische Ansätze, welche das aktuell erlernte Modell nur zu festen Zeitpunkten durch ein Training verbessern. Dies geschieht online zur Zeit einer Anfrage unabhängig davon, ob zwischenzeitlich neue Daten in den Trainingssatz gefügt werden (Aha, 1997b; Auer, 1997). Da gesammelte Daten in angefragten Modellanwendungen zum Zeitpunkt der Anfrage trainiert werden, neigen träge Lernverfahren dazu Zielfunktionen lokal also in der Umgebung eines aktuellen Arbeitspunktes zu approximieren. Einfachen Lernprozessen, steht dementsprechend die durch ein erst zu realisierendes Training verzögerte Verwendung von Modellen gegenüber (Aha; Atkeson, Moore & Schaal, 1997; Birattari, Bontempi & Bersini, 1999). Somit sind träge Lernverfahren insbesondere dann attraktiv, wenn sich ändernde bzw. schnell obsolet werdende Daten dem Trainingsdatensatz hinzugefügt werden und ein kontinuierliches Trainieren nicht möglich ist, da veraltete Daten Trainingsergebnisse verzerren würden.

2.1.9 Eifriges Lernen

Das Eifrige Lernen (Engl. „Eager Learning“) bezieht sich auf algorithmische Ansätze, welche das aktuell erlernte Modell durch Training offline verbessern, sobald neue Daten in den Trainingssatz eingefügt werden (Aha, 1997b; Auer, 1997). Da ein Trainieren unabhängig von Anfragen realisiert wird, also sozusagen offline zwischen den Anfragen, fallen Lernprozesse komplex aus und Modelle sind deutlich aufwändiger zu trainieren: sie benötigen aufgrund des ständigen Anpassens durch neue Daten deutlich häufiger Trainingsvorgänge als „weniger eifrige“ oder „träge“ Lernansätze. Den aufwändigen Ansätzen und stets aktuellen und schnell einsetzbaren Modellen steht insbesondere die schnelle Ergebnisbereitstellung durch vortrainierte Modelle gegenüber, sodass das Eager Learning gegenüber dem Lazy Learning also deutlich Zeitvorteile einer Abfragezeit liefert. Ein weiterer Vorteil eifriger Lernansätze ist, dass erstellte Modelle dazu neigen das globale Optimum zu fokussieren.

2.1.10 ML-Methoden und -Ansätze

Aus einer Fülle von ML-basierten Methoden und Ansätzen der wichtigsten Vertreter (Larrañaga et al., 2018) sind im Folgenden Bereiche definiert, in welche Funktionen und Kriterien von ML-Tools eingeordnet werden können.

Entscheidungsbäume verwenden einen baumartigen Graphen oder ein Modell von Entscheidungen und entsprechenden Konsequenzen. Sie führen eine Reihe von rekursiven Aktionen durch, indem sie die Daten in zwei Gruppen aufteilen. Dabei werden

komplexe Beziehungen zwischen den Eingabevariablen vereinfacht, indem diese in Untergruppen unterteilt und schlussendlich in ein Baummodell überführt werden. Die entdeckten Erkenntnisse können vom Benutzer leicht verstanden und interpretiert werden, da sie in Form eines hierarchischen Baums dargestellt werden. Entscheidungsbäume finden insbesondere im Geschäfts- und Kundenbeziehungsmanagement, sowie im Gesundheitswesen oder im Energieverbrauchswesen ihre Anwendung (Brink et al., 2016).

Support Vector Machines (SVM) finden eine Linie oder Hyperebene, die zwei Gruppen bereits klassifizierter Daten entsprechend ihrer Merkmale so weit wie möglich voneinander trennt. Kommende neue Daten können somit unterschieden werden, indem analysiert wird, auf welcher Seite der Linie sich diese befinden. SVMs werden für Textklassifizierungen, Spamerkennung, Stimmungsanalyse und Bilderkennung (farbbaasierte Klassifizierung) verwendet (Brink et al., 2016).

Ensemble Learning bezieht sich auf die Verfahren, mit denen mehrere Lernmaschinen trainiert und deren Ergebnisse kombiniert werden. Zahlreiche empirische und theoretische Studien haben gezeigt, dass Ensemblemodelle sehr oft eine höhere Genauigkeit erreichen als Einzelmodelle (Brink et al., 2016).

Instanzbasierte Algorithmen erstellen typischerweise eine Datenbank mit Beispieldaten (Instanzen) während des Trainings und vergleichen neue Daten (neue Instanzen) mit dieser Datenbank, um somit Übereinstimmungen zu finden und Vorhersagen zu treffen. Das Vorgehen wird als Instanz-basiert bezeichnet, weil mit dessen Hilfe Vorhersagen entsprechend den Trainingsinstanzen gemacht werden können. Die Trainingsdaten werden in einer Datenbank gespeichert. Kommt eine neue Instanz auf, so wird die Ähnlichkeit zwischen der neuen Instanz und den bestehenden Trainingsinstanzen festgestellt sowie darauf aufbauend eine Entscheidung getroffen (Brink et al., 2016).

Clustering ist eine Methode, deren Ziel es ist, eine Gruppe von Objekten in Klassen ähnlicher Objekte zu gruppieren. Ein Cluster ist eine Sammlung von Objekten, die sich innerhalb eines Clusters ähneln und sich von den Objekten in anderen Clustern unterscheiden. Das Hauptziel besteht darin, die Daten in Gruppen mit gemeinsamen Attributen zu maximieren und hierarchisch oder zentrumsbasiert zu organisieren. Clustering wird in verschiedenen Bereichen wie etwa Bioinformatik, Medizin, Marktforschung, Analyse sozialer Netzwerke, Kriminalitätsanalyse, Text Mining, Empfehlungssysteme und Bildsegmentierung (Objekterkennung) angewendet (Everitt, 2011).

Algorithmen zur Dimensionsreduktion fokussieren auf die Dimensionalität in einem Datensatz. Diese beschreibt die Anzahl der Merkmale in einem Datensatz. Ist die Anzahl der Features zu hoch, können Probleme beim Training des Modells auftreten. Unter Dimensionsreduktion wird somit der Prozess verstanden, die Einflussfaktoren zu reduzieren und auf die wesentlichen Faktoren zu beschränken. Um dies zu bewerkstelligen, kann sich ein Algorithmus zwei Möglichkeiten bedienen: Feature-Selektion und Feature-Extraktion. Die Feature-Selektion filtert irrelevante oder redundante Merkmale aus dem Datensatz. Durch die Feature-Extraktion wird ein neuer kleinerer Satz von Features erstellt, mit denen die wichtigsten Informationen erfasst werden können (Roweis & Saul, 2000).

Frequent Pattern Mining zielt darauf ab, Beziehungen zwischen den Einträgen in einer Datenbank zu finden. Kommt eine Struktur wiederholt in einem oder mehreren anderen Datensätzen vor, so ist dies ein Frequent-Pattern. Die Datenbank kann verschiedene Arten von Daten enthalten, z. B. Zeichenfolgen, räumliche Daten oder Grafiken. Pattern Mining ist auf Algorithmen spezialisiert, die interessante, unerwartete oder nützliche Muster in den Daten entdecken. Einige Forscher definieren interessante Muster als häufig vorkommend, andere versuchen seltene Muster zu entdecken. Frequent Pattern Mining wurde erstmals von Agrawal et al. in Form von Association Rule Mining für die Analyse von Marktkörben vorgeschlagen (Agrawal, Imieliński & Swami, 1993). Basierend auf einer Reihe von Transaktionen sollen mittels Association Rule Mining Regeln definiert werden. Diese unterstützen die Forscher, die Häufigkeit eines bestimmten Artikels, basierend auf der Häufigkeit anderer Artikel in der Transaktion, zu bestimmen. Ein typisches Beispiel ist ein Supermarkt, in dem festgestellt wird, dass Kunden, die Milch und Butter kaufen, auch Getreide kaufen. Andere Anwendungsbereiche für das Frequent Pattern Mining sind die Analyse von Weblinks, die Genomanalyse und die Klick-Stream-Analyse (die Route, die Besucher bei der Navigation auf einer Website wählen). Der beliebteste Algorithmus aus dieser Gruppe ist die Apriori-Assoziationsanalyse, die erstmals 1993 veröffentlicht wurde. Seitdem wurden viele Varianten von Apriori-basierten Algorithmen entwickelt, z. B. FP-Growth und Eclat (Aggarwal & Han, 2014).

Sequential Pattern Mining behandelt Daten, die als eine Reihe von Sequenzen dargestellt werden. Eine Sequenz beschreibt eine Reihe von Transaktionen. Sequential Pattern Mining wird in vielen Fällen angewendet, z. B. zur Auffindung von Kaufmustern bei Kunden (z. B. Kauf eines Laptops, einer Digitalkamera, eines Kartenlesegeräts innerhalb von mehreren Monaten) oder bei der Analyse von DNA-Sequenzen und Genen (Li, Xu, Ruan & Pan, 2005).

Deep Learning (DL) Methoden entstehen von künstlichen neuronalen Netzen. Diese Methode wird häufig bei teilüberwachten Lernproblemen angewendet. Der Unterschied zu Neuronalen Netzwerken ist die Anzahl der versteckten Schichten, auf denen der Lernprozess stattfindet. Tiefe neuronale Netze sind Neuronale Netzwerke mit mehr als zwei versteckten Schichten (Kennedy, 2010).

Natural Language Processing (NLP) ist ein Forschungsgebiet, das sich mit der Analyse, dem Verständnis und der Produktion menschlicher Sprache und Texte befasst. NLP wird in verschiedenen Bereichen wie maschineller Übersetzung, Spracherkennung, Erzeugung natürlicher Sprache, Expertensystemen, Textverarbeitung, Textverständnis sowie zur Dialogführung mit Benutzern als auch zur Abfrage von Informationen angewendet (Chowdhury, 2003).

2.2 Künstliche Intelligenz

Während ML hauptsächlich darauf fokussiert, Maschinen durch Lernansätze eine Intelligenz zu verleihen, umfassen Methoden der Künstlichen Intelligenz (Englisch „Artificial Intelligence“) als Obermenge des ML ebenso Ansätze, welche die Automatisierung intelligenten Verhaltens fokussieren oder zumeist versuchen, menschliches Verhalten zu imitieren. Hierzu zählen die symbolische KI, welche sich einer Intelligenzleistung von einer begrifflichen Ebene nähert oder die neuronale KI, welche sich einer Intelligenzleistung von einer nicht-begrifflichen, datenorientierten Ebene nähert. Weiter existieren Algorithmen einer Simulationsmethode, welche sich möglichst realitätsgetreu an der kognitiven Leistung des Menschen orientieren, sowie Algorithmen einer phänomenologischen Methode, die sich an einer KI möglichst nah an der ergebniswirksamen Darstellung orientieren.

Die Betrachtungen der KI sind insbesondere vom Bereich des Cognitive Computing zu unterscheiden, da sie nur schwer abzugrenzen sind. Unter *Cognitive Computing* (CC) wird zum einen die Simulation menschlicher Gedanken verstanden, sodass selbstlernende Systeme die Datenverarbeitung, -wahrnehmung und Mustererkennung in einer Weise organisieren, wie es auch das menschliche Gehirn durchführt (Marr, 2016). Zum anderen wird CC auch als Plattform verstanden, welche diverse Ansätze der KI und des ML eint sowie weiterhin die hardwareseitige Realisierung betrachtet. Die Einordnung als Teilmenge von KI wie auch zu ML ist somit zurzeit nicht allgemein anerkannt, denn kognitive Systeme beziehen verschiedene ML-Techniken wie auch verschiedene Teilbereiche der KI ein, bringen diese zusammen und lassen sie ineinandergreifen.

Weiter sind regionale Unterschiede in der Verwendung der Begrifflichkeiten zu erkennen. Einige Berichte sprechen auch von einer maschinellen Intelligenz, meinen jedoch KI. Andere identifizieren dessen Teilgebiete in ML, maschinelles Sehen (Computer Vision), Natural Language Processing (NLP) und Robotik (Marr, 2016), sodass im Folgenden ML als ein wichtiger Bestandteil von CC verstanden wird. Nachfolgende Unterkapitel fokussieren hierbei diverse Interpretationsansätze von KI sowie Methoden und Ansätze der KI.

2.2.1 Symbolische KI

Basierend auf einem Top-down-Ansatz ist eine *Symbolische KI* (SKI) bestrebt, Intelligenz durch verständliche Begriffe nachzubilden. Durch die sogenannte Symbolik bleiben Wissensdeterminanten eindeutig benennbar und die Arbeitsweise eines KI-basierenden Systems bleibt transparent.

2.2.2 Neuronale KI

Neuronale Netze gehören zum Ansatz des überwachten Lernens. Durch gelabelte Trainingsdaten und eine Rückverzweigung kann das neuronale Netz trainiert werden. Neuronale Netze sind die Basis vieler Deep-Learning-Methoden (Kennedy, 2010). Hierunter wird ein tiefes neuronales Netz verstanden, was viele Knotenpunkte in unterschiedlichen Schichten enthält (Marr, 2016).

Basierend auf einem Bottom-up-Ansatz ist die *Neuronale KI* (NKI) bestrebt, das menschliche Gehirn so präzise wie möglich nachzubilden. Auch wenn im Verbund diverse Einheiten kooperieren, um einen relevanten Output zu erzeugen, so können Wissensdeterminanten nicht eindeutig benannt werden. Weiterhin bleibt die Arbeitsweise eines KI-basierten Systems intransparent. Diverse Methoden und Ansätze einer NKI sind im Bereich von künstlichen neuronalen Netzwerken zu finden.

Künstliche Neuronale Netze (KNN), im Englischen als *Artificial Neural Networks* bezeichnet, basieren auf der Funktionsweise des menschlichen Gehirns. Ziel ist es, für eine gewisse Problemstellung ein Netz aus Knotenpunkten und verknüpfenden Kanten aufzubauen. Diese Knotenpunkte sind die Neuronen des Netzes. Sie sind über mehrere Schichten angeordnet, die unterschiedliche Zwecke verfolgen. Das Lernen wird durch Veränderungen der Querverbindungen ermöglicht. Ein einfaches neuronales Netz besteht aus einer Eingabeschicht, einer versteckten Schicht sowie einer Ausgabeschicht. Die Eingabedaten werden in mathematische Werte überführt. Dieser voranstehende Schritt wird als Feature-Konstruktion bezeichnet. Die Features stellen die Eingabewerte für das zu erstellende ML-Modell dar. Die Werte der Eingabeschicht werden in der

versteckten Schicht mittels mathematischer Operationen verarbeitet. Die entstehenden neuen numerischen Werte werden in der Ausgabeschicht gesammelt und schlussendlich in ein für den Benutzer verständliches Ergebnis interpretiert. Der Einsatz von neuronalen Netzen reicht von Bild- und Spracherkennung, der Filterung von sozialen Netzwerken bis hin zu medizinischer Diagnostik.

Neuronale Netze gehören größtenteils zum Ansatz des überwachten Lernens. Durch gelabelte Trainingsdaten und eine Rückverzweigung kann das neuronale Netz trainiert werden. Neuronale Netze sind hierbei die Basis vieler Deep-Learning-Methoden (Kennedy, 2010). Hierunter wird ein tiefes neuronales Netz verstanden, was viele Knotenpunkte in unterschiedlichen Schichten enthält (Marr, 2016).

2.2.3 Simulationsmethodenbasierte KI

Basierend auf Simulationsmethoden ist eine *Simulationsmethodenbasierte KI* (SMKI) bestrebt, sich so nah wie möglich an den kognitiven Prozessen des Menschen zu orientieren. Eine Problemlösung soll sich somit an menschlichen Problemlösungsstrategien orientieren und intelligentes Verhalten imitieren (Ramacher & von der Malsburg, 2009).

2.2.4 Phänomenologiemethodenbasierte KI

Eine *Phänomenologiemethodenbasierte KI* (PMKI) ist bestrebt, sich so nah wie möglich an den tatsächlichen Ergebnissen zu orientieren. Der Ursprung der Erkenntnisgewinnung wird somit in unmittelbar gegebenen Erscheinungen vermutet, welche als Phänomene bezeichnet werden. Insofern es möglich ist Phänomene formal zu beschreiben, können algorithmische Ansätze intelligentes Verhalten generieren (Bower & Beeman, 2012).

2.2.5 Schwache KI

Zurzeit sind KI-Anwendungen im Bereich der schwachen KI einzuordnen. Diese definiert sich dadurch, dass sie auf mathematische Modellierungen zurückzuführen sind, welche für einen spezifischen Problembereich optimiert wurden und konkrete Anwendungsprobleme meistern können. Sie unterstützen somit das menschliche Denken und technische Anwendungen (Nilsson, 2009). Aufgrund der hohen Fokussierung auf spezifische Anwendungsprobleme einer schwachen KI geht es hier nicht um die Schaffung eines Bewusstseins, sondern um die Simulation eines intelligenten Verhaltens mittels mathematischer und Informatikbasierter Ansätze (Buxmann & Schmidt, 2019; Otte, 2019; Winter, 2018).

2.2.6 Starke KI

In dem gegenwärtig noch theoretischen Bereich einer starken KI sind Ansätze zu finden, welche mit dem Menschen eigenständig und auf Augenhöhe arbeiten können. Hier wird ein intelligentes Verhalten nicht nur simuliert, wie es im Bereich der schwachen KI-basierten Anwendungen typisch ist, sondern es wird aufgrund der Schaffung eines eigenständigen Bewusstseins des Systems eine echte Intelligenz geschaffen (Buxmann & Schmidt, 2019; Otte, 2019; Winter, 2018).

2.2.7 Heuristikbasierte KI

Ein heuristischer Algorithmus basiert auf einer Methode zur Lösung komplexer Optimierungsprobleme, welcher mit unvollständigen Informationen und begrenztem Wissen sowie wenig Zeit zu ebenso wahrscheinlichen Aussagen oder praktikablen Lösungen gelangt, wie viele nicht-heuristische Methoden (Gigerenzer & Todd, 1999). Somit arbeiten heuristische Algorithmen häufig mit Schätzungen, intuitiv-intelligenten Vermutungen oder zusätzlichen Annahmen, um so schnell wie möglich zu einer akzeptablen Lösung zu gelangen. Es kann jedoch nicht garantiert werden, dass eine optimale Lösung gefunden wird. Eine Heuristik eignet sich somit besonders, wenn eine genaue Berechnung der optimalen Lösung nicht möglich oder zu zeitaufwendig ist.

2.2.8 KI-Methoden und -Ansätze

Aus einer Fülle von KI-basierten Methoden und Ansätzen finden sich aus Sicht der Autoren im Folgenden die wichtigsten Vertreter an KI-Methoden und -Ansätzen. Sie dienen im Vergleich und der Auswahl geeigneter KI-Tools als Grundlage zur Analyse inwiefern derartige KI-Methoden und -Ansätze vom jeweiligen Tool zur Verfügung gestellt werden. Neben den Bereich der mathematischen Optimierungen sowie iterativen Vorgehen stellt insbesondere der Bereich der Heuristik einen Fokus dar, da heuristische Algorithmen alleine oder in Verbindung mit weiteren Algorithmen angewendet werden können, um die Effizienz der Lösungsfindung zu verbessern. Abbildung 2-1 visualisiert hierfür mögliche und weit verbreitete, heuristische Algorithmen. Die darin gewählten Farben dienen dabei zur unterstützenden Zuordnung zu den jeweiligen Überpunkten. Wesentliche Hauptkategorien werden im Folgenden beschrieben. Aufgrund der besonderen Rolle genetischer Ansätze erfolgt nachfolgend eine detailliertere Betrachtung dieser.

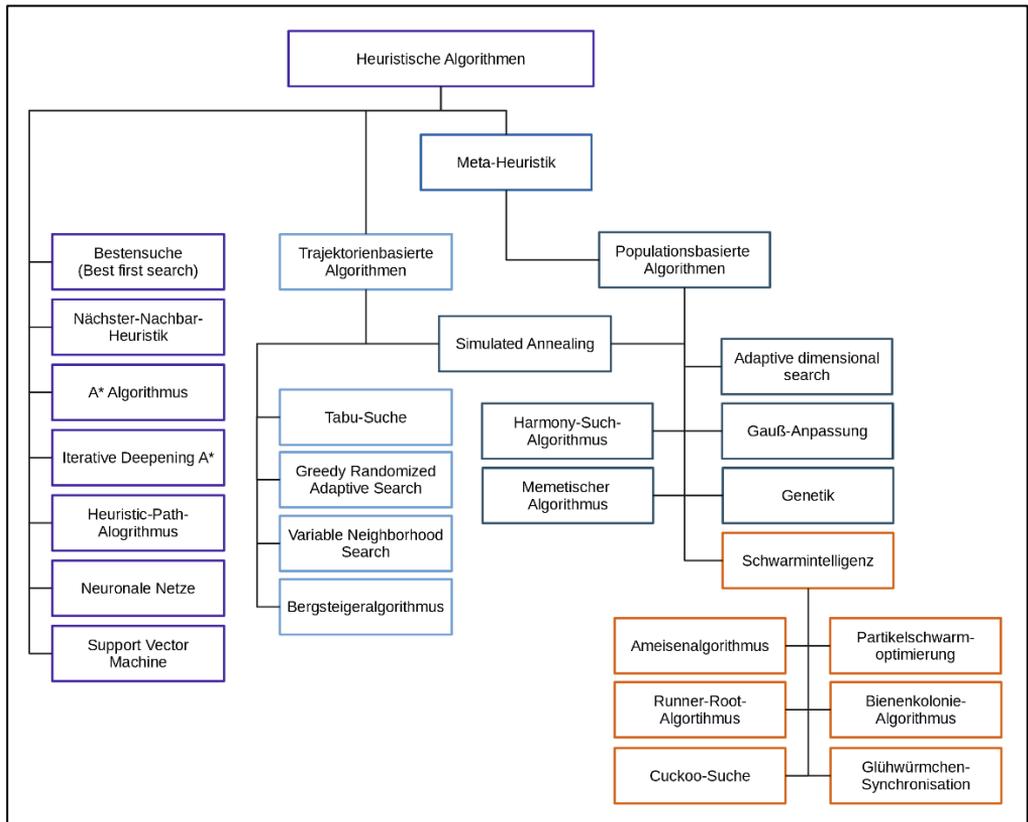


Abbildung 2-1: Heuristik-Algorithmenbaum (Eigendarstellung).

Eine **Heuristik** bezieht sich auf Verfahren, mittels welcher aufgrund von beschränktem Wissen und unvollständigen Informationen unter geringem Zeiteinsatz dennoch zu wahrscheinlichen Aussagen oder praktikablen Lösungen gelangt werden kann. Sofern von einem heuristischen Verfahren auf einer höheren Ebene gesprochen wird, ist eine Ebene gemeint, die unabhängig von der problemspezifischen Implementierung einer Heuristik das Finden einer näherungsweise Lösung behandelt. Hier wird auch von einer **Meta-Heuristik** gesprochen. Beispielsweise werden dabei mehrere Suchstrategien in einer abstrakten Abfolge von Heuristiken kombiniert. Die hier zu Grunde gelegte Unterteilung von Meta-Heuristiken folgt der Unterscheidung zwischen Trajektorien- und Populationsbasierten Algorithmen.

Trajektorienbasierte Algorithmen verwenden einen Agenten, der gleichzeitig einen Pfad durchläuft und somit einer einzelnen Lösung entspricht. Im Vergleich zu diesen bedienen sich **Populationsbasierte Algorithmen** diverser Agenten, die interagieren und

diverse Pfade durchlaufen. Es werden also multiple Lösungen produziert. Diese können weiter in genetische Intelligenz und Schwarmintelligenz unterteilt werden, die allesamt von der Natur inspiriert sind.

Algorithmen einer **Schwarmintelligenz** ahmen das selbstorganisierte Verhalten von zu meist tierischen Schwärmen nach. Einzelne Tiere werden hierbei als einfache Agenten betrachtet. Sie folgen einfachen Regeln und interagieren miteinander, um eine Lösung für ein Problem zu finden (Bierwirth, 2014; Kramer, 2006; Poli, Kennedy & Blackwell, 2007; Salhi, 2017).

Die **genetischen** Algorithmen orientieren sich an der biologischen Evolution von Lebewesen. Ansätze verfügen über eine Grundgesamtheit möglicher Lösungen, der sogenannten Population, von denen die am wenigsten mit einer Ziellösungen übereinstimmenden Lösungen nach und nach beseitigt werden (Goldberg, 1989). Dieser Vorgang wird solange wiederholt, bis nur noch eine Generation, nämlich die attraktivste Lösung, übrigbleibt. Abbildung 2-2 visualisiert die generische Funktionalität genetischer Ansätze, welche nachfolgend näher spezifiziert werden.

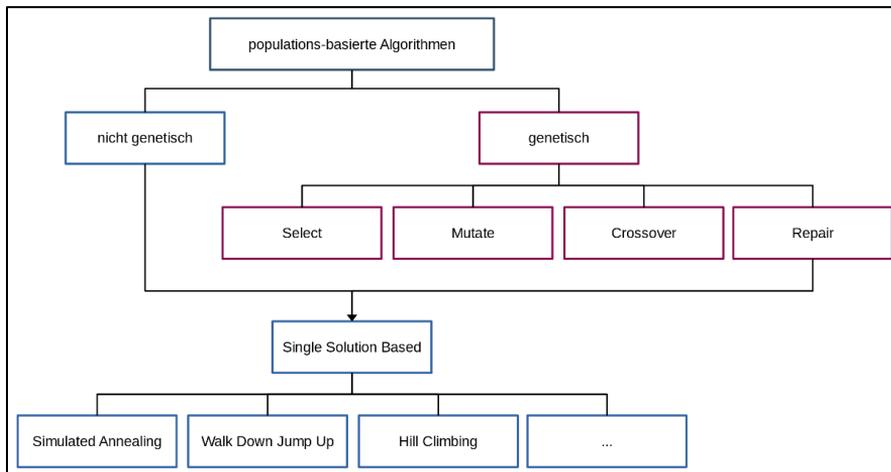


Abbildung 2-2: Genetischer Algorithmenbaum (Eigendarstellung).

Eine genetische Population beschreibt eine "Generation", die normalerweise keine Referenzen zu Vorgängergenerationen aufweisen. Da wir auf einer Grundgesamtheit basierende Algorithmen als iterativ betrachten, ist der Begriff "Generation" auch als Ergebnis einer Iteration zu verstehen. Konkreter formt eine Generation eine Reihe von Lösungen. Jede Iteration eines genetischen Ansatzes kann aus den folgenden vier Schritten oder deren Kombination bestehen (Kramer, 2006):

- **Das Auswählen („Select“):** Nachdem die vorherige Iteration erstellt wurde, können die besten Lösungen für die aktuelle Iteration ausgewählt werden. Weiter ist es sinnvoll, auch einige schlechte Lösungen zu behalten ("Bas-tards"), um eine Vielfalt in nachfolgenden Generationen zu bewahren. Beispielsweise können die Lösungen ausgewählt werden, die es zu verwandeln gilt.
- **Das Mutieren („Mutate“):** Die Lösungen der aktuellen Generation werden geändert, bzw. mutiert. Ggf. wird dieser Schritt ausgeführt, nachdem der Schritt "Select" bereits angewandt wurde. Mutieren bedeutet hier die Attribute der Lösung in einer Weise zu ändern, sodass nachfolgende Generationen auf diesen Mutationen aufbauen können.
- **Das Kreuzen („Crossover“):** Es werden Lösungen kombiniert, die möglicherweise in einem vorherigen Schritt bereits ausgewählt wurden. Die Attribute der Lösungen, die gekreuzt werden, werden miteinander vermischt. Beispielsweise können so einige Attribute einer Lösung weggelassen und durch Attribute einer anderen Lösung ersetzt werden.
- **Das Reparieren („Repair“):** Es liegen Generationen vor, die bereits verändert, vermischt oder verändert und vermischt wurden. Mit Hilfe von nicht-genetischen "Einzellösungsalgorithmen" werden diese Schritt für Schritt verbessert, sodass in der Metapher der Genetik eine mutierte, gekreuzte oder ausgewählte Generation repariert wird. Dies bedeutet, dass eine bestimmte Generation hinsichtlich eines Optimierungsziels gezielt weiterentwickelt wird.

Die Ausführung der beschriebenen Schritte folgt keiner festgeschriebenen Reihenfolge und man ist nicht gezwungen, sämtliche Schritte auszuführen. Es ist jedoch sinnvoll den Repair-Schritt als letztes durchzuführen, da beste Generationen gezielt zu einem Optimum geführt werden.

Nicht-genetische Einzellösungsalgorithmen enthalten typischerweise heuristische Ansätze. Der Name der Kategorie leitet sich von dem Anwendungsziel der Klasse zugehöriger Algorithmen ab: Sie betrachten eine gewählte Generation und produzieren durch Ihre Anwendung eine verbesserte Lösung. Weitere Ausführungen, insbesondere zu Crossover-Algorithmen, sind in Kramer (2006) zu finden.

3 Anwendungsbereiche für das Maschinelle Lernen

Angesichts einer immer größer werdenden Datenflut und dem Potenzial von ML-Verfahren diese Datenflut handhaben zu können, öffnen sich zahlreiche Anwendungsbereiche für ML. Im Folgenden werden Einblicke in die Anwendung von ML in verschiedenen Kontexten, Anwendungsgebieten und Domänen gegeben, die das weitreichende Potenzial verdeutlichen sollen. So ist in Abbildung 3-1 eine Übersicht dieser zu finden, sowie eine Kategorisierung einzelner Unterbereiche. In den nachfolgenden Unterkapiteln werden aus Sicht der Autoren die wichtigsten Bereiche vorgestellt.

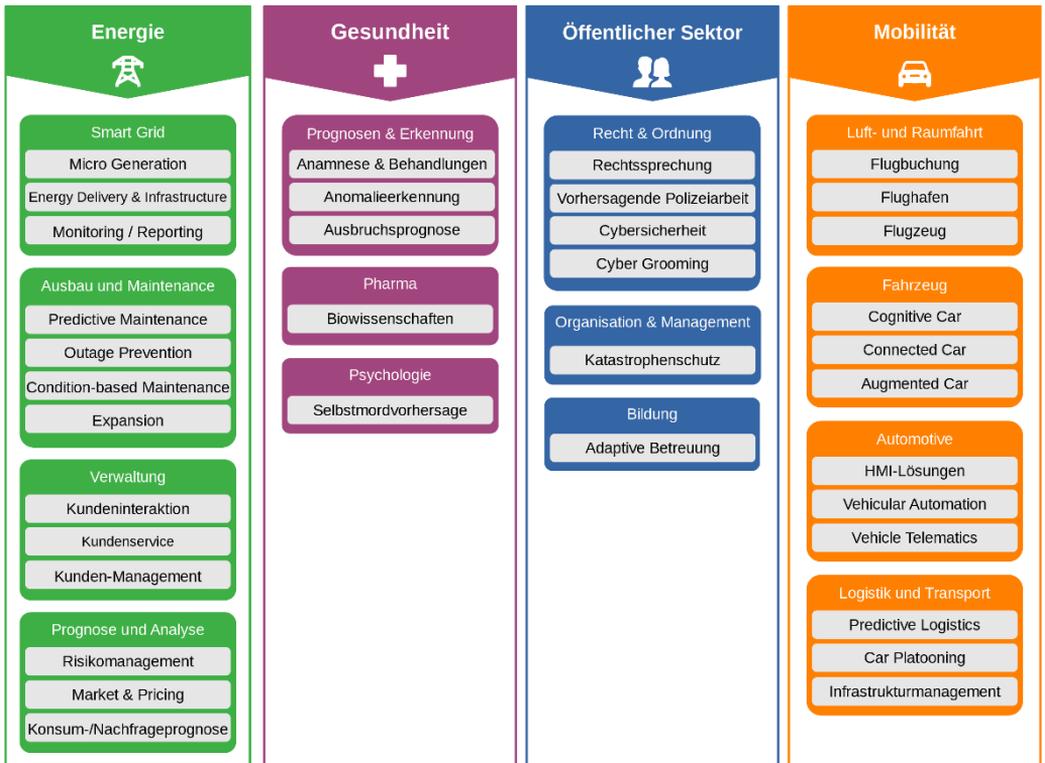


Abbildung 3-1: Anwendungsbereiche für ML (Eigendarstellung).

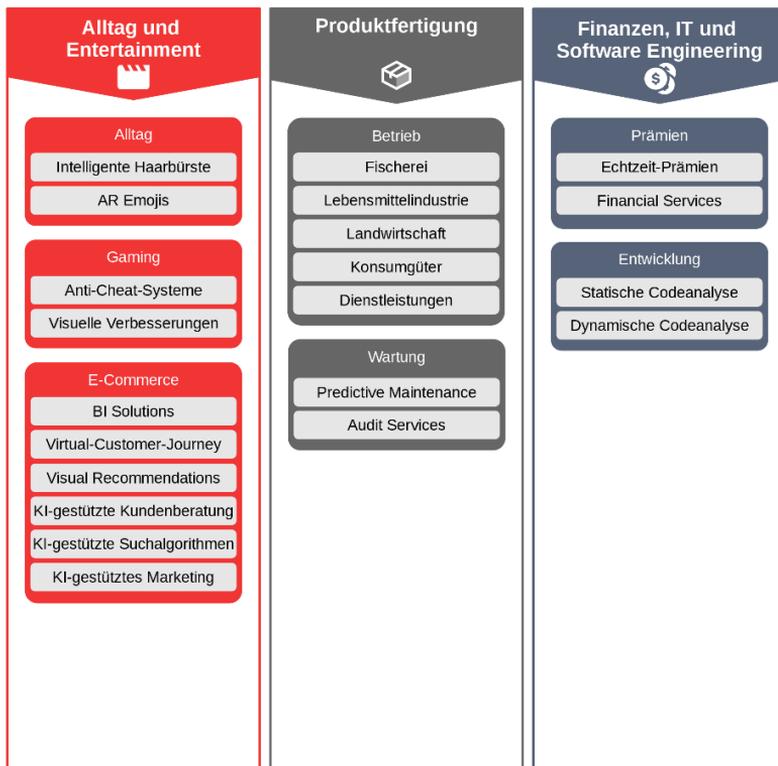


Abbildung 3-1: Anwendungsbereiche für ML, Fortführung (Eigendarstellung).

3.1 Energie

Im Energiesektor existieren unterschiedliche Anwendungsszenarien, von denen im Folgenden einige beschrieben werden. Insgesamt ist der Bereich der Energie vom Schaffen autonomer Energie-Cluster geprägt, welche eigenständig operieren können und im intelligenten Netzwerk, einem sogenannten „Smart Grid“ flexibel und nachhaltig gesteuert werden können.

3.1.1 Smart Grid

Micro Generation

Im Anwendungskontext der Micro Generation wird das intelligente Identifizieren und Handhaben von Micro- und Macro-Strukturen im Smart Grid fokussiert, die auf einer beliebig detaillierten und ausgeprägten Stufe spezifiziert werden können. Aufgrund der hierarchischen Struktur lassen sich diverse Managementebenen identifizieren, welche die große Anzahl an autonomen Einheiten für ein Management überhaupt handhabbar macht. Die wichtigsten Anwendungsbeispiele in diesem Kontext sind:

- **Managing Virtual Power Plants:** Im Kontrast zur Sicherstellung der ordnungsgemäßen Koppelung von Microgrids an das Macrogrid adressiert dieser Bereich das Aggregieren von Microgrids zu virtuellen Kraftwerken (Pudjianto, Ramsay & Strbac, 2007). Dies ermöglicht es Netzbetreibern, Gruppen von Microgrids wie vollwertige Kraftwerke zu betrachten und in die Gesamtverwaltung miteinzubeziehen. In diesem Zusammenhang ist zu beachten, dass einzelne Microgrids aufgrund ihrer Größe für Netzbetreiber entweder nicht sichtbar sind, oder aber die Integration tausender einzelner Microgrids unzumutbar ist (Pudjianto et al., 2007).
- **Predicting Microgrid Load:** Aufgrund ihrer Größe und der unregelmäßigen Produktionsleistung ist die Planung und Steuerung des Einflusses von Microgrids auf die Energienetzlast schwieriger als bei konventionellen Kraftwerken. Jedes Microgrid muss deshalb seine zukünftig erwartete Netzbelastung vorhersagen und an den Netzbetreiber kommunizieren, welcher diese Informationen aggregiert und in die Steuerung des Macrogrid einfließen lässt (Amjady, Keynia & Zareipour, 2010).

Energy Delivery & Infrastructure

Energy Delivery & Infrastructure bildet das Bindeglied zwischen den Energieerzeugern und dem Energieverbraucher. In ihr konzentrieren sich Aufgaben, welche den Betrieb

und die Wartung des Netzes betreffen, um schlussendlich Energie zum Kunden befördern zu können. Des Weiteren ist auch der Ausbau des Netzes ein fester Bestandteil dieser Domain. Hierbei soll das Netz der Zukunft gleichzeitig Ressourceneffizienz und Stabilität verbessern und neue Anforderungen durch den erhöhten Koordinationsaufwand von vermehrten Energieerzeugern bewältigen (Bundesnetzagentur).

Application Architecture ist eine weitere Domain. Der Grid Manager (bspw. SIMATIC WinCC) aggregiert aus dem Netz empfangene Daten im Sinne eines Data Aggregator/Concentrator, überwacht und steuert, sodass eine gleichmäßige Auslastung von Spannung und Frequenz erreicht wird. Bei einem Ausfall von Teilen im Netz findet der Outage Manager (bspw. Milsoft DisSPatch (Milsoft, 2017)) eine Lösung, um den Fehler zu überbrücken. Durch den Meter Data Manager (bspw. Oracle Utilities Meter Data Management (Oracle, 2017)) werden die Daten gespeichert und die Last in einem bestimmten Intervall für eine Region gemessen, wodurch entschieden werden kann, wie sich die Last insgesamt verteilen lässt. Außerdem wird durch den Demand Forecaster (bspw. Oracle Utilities Meter Data Analytics (Oracle, 2017)) die zukünftige Nachfrage vorhergesagt, wodurch entschieden werden kann, ob ad hoc Energie von weiteren Erzeugern eingespeist werden muss.

Der Event Manager (bspw. SAP Maintenance Planner (SAP, 2017)) verwaltet und optimiert geplante Termine, so dass die Häufigkeit und Dauer von Ausfällen minimiert werden kann, wohingegen ein Parts Manager Informationen über Verfügbarkeit und Haltbarkeit von Ersatzteilen verwaltet (bspw. Siemens SSPS (Mayer, 2016)).

Grid Expansion Planer stellt eine Software zur Netzplanung dar (bspw. Siemens SIN-CAL (Siemens, 2017)). Auf Grundlage des Szenariorahmens der Bundesnetzagentur und des existierenden Netzes wird eine Netzerweiterung erstellt und simuliert. Der Szenariorahmen umfasst errechnete Standardlastprofile mit Energiemengen, Budget-Informationen sowie Prognosen über zukünftige Erzeugungsanlagen und ihre Verteilung (Bundesnetzagentur). Zukünftig ist mit einer automatisierten Netzplanung zu rechnen (Stromnetze, 2017).

Innerhalb der Technologieinfrastruktur liegen ein Data Manager, welcher über Automated Meter Management Funktionen verfügt, und der Demand Forecaster – sie beide bilden mit der Advanced Metering Infrastructure (oft auch als AMI bezeichnet) ein Meter Data Management System (MDMS). Dieses Kommunikations-Netzwerk besteht hauptsächlich aus Smart Metern und Smart Gateways, die die Informationen zu den oben genannten Applikationen des Betreibers übertragen. Eine Schlüsselanforderung für das Meter Data Management System ist dabei die Sicherstellung der Interoperabilität zwischen den technischen (syntaktischen), informationsorientierten (semantischen)

und organisationalen (pragmatischen) Ebenen (vgl. Smart Grid Coordination Group, 2012). In der Praxis findet man zudem dedizierte Outage Management Systeme (OMS), welche aber Überschneidungen in der Funktionalität zu moderneren SCADA Systemen haben. Gleiches gilt für Demand Management Systeme (DMS), welche durch MDM ersetzt werden. Für Letztere wächst der Markt laut GTM Research um 500 Prozent.

Monitoring / Reporting

Die **Operation** dient zur gleichmäßigen Auslastung des Verbundnetzes und der Energieerzeuger und ermöglicht einen stabilen und effizienteren Betrieb, wodurch auch der Verschleiß minimiert wird. Ein Smart Grid, welches diese Anforderungen erfüllt, zeichnet sich durch folgende, zum Teil automatisierte Fähigkeiten, aus (Sultanow, von Tucher, Biedulski, Lesny & Hanser, 2017):

- **Aggregate Data:** Da durch Sensoren und Zeigermessgeräte (Phasor Measurement Units – kurz PMU) gesammelte Daten innerhalb des Netzes von einer Vielzahl von Systemen zur weiteren Analyse benötigt werden, müssen diese an einer zentralen Stelle gesammelt werden.
- **Monitor Grid:** Die elektronischen Bauteile im Netz können trotz Optimierungen nicht komplett vor dem Verschleiß bewahrt werden. Die aggregierten Daten ermöglichen eine unabdingbare, umfangreiche Echtzeitüberwachung des Netzzustandes. Sofern dabei Schwellwerte unterschritten werden, muss ein Maintenance Event erstellt werden. Diese Echtzeitüberwachung aus größerer Distanz zum überwachten Objekt wird in der Literatur auch Remote Maintenance genannt.
- **Control Grid:** Parallel zur Überwachung muss das Netz ständig gesteuert werden, denn das Zu- und Abschalten von zusätzlichen Erzeugern verändert den Stromfluss. Parameter wie Spannung und Frequenz müssen stabil gehalten werden.
- **Manage Outage:** Bei einem Ausfall wird durch ein Monitor-Grid bereits das Maintenance-Ereignis erstellt. Allerdings müssen selbstheilende Netzwerke dazu in der Lage sein, sich zwischenzeitlich rekonfigurieren zu können, um somit die ausgefallene Einheit zu isolieren und die größtmögliche Anzahl an Verbrauchern zu versorgen. Hierbei wird auch von Self-healing Grid gesprochen (EATON, 2017).
- **Store Meter Data:** Dieser Service beschreibt das Speichern von Energiezähler-Daten von privaten Haushalten sowie der Industrie.

- **Report Load:** Dieser Service erstellt einen Report über die Last im Netz.
- **Predict Demand:** Dieser Service sagt die Nachfrage auf Grundlage der Verbrauchshistorie aller Kunden voraus, um so Aussagen für den Einkauf treffen zu können.

3.1.2 Ausbau und Maintenance

Der Bereich des Ausbaus und Maintenance fasst jegliche geplanten Installations- und Wartungsarbeiten zusammen. Hierbei setzt die Planung der Wartung auf einer durch Analyse vorausbestimmten erwarteten Lebenszeit eines Verschleißteils auf. Dies steht in engem Zusammenhang zu *Predictive Maintenance*, wobei die Wahl des optimalen Wartungszeitpunktes einen bevorstehenden Ausfall verhindert. In der Literatur wird dann auch von *Outage Prevention* gesprochen. Hierzu werden zusätzlich Temperatur- oder Vibrationssensoren an Geräten verbaut, um Abweichungen besser festzustellen (Evsan, 2016). Zusätzlich ist die Analyse des vom Sensor gemeldeten Zustands (*Condition-based Maintenance*) wichtig, da sich Ausfälle auch zufällig ereignen können (Palem, 2013). Unter Installationsarbeiten fällt bspw. das Verbauen von neuartiger Smart-Metering-Technologie im existierenden Netz.

- **Manage Events:** Verwaltet geplante Termine für Wartungs- und Installationsarbeiten.
- **Optimize Events:** Optimiert und bündelt Termine, um die Ausfallzeit gering zu halten.
- **Parts Availability:** Prüft die Verfügbarkeit von Ersatzteilen.
- **Forecast Part Durability:** Sagt die Haltbarkeit von Verschleißteilen auf der Grundlage von historischen Daten und Meldungen von Sensoren voraus.

Expansion ist für den Ausbau des Netzes verantwortlich. Der Ausbau wird von der Bundesnetzagentur reguliert.

- **Create Grid Expansion Plan:** Auf Grundlage des von der Bundesnetzagentur erstellten Szenariorahmens, der ein politisch-regulatives Framework darstellt (Smart Grid Coordination Group, 2012), muss ein Plan für die Entwicklung des Netzes erstellt werden, welcher bereits als Entwurf der Öffentlichkeit vorgestellt und kollaborativ weiterentwickelt werden muss.
- **Simulate Plan:** Dieser Service simuliert den Entwurf mittels historischer Verbrauchsdaten und prüft seine Umsetzbarkeit sowie einen rechnerischen Vorteil gegenüber anderen Entwürfen.

3.1.3 Verwaltung

Kundeninteraktion

Interaction ist klar kundenorientiert und umfasst sowohl klassische Marketingaufgaben als auch das Beschwerdemanagement. **Lead Generation** kümmert sich um die Identifikation potenzieller Kunden sowie deren Aufnahme in den Kundenbestand. Dies geschieht anhand von zwei Business Capabilities: *Identify Lead*, die Auswahl potenzieller Kunden bzw. -gruppen und *Acquire Lead*, die Aufnahme in Bestand und Kategorisierung des Kunden. **Customer Engagement** dient der Maximierung von Kundenzufriedenheit sowie der langfristigen Kundenbindung. Dabei werden auf Basis des Kundenverhaltens gezielte Maßnahmen ergriffen. Eine Besonderheit der Energiebranche der Zukunft ist der Einbezug konkreter Verbrauchsdaten und Nutzerpräferenzen. Die Verbrauchsdaten werden anhand von *Smart Meters* erhoben. Diese stellen Messstationen dar, die mit allen Geräten eines Nutzers verbunden sind und deren Energieverbrauch in zumindest stündlichen Intervallen an den Netzbetreiber senden (Fang, Misra, Xue & Yang, 2011). Gleichzeitig können diese *Smart Meter* auch Informationen vom Betreiber über Belastungszeiten und Strompreise erhalten und anhand dieser Daten die verbundenen Geräte im Haushalt steuern. *Smart Meters* sind durch die Nutzung sogenannter Smart Meter Gateways mit Wide Area Networks (WAN) verbunden, über welche die Daten an den Betreiber gesendet werden können (Müller, 2011). Aufgrund ihrer Funktion als zentrale Schnittstelle auf Kundenseite ist die Erstellung technischer Integrationsprofile hier kritisch (IEC-62559, 2013).

- **Establish Engagement Plan:** Generierung eines verkaufs- und supportbezogenen Maßnahmenplans für jeden Kunden.
- **Perform Engagement:** Durchführung einer Maßnahme (Kontaktaufnahme) im Rahmen des Maßnahmenplans.
- **Monitor Customer Behavior:** Überwachung und Dokumentation der Nutzerreaktion auf Maßnahmen sowie die Verwaltung von Kundenpräferenzen.
- **Analyze Consumption:** Untersuchung von Nutzerverbrauchsdaten, die über *Smart Meters* übermittelt werden. Auf Basis dieser Analyse und des bisherigen Nutzerverhaltens wird ein Maßnahmenplan abgeleitet/angepasst.

Kundenservice

Customer Service umfasst schließlich den Support und das Beschwerdemanagement. Dabei wird jede kundenseitige Anfrage als Request registriert, danach entsprechend klassifiziert und an die jeweilige Abteilung zur Lösung weitergeleitet.

- **Handle Customer Request:** ist für die Registrierung, Klassifizierung und Weiterleitung und Lösung der jeweiligen Kundenanfrage/Beschwerde verantwortlich.
- **Evaluate Customer Request:** nach Bearbeitungsabschluss jeder Anfrage wird diese zur internen Identifikation von Defiziten bzw. Optimierungspotenzialen untersucht.

Applikationen gliedern sich in einen Lead Manager, einen Engagement Manager (Salesforce, 2016), der Maßnahmen bezüglich des Kontakts zum Kunden erlaubt sowie den Consumption Manager, welcher Verbrauchsdaten auf Muster untersucht und zur Ableitung von Maßnahmen aufbereitet (Flath, Nicolay, Conte, van Dinther & Filipova-Neumann, 2012). Zuletzt kümmert sich ein Complaint Manager um die Abwicklung von Kundenanfragen. Ein umfangreiches Customer Relationship Management System (Salesforce, 2016) stellt die Infrastruktur zur Umsetzung aller marketingbezogenen Anwendungen zur Verfügung. Zur Analyse und Aufbereitung von Verbrauchsdaten wird auch hier das Meter Data Management System (MDMS) eingesetzt.

Kunden-Management

Kunden Management kümmert sich um die administrative Verwaltung und Fakturierung an einzelne Kunden. *Demand Profiling* kümmert sich darum, zusätzlich zur Verrechnung von Verbrauchsmengen im Rahmen der vorausschauenden Planung von Bedarfen und Kapazitäten für jeden Kunden ein Nachfrageprofil zu erstellen. Dieses besteht aus einer kundenspezifischen Nutzenfunktion, die aus Verbrauchsmustern der verwendeten Geräte errechnet wird. Es lässt sich beispielsweise die Nutzenfunktion eines Kunden auf Basis von Energieverbrauch (Li, Chen & Low, 2011) und persönlichen Präferenzen errechnen. Letztere umfassen zumeist den Kundentyp (privater Haushalt, Geschäftskunde), Preissensibilität und Hauptnutzungszeiten.

Eine *Billing Engine*, die Teil des *Account Management Systems* (Software Systems, 2017) ist, befasst sich mit der Verrechnung von Verbrauchsdaten, während die *Demand Profiling Engine* für die Erstellung und Aktualisierung von Verbrauchsprofilen bzw. deren zugrundeliegenden Verbrauchsmustern und Nutzenfunktionen zuständig ist. Die *Demand Profiling Engine* ist dem *Consumption Manager* der *Subdomain Custo-*

mer Interaction sehr ähnlich und nutzt dazu sowohl das *Meter Data Management System* als auch *Customer-Relationship Management-System*, um die Zuordnung von Profilen vornehmen zu können.

3.1.4 Prognose und Analyse

Risikomanagement

Auch in der Energiebranche wird im Zuge der Digitalisierung und Deregulierung der Märkte Risikomanagement wichtiger. Der flächendeckende Einsatz von Sensornetzwerken und Smart Meters sowie die dahinterliegende IT-Infrastruktur erfordern den Einsatz ausgeklügelter Sicherheitsmechanismen, um Datenintegrität und -authentizität sicherzustellen (Smart Grid Coordination Group, 2012). Hierfür spielt wiederum die Definition technischer Use Cases eine signifikante Rolle (IEC-62559, 2013). Darüber hinaus stellt auch die Volatilität deregulierter Energiemärkte ein nicht unerhebliches Risiko für Energiebetreiber dar. Schlussendlich unterliegen Unternehmen der Einhaltung rechtlicher Rahmenbedingungen, welche sich insbesondere im Zuge der Weiterentwicklung des Elektrizitätsnetzwerks verändern werden.

Weiterhin kommt das Monitoring der Verteilungsnetzwerke und Umspannwerke hinzu, um Anomalien zu erkennen. Die Überwachung und Sicherstellung der Konnektivität von Übertragungsnetzwerken zählen wie nachfolgend beschrieben auch dazu.

- **Manage Devices:** Verwaltung der im Smart Grid verwendeten Geräte wie bspw. Smart Meters und Sensoren in Verteilungsnetzwerken und Umspannwerken.
- **Monitor Networks:** Überwachung und Sicherstellung der Konnektivität von Übertragungsnetzwerken zur Aufgabe.
- **Detect Fraud:** Identifikation der Manipulation von Verbrauchsdaten mithilfe von Mustererkennung auf Basis neuronaler Netze (Ford, Siraj & Eberle, 2014).
- **Prevent Fraud:** Kontinuierliche Verbesserung der Betrugserkennung und Sicherheitsinfrastruktur durch Analyse und Aufnahme von Betrugsfällen in bestehende Muster.

Um die Manipulation von Verbrauchsdaten durch unbefugte Personen zu verhindern, bedarf es besserer Betrugserkennungs- und -präventionsmaßnahmen, insbesondere innerhalb der *Advanced Metering Infrastructure* (Jiang et al., 2014).

Market & Pricing

Angesichts der Tatsache, dass der globale Energiemarkt großen Dynamiken unterliegt, werden in der Domain Market & Pricing mithilfe umfassender Marktanalysen und Trading-Tools Energiepreise errechnet und an Kunden kommuniziert. Eine Besonderheit stellt zudem das Demand Response Management dar, welches in erster Linie für die Reduzierung von Belastungsspitzen verantwortlich ist (Qian, Zhang, Huang & Wu, 2013). In einem typischen Use Case werden dazu in Abhängigkeit der Auslastung variierende Preise an Kunden bzw. deren Geräte kommuniziert, um ökonomische Anreize zur Nutzung von weniger ausgelasteten Zeitfenstern zu schaffen (IEC-62559, 2013). Somit ist der Strompreis kein fixer Tarif, sondern unterliegt Schwankungen, die nicht nur von Angebot und Nachfrage, sondern auch von der Netzauslastung abhängig sind (Faruqui & Sergici, 2010). Daraus ergibt sich das Konzept des Dynamic Pricing – für jeden Zeitpunkt und geographischen Raum existiert ein Preis pro Verbrauchseinheit, der sowohl die Netzauslastung konstant halten als auch den Nutzen für Nutzer und Anbieter maximieren soll.

- **Perform Market Analysis:** Sammeln von Marktinformationen sowie Durchführen von Analysen aktueller Angebots- und Nachfrageentwicklungen.
- **Perform Market Prediction:** Vorhersage von Marktentwicklungen und Strompreisen (Conejo, Contreras, Espínola & Plazas, 2005) mithilfe stochastischer Vorhersagemethoden.
- **Optimize Prediction Method:** Laufende Verbesserung von Marktvorhersagemethoden durch Ansätze maschinellen Lernens (Conejo et al., 2005).

Mithilfe zugrundeliegender Markt- und Auslastungsanalysen werden dynamische Preise errechnet, die in Folge an *Smart Meter* und Geräte des Kunden weitergeleitet werden.

- **Calculate Prices:** Kalkulation von dynamischen Preisen.

Konsum- und Nachfrageprognosen

Das *Demand Response Management* ist besonders wichtig, da es maßgeblich für die Sicherstellung der Netzstabilität verantwortlich ist (Sæle & Grande, 2011). Um die Nachfrage beeinflussen zu können, muss zunächst die Auslastung von Übertragungs- und Verteilungsnetzen gemessen werden, bevor Preisänderungen berechnet werden können. Die aktualisierten Preise werden danach an die Nutzer und deren Geräte gesendet.

- **Calculate Demand Peaks:** ist für die Errechnung von Belastungsspitzen sowie für das Aggregieren zu regionsspezifischen Modellen zuständig. Dies geschieht unter Zuhilfenahme der in der Domain Energy Generation verwalteten Lastprofile (vgl. 4.1.1)
- **Transmit To Meters:** Kommunikation der Preisänderungen an *Smart Meter* und Nutzergeräte.

Der länderübergreifende Handel mit Energie ist nicht nur wesentlich für die Setzung von Strompreisen verantwortlich, sondern ist besonders angesichts der immer strikter werdenden Auflagen zum Energiemix von Bedeutung. Viele zentraleuropäische Länder importieren bspw. erneuerbare Energie aus skandinavischen Ländern, welche aufgrund ihrer geografischen Lage und den entsprechenden Wetterbedingungen günstigere Windenergie produzieren können (Izadian, Girrens & Khayyer, 2013).

- **Manage Trade:** Verwaltung und Steuerung länderübergreifender Handelsaktivitäten.

Eine Market Monitoring Engine (Instream Oy, 2017) wird eingesetzt, um eine Marktanalyse und Vorhersage zu machen.

3.2 Gesundheit

Im Bereich Gesundheit existiert eine Vielzahl unterschiedlicher Einsatzmöglichkeiten von Ansätzen des maschinellen Lernens. Diese reichen von der Früherkennung von Krankheiten oder Anomalien, bspw. diverser Arten von Krebs (Cruz & Wishart, 2006), über die Entwicklung von Medikamenten (Murphy, 2011), die Gestaltung von Heilungsplänen, bis hin zur Unterstützung sowie Bilderkennung bei Operationen (Lundberg et al., 2018).

3.2.1 Prognosen und Erkennung

Anamnese und Behandlungen

Evidenzbasierte und medikamentöse Behandlung erfordern ein Maß an Präzision, das den Patienten hilft, Vertrauen in ihren Arzt zu entwickeln. In diesem Bereich können die manuellen und auf Erfahrungen beruhenden Beurteilungen des Arztes durch KI ergänzt werden.

Aufgrund der Vielzahl an Informationsverarbeitungsmöglichkeiten, die in Bereichen wie Bioinformatik erforderlich sind, ist der Einsatz von KI-basierten Algorithmen und Lösungen unumgänglich. KI-Lösungen im Gesundheitswesen, in der Medizin und in

der Biotechnologie umfassen unterstützende Systeme zur Identifizierung genetischer Risiken aus groß angelegten Genomstudien, die Vorhersage der Sicherheit und Wirksamkeit neu eingeführter Medikamente zur Entscheidungsunterstützung bei medizinischen Beurteilungen sowie die Anpassung der Arzneimittelverabreichung an den einzelnen Patienten (Denecke, Lutz Hochreutener, Pöpel & May, 2018).

Anomalieerkennung

Aktuell ist die Gaschromatographie gekoppelt mit Massenspektrometrie eine der weltweit meistverbreiteten Routinetechnologien, mit welcher Biomarker erkannt werden können. Biomarker sind messbare biologische Prozesse, welche eine pro- oder diagnostische Aussagekraft haben und als Indikator für Krankheiten herangezogen werden können. Hierbei wird mit dem Gaschromatograph das zu untersuchende Stoffgemisch aufgetrennt und mit dem Massenspektrometer die einzelnen Komponenten identifiziert. Da es jedoch nicht genug authentifizierte Referenzsubstanzen gibt, welche für eine Identifikation nötig sind, wird mit Hilfe von überwachten maschinellen Lernansätzen versucht weitere Massenspektralmarkierungen anhand schon vorliegender Strukturen von Metaboliten zu identifizieren (Hummel, Strehmel, Selbig, Walther & Kopka, 2010).

Ausbruchsprognose am Beispiel von Malaria

Malaria ist eines der größten Probleme der öffentlichen Gesundheit in Indien und die weltweit häufigste Infektionskrankheit. Die Verbreitung der Krankheit wird durch Faktoren wie Temperatur, Niederschlag, Flut oder Dürre, aber auch durch menschliche Migration und Bautätigkeiten beeinflusst. Die frühzeitige Vorhersage eines Malaria-Ausbruchs ist der Schlüssel zur Kontrolle der Ausbreitung und damit der Reduzierung von Infizierten und Toten. Forscher aus Indien haben es geschafft, mittels einer Support Vector Maschine den Ausbruch von Malaria bereits 15-20 Tage im Voraus vorherzusagen (Sharma, Kumar, Lakshmi Panat & Karajkhede, 2015).

3.2.2 Pharma

Biowissenschaften

Unter Biowissenschaft, auch Life Science genannt, ist die Forschung rund um die Natur und deren Lebewesen bekannt. Life Science geht über die traditionelle Biologie hinaus und beschäftigt sich auch mit Randgebieten wie etwa der Chemie oder der Medizin. Durch technologische Weiterentwicklungen, wie etwa in der Robotik, erhöht sich die Anzahl an anfallenden und auszuwertenden Daten (Ramsundar, Eastman, Walters & Pande, 2019). Der Einsatz von maschinellen Lernsystemen wird daher auch für den

Bereich der Biowissenschaften zunehmend relevant. Gerade die Molekularforschung kann durch den Einsatz von neuronalen Netzen profitieren. So können diese beispielsweise dazu verwendet werden, um das Zusammenspiel zwischen einem neuen Medikament mit den Proteinen eines Lebewesens zu evaluieren (Ramsundar et al., 2019). Solche Möglichkeiten der Evaluation sind hilfreich, da umfangreiche und belastbare klinische Studien mit hohen Kosten verbunden sind. Eine vorzeitige Evaluation mittels eines ML-Modells kann eine Kostenexplosion in der Forschung vermeiden. Weiterhin ermöglicht die Analyse oder Simulation genetischer Strukturen neue Angriffspunkte für die Therapie oder Heilung bisher noch nicht bewältigter oder noch kommender Krankheiten. Damit haben Erfolge beim Einsatz von KI-Systemen im Bereich der Biowissenschaften einen nachhaltigen Einfluss auf unser Leben.

3.2.3 Psychologie

Selbstmordvorhersage

Um Warnung zu geben und rechtzeitig psychiatrische Maßnahmen ergreifen zu können, wird versucht eine Suizidgefahr rechtzeitig zu erkennen. Mit Hilfe von maschinellem Lernen kann die Selbstmordabsicht eines Patienten weitestgehend genau vorhergesagt werden. Anhand von Datensätzen, welche unter anderem Alter, Geschlecht, und Medikamentierung aus der Krankenakte enthalten, kann beispielsweise mit 80% Genauigkeit vorhergesagt werden, ob ein Patient innerhalb der nächsten zwei Jahre Selbstmord begeht (Goldhill, 2018).

3.3 Öffentlicher Sektor

Die Einsatzmöglichkeiten von maschinellem Lernen im Bereich des öffentlichen Sektors sind vielfältig. Beispielsweise wird ML zur Vorhersage von Gerichtsentscheidungen (Aletas, Tsarapatsanis, Preoțiu-Pietro & Lampos, 2016) und Rechtsprechung, im Bereich des Katastrophenschutzes, zur Analyse der Nachrichten in den sozialen Medien oder aber zur Suche von Personen im Gelände mittels Drohnen und Bildererkennung eingesetzt.

3.3.1 Recht und Ordnung

Rechtsprechung

Einige der Bereiche, in denen KI-Rechtsverfahren verbessern kann, umfassen eine verbesserte Ermittlung und Analyse basierend auf der Rechtsprechung und die Formulierung von rechtlichen Argumenten basierend auf der Identifizierung relevanter Beweise (PwC, 2017).

Vorhersagende Polizeiarbeit (Predictive Policing)

Beim *Predictive Policing* geht es darum, potenzielle Verbrechen im Vorfeld zu verhindern oder auch vorherzusagen, wann und wo ein Täter erneut zuschlagen wird. Als Grundlage werden Datensätze zu polizeilich registrierter Kriminalität verwendet. Diese werden mit sozialräumlichen, kalendarischen und meteorologischen Daten ergänzt. Bereits vor 20 Jahren wurden derartige Daten – in deutlich geringerem Ausmaß – in Excel-Tabellen erfasst und ausgewertet. Dieser Vorgang ist zeitaufwendig und auf eine maximale Komplexität beschränkt. Durch den Einsatz von ML-basierter Software kann die Masse an verfügbaren Daten effizient ausgewertet werden. Dies erlaubt das schnellere und bessere Ziehen von Schlüssen. Auf Basis dieser können beispielsweise Streifenwagen bevorzugt in Gegenden geschickt werden, in denen eine erhöhte Wahrscheinlichkeit für eine bevorstehende Straftat besteht (Krassmann & Egbert, 2018; Kreml, 2018).

Cybersicherheit

Im Bereich der Cybersicherheit wird seit vielen Jahren an der Verbesserung von Angriffserkennungssystemen gearbeitet. Ziel ist es immer, Angriffe gegen Computersysteme oder Rechnernetze zu erkennen. Der Ansatz der Anomalie-Erkennung basiert darauf, dass Angreifer sich deutlich anders verhalten als autorisierte Benutzer. Bei einer Vielzahl von Nutzern und abweichendem Verhalten zwischen diesen ist es ohne effiziente Softwarelösungen nahezu unmöglich, ein Muster im Verhalten von Benutzern zu erkennen. Damit ist es dann noch schwieriger zu erkennen, wann eine Anomalie und somit ein Angriff vorliegt. Durch den Einsatz von ML können die zugrundeliegenden Muster und dadurch potenzielle Angriffe erkannt werden (Omar, Ngadi & Jebur, 2013).

Im Bereich Verteidigung und Sicherheit dient die Erkennung von anomalem Verhalten dem Schutz von Infrastrukturen wie Flughäfen, Kraftwerken und Wirtschaftssektoren, welche anfällig für Angriffe sind. Das Erkennen von anomalem Verhalten von Individuen sowie die Vorhersage von Infrastrukturstörungen aufgrund natürlicher oder künstlicher Ursache wird dabei durch die Verwendung von verteilten Sensoren und Mustererkennung ermöglicht.

Cyber Grooming

Cyber Grooming bezeichnet den kriminellen Vorgang einer Person, sich mittels sozialer Netzwerke an eine minderjährige Person mit dem Ziel zu wenden, diese zum gemeinsamen Geschlechtsverkehr zu bringen. Es ist zu beobachten, dass der Prozess der Annäherung eines Pädophilen an eine minderjährige Person einem unterteilbaren Muster folgt, welches typischerweise in Gesprächen auftritt (Meyer, 2015). Aufgrund

der hohen Anzahl täglicher Gespräche über das Internet muss ein automatisiertes System entscheiden, ob ein Gespräch als alarmierend anzusehen ist oder nicht. Nicht nur Pädophile verwenden in ihren Gesprächen sexuelle Ausdrücke, auch die Sprache der Jugendlichen enthält diese Wörter, allerdings meist in einem anderen Kontext. Das richtige Erkennen des Kontexts, in welchem das Gesagte steht, macht die Klassifizierung zu einer schwierigen Aufgabe. ML kann bei dieser Klassifizierung helfen, indem die Wörter und der Kontext untersucht werden.

3.3.2 Organisation und Management

Katastrophenmanagement und Notfallplan

KI-basierte Ansätze zeigen ein bemerkenswertes Potenzial bei der Unterstützung von Kontroll- und Abhilfemaßnahmen nach Umweltkatastrophen oder von Menschen verursachten Katastrophen auf. Nach derartigen Ereignissen erhöht sich normalerweise der Kommunikationsbedarf und Netzwerke können überlastet werden. Um die Kontinuität des Netzwerkdienstes inmitten von Katastrophen zu gewährleisten, kann eine KI bei der Optimierung von Mobilfunknetzen und einer intelligenten Breitbandzuweisung helfen.

Unbemannte Drohnen und Satellitenübertragungen in Kombination mit Bildverarbeitung und -erkennung können zur Bewertung und Vorhersage von Infrastrukturschäden verwendet werden. Katastrophenteams können auf der Grundlage von diesen Informationen strukturiert und aktiv gesteuert werden. Eine KI kann auch für die Analyse von sozialen Medien verwendet werden. Hierbei werden ortsspezifische Dringlichkeiten gemessen und gezielte Warnungen gesendet, um somit den Verlust von Leben und Eigentum zu minimieren.

3.3.3 Bildung

Adaptive Betreuung

In weiten Teilen des Landes herrscht ein Mangel an Akademikern und Lehrern, wenn es darum geht, den Schülern einen effektiven Unterricht mit unterschiedlichsten sozialen und kognitiven Fähigkeiten zu ermöglichen. KI-Lösungen können durch adaptive Betreuung, basierend auf der Aufnahmebereitschaft der Schüler und einer genauen Einschätzung der Entwicklung der Schüler, unterstützend eingreifen und dadurch den Lehrer im Klassenzimmer komplementieren.

3.4 Mobilität

Im Industriesektor der Luftfahrt bestehen vielversprechende Einsatzzwecke von maschinellen Lernmethoden. Wie Abbildung 3-2 zeigt kann die Industrie zunächst in die Bereiche Buchung, Flughafen und Flugzeug unterteilt werden. In jedem dieser Bereiche finden sich Teilbereiche die sich in die Kategorien Sicherheit, Kundenservice und Steuerung bzw. Automatisierung eingliedern lassen. Nachfolgend finden sich die in der Grafik grau hinterlegten Bereiche im Detail beschrieben.

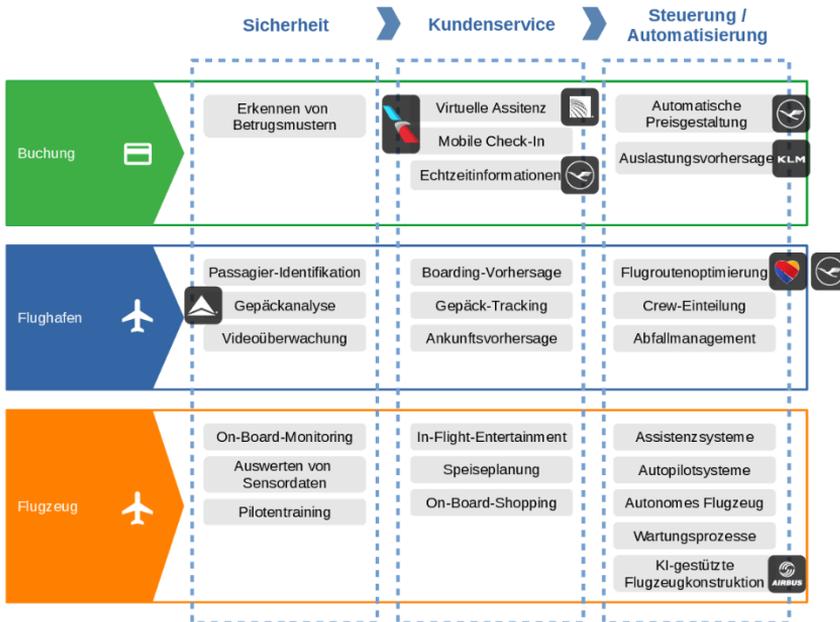


Abbildung 3-2: Übersicht ML-Anwendungsbereiche in der Luftfahrt (Eigendarstellung auf Basis von Mousa (Mousa, 2019, Folie 46)).

3.4.1 Luft- und Raumfahrt

3.4.1.1 Flugbuchung

In der Anbahnung, der Buchung und der Abwicklung von Flügen ist aufgrund der Abbildung von Prozessen ein Potenzial für die Anwendung von KI-Ansätzen zu finden, da Prozessdaten zumeist online leicht erfasst und bereitgestellt werden können. Es zeigen sich somit bspw. folgende Anwendungsszenarien für KI in der Flugbuchung.

Erkennen von Betrugsmustern

Aufgrund der enormen Anzahl an Transaktionen bei Ticketkäufen ist die Luftfahrt ein beliebtes Angriffsziel für Betrüger (IATA). Aus diesem Grund hat sich das Projekt „Industry Fraud Prevention“ als Ziel gesetzt, das Betrugsrisiko bei den Fluggesellschaften besser einschätzen zu können. Betroffen sind hierbei Bereiche wie traditionelle Kartenzahlungen, Vielfliegerprogramme und genereller Cyber-Betrug.

Virtuelle Assistenz

Um gängige Fragestellungen an die Fluggesellschaften schnell und zielgerichtet beantworten zu können, setzen Fluggesellschaften unter anderem auf virtuelle Assistenzsysteme. Im September 2017 stellte United Airlines seinen „United skill“ vor, dessen Ziel es ist, die Fragen rund um United Flüge mittels natürlichsprachlicher Interaktion zu beantworten (Mousa, 2019, Folie 47). Neben natürlichsprachlichen Systemen existiert auch die Option, Informationsabfragen über Flüge oder Hilfestellungen bei einem Online-Ticketkauf mittels Chatbot-Systeme zu unterstützen.

Mobile Check-In

Um den Check-In-Prozess des Flugpassagiers zu beschleunigen, setzen Fluglinien auf mobile Check-In-Systeme, mit denen sich der Kunde selbstständig für einen Flug einchecken kann. Der Vorteil liegt darin, dass der Passagier direkt zum Gepäckschalter gehen kann, ohne lange Wartezeiten für einen vorherigen Check-In-Prozess hinnehmen zu müssen. Die Unterstützung für einen mobilen oder Online-Check-In ist nicht nur von der Fluggesellschaft, sondern auch vom jeweiligen Flughafen abhängig (Airlines, 2020).

Echtzeitinformationen

Den Kunden mit Echtzeitinformationen zu anstehenden oder verspäteten Flügen zu informieren ist nicht nur eine nette Geste von Fluggesellschaften, sondern sorgt auch für eine Optimierung des Ablaufs. Der Passagier kann sich besser darauf einstellen. Auch Personal am Boden profitiert von Echtzeitinformationen zu einem Flug. Um das Ausmaß dieser Daten verarbeiten und wertvolle Schlüsse aus den Datenmengen ziehen zu können, werden ML-Methoden eingesetzt. Lufthansa hat hierzu ein Projekt namens „Deep Turnaround“. Mit Turnaround-Management wird der Prozess bezeichnet, welcher dazu führen soll, die Standzeiten der Flugzeuge so gering wie möglich zu halten. Das Projekt der Lufthansa informiert alle Beteiligten über den aktuellen Status einer Maschine (Lufthansa, 2019). So können bei Bedarf schneller Gegenmaßnahmen eingeleitet werden.

Automatische Preisgestaltung

Die Nachfrage für einen Flug entscheidet über dessen Preis. Automatisierte Systeme sind in der Lage, anhand der aktuellen Auslastung den Preis dynamisch zu bestimmen. Hierfür können KI-basierte Systeme eingesetzt werden, die Faktoren wie Urlaubszeiten, Wetter, Messetermine, Kongresse und vieles mehr für eine Preisgestaltung berücksichtigen. Die Systeme assoziieren das Flugzeug dann mit Buchungsklassen. Ist die Nachfrage hoch, so werden günstige Flugklassen nicht angeboten (Hecking, 2016). Doch die automatische Preisgestaltung dient nicht nur der Gewinnmaximierung. Sie verhindert auch, dass Flüge quasi leer abfliegen müssen und hat damit nicht nur eine wirtschaftliche, sondern auch imagewahrende und ökologische Motivation.

Auslastungsvorhersage

Mithilfe von *Passenger Flow*-Modellen werden der Passagierfluss optimiert und die dafür notwendigen Kapazitäten ermittelt. Fluglinien haben in der Vergangenheit teilweise sogenannte *Overbookings* vorgenommen (Püttmann, 2017), um kurzfristige Ausfälle (die sie statistisch vorhersagen) im Vorfeld zu kompensieren. Dies ist jedoch nicht optimal, falls die eingeplanten Ausfälle nicht zustande kommen und Passagiere nicht mitfliegen können. ML-basierte Vorhersagemodelle können genutzt werden, um Ausfälle präzise vorherzusagen, sodass eine sicherere Kapazitätsplanung realisiert werden kann.

3.4.1.2 Flughafen

Da im Bereich des Flughafens diverse Anwendungsbereiche für KI-Systeme gefunden werden können, finden sich diese entsprechend thematischer Cluster nachfolgend dargestellt.

Passagier-Identifikation

Vergangene Ereignisse des Terrorismus haben dazu geführt, dass die Sicherheit an den Flughäfen massiv erhöht wurde. Neben einer höheren Präsenz von Sicherheitskräften wird auch stetig die technische Unterstützung verbessert. Beispielsweise hat Delta Airlines hierfür am Hartsfield-Jackson-Flughafen ein System zur biometrischen Erkennung von Fluggästen installiert (Contributor, 2019). Das AI-System entscheidet darüber, ob der Fluggast an Board darf oder nicht. Das Vorzeigen eines Reise- oder Personalausweises ist somit hinfällig. So werden sowohl die Wartezeit für den Gast verkürzt, als auch die Sicherheit bei dessen Identifikation gesteigert.

Gepäckanalyse

Ein weiterer Bereich, welcher sowohl den Abwicklungsprozess beschleunigt als auch die Sicherheit nachhaltig steigert, ist die KI-gestützte Analyse von Gepäckstücken. Heutzutage liefern moderne Scan-Mechanismen gute Ergebnisse, wenn es darum geht, bestmögliche Bilder des Innenlebens von Gepäckstücken zu liefern. Doch auch wenn das Bild noch so gut ist, muss ein Mensch eine Klassifikation vornehmen. Im Zweifelsfall heißt es, den Gast zurückrufen und das Gepäckstück aufmachen lassen. Das ist nicht nur dem Gast unangenehm, sondern führt auch zu Verzögerungen im Prozess und eventuell zu einer Verschiebung des Abflugzeitpunkts. Systeme wie beispielsweise *BaggageAI* können dabei helfen, unzulässige Gegenstände im Gepäck zu identifizieren (*BaggageAI*, 2020). Dabei kann das System in den bestehenden Scanprozess integriert werden. Bevor das Bild dem Mitarbeiter präsentiert wird, geht es zur Objektidentifikation in das AI-System. Das resultierende Videosignal liefert nicht nur den Scan, sondern zeigt gleich identifizierte Gegenstände auf. Erst kürzlich wurde von Indiens Flughafenaufsicht angekündigt, dass in acht verschiedenen Flughäfen KI-gestützte Gepäckanalyse installiert werden soll (*Yolmo*, 2020). Auch wenn die Analyse mittels KI nur unterstützend eingesetzt werden kann, weist diese gegenüber dem Menschen einige entscheidende Vorteile auf. KI-Systeme werden nicht müde, sind nicht abgelenkt, in Gedanken oder krank, wie es mit jedem menschlichen Mitarbeiter passieren kann. Zudem können die Algorithmen weiter trainiert werden, um die entscheidende False-Positive-Rate zu verbessern.

Videoüberwachung

ML hilft nicht nur bei der Gepäckanalyse, sondern auch bei der grundsätzlichen Videoüberwachung im und um den Flughafen sowie Boardingprozess. Die Bilddaten der Kamera werden dann einem ML-Modell übergeben, welches z.B. eine Klassifizierung von Objekten auf der Szene vornimmt. So können bedrohliche Gegenstände bereits vor Beginn der Boardingprozedur identifiziert werden. Auch auffälliges Verhalten von Personen sowie das Erkennen möglicher Sabotageakte gehören zu den Aufgabebereichen. Eine beispielhafte Softwarelösung für diese Anwendungsfälle ist *SCANNERA* (*Scannera*, 2020). Die Software ist in der Lage, Bilddaten in Echtzeit zu verarbeiten und eine Musteranalyse vorzunehmen. Der Vorteil für z.B. Flughafenbetreiber ist es, dass die bestehende Infrastruktur für die Videoüberwachung nicht geändert werden muss. Stattdessen werden die Bilddaten direkt – entweder auf hauseigene Server oder in die Cloud – zur Analyse übertragen. Werden bedrohliche Objekte entdeckt, so können mit Echtzeitmeldungen die Sicherheitsdienste vor Ort informiert werden. Neben einer erhöhten Sicherheit können ML-gestützte Überwachungssysteme auch bei Aufgaben wie

etwa dem Zählen von Gästen helfen. So kann sich ein Flughafen besser auf die zu erwartende Auslastung einstellen.

Boarding-Vorhersage

Verzögerungen sind keine Seltenheit im Flugverkehr, da viele Komponenten wie verspätete Passagiere, technische Probleme, Wetterverhältnisse, funktionierende Abläufe am Boden, uvm. einen Einfluss auf den gesamten Ankunfts- und Boarding-Prozess haben. Es lässt sich feststellen, dass die Verzögerungen in den Boden-Prozessen einen höheren Einfluss auf etwaige Verspätungen haben als die in der Flugphase (Schultz & Reitmann, 2019). Der Einsatz von ML und insbesondere der Zeitreihenvorhersage kann dazu beitragen, proaktiv die Prozesse optimieren zu können. (Schultz & Reitmann, 2019) stellen fest, dass univariante Eingabevariablen nicht so vielversprechende Ergebnisse liefern, wie der Einsatz von multivarianten Eingabevariablen. Mithilfe eines LSTM-Netzwerks ist es den Autoren gelungen, erste Vorhersagen für den Boarding-Prozess zu machen. Die multivariate Eingangsvariable ist hierbei eine aggregierte Zusammensetzung aus Passagier-Bewegungen und operativen Prozessen wie etwa Wartezeiten im Gang oder das Einlagern von Handgepäck. Die Vorhersage kann dazu verwendet werden, um die Prozesse zu optimieren und insbesondere die, von den Passagieren, verursachten Verzögerungen besser einzuplanen.

Gepäck-Tracking

Der Verlust von Gepäckstücken ist für Passagiere eine ärgerliche Angelegenheit. Zwar sollten wichtige Dokumente und Gegenstände im Handgepäck transportiert werden, aber die Kleidung für die kommenden Tage befindet sich oftmals im aufgegebenen Zusatzgepäck. Insbesondere in der Urlaubssaison ist ein erhöhtes Aufkommen von Gepäckstücken, die verloren gehen, zu verzeichnen (LuggageHero, 2019). Es existieren unterschiedliche Ansätze des Trackings verlorengegangener Gepäckstücke. Traditionell markiert die Fluggesellschaft mittels Aufkleber, wem das Gepäckstück gehört. Bis allerdings eine Meldung hinsichtlich des Aufenthaltsortes des Gepäckstückes erscheint, müssen viele zeit- und somit kostenintensive Prozesse manuell angestoßen werden. Der Einsatz von RFID für Gepäckstücke versucht dieses Problem zu beheben (Prince, 2020). Der Nachteil an diesem Ansatz ist allerdings, dass die RFID-Chips wieder ausgelesen werden müssen. Zudem stellen die RFID-Chips eine zusätzliche Investition dar. Neben RFID existiert auch ein ML-Ansatz zur Identifikation und Ortsbestimmung von Gepäckstücken (Cloudflight, 2020). Die Idee ist hierbei das optische Identifizieren der aufgegebenen Gepäckstücke, um so ein Bewegungsprofil für diese zu erstellen. Sowohl bei der Aufgabe eines Koffers als auch beim Einladen, Ausladen und

Abholen können Kamerasysteme mithilfe eines ML-Modells z.B. in der Cloud die Identifikation desselben Gepäckstücks vornehmen. So kann zu jeder Zeit eine Aussage darüber getroffen werden, wo sich das Gepäckstück aktuell befindet.

Ankunftsvorhersage

Nach Recht der EU hat ein Passagier einen Anspruch auf Entschädigung, wenn die Verspätung mehr als drei Stunden beträgt und keine außerordentlichen Umstände, wie etwa politische Instabilität, ungünstige Wetterbedingungen sowie Sicherheitsrisiken vorliegen (europa.eu, 2020). Technische Probleme, die bei der Wartung auftreten oder durch die Nicht-Wartung verursacht werden, werden hier explizit von der Definition der außerordentlichen Umstände ausgeschlossen. Sowohl Passagiere, die ein Ziel rechtzeitig erreichen wollen, als auch die Fluggesellschaften, welche finanzielle Verluste vermeiden möchten, haben demnach ein großes Interesse Flugverspätung so gering wie möglich zu halten. Um das Vertrauen der Passagiere in die Fluggesellschaft auch im Falle einer Verspätung nicht weiter zu strapazieren, ist es hilfreich, eine weitestgehend verlässliche Ankunftsvorhersage zu stellen. So können Passagiere im Fall einer sich ankündigenden Verspätung diese frühzeitig in ihre Abläufe einplanen. Darüber hinaus gibt es auch kritische Anschlussflüge, welche zwingend erreicht werden müssen, da sonst z.B. erst am nächsten Morgen ein Flug ermöglicht werden kann. Übersteigt die Aufenthaltsdauer an bestimmten Flughäfen 12 Stunden (z.B. in Riad), dann hat das für Transit-Passagiere schwere Folgen, da sie plötzlich ein Visum benötigen, da deren Aufenthaltsdauer unter 12 Stunden sein muss. Mithilfe von Vorhersagemodellen und Warnsystemen können diese Probleme vermieden werden. Beispielsweise werden Passagiere mit einer betroffenen Staatsbürgerschaft, denen ein Aufenthalt über 12 Stunden an den speziellen Flughäfen nicht erlaubt ist, per Direktflug umbucht. Wie ML-Modelle bei der Herausforderung der Ankunftsvorhersage helfen können, zeigt Chakrabarty (2019) auf. Ihm ist es gelungen, ein Hyperparameter-getuntes ML-Modell für die Vorhersage von Flugverspätungen zu entwickeln und dabei eine Validationsgenauigkeit von über 85% zu erreichen. Dieses Ergebnis zeigt das Potential hinter dem Anwenden von ML-Modellen im Bereich der Ankunftsvorhersage.

Flugroutenoptimierung

Um die Effizienz des Flugprozesses zu erhöhen, können AI-Systeme bei der Optimierung der Flugrouten (oder Flugbahnen) assistieren. Im Allgemeinen werden zur Kontrolle und der Optimierung von Flugrouten vordefinierte Modelle und Regeln, basierend auf Expertenwissen, angewandt (European Organisation for the Safety of Air Navigation (EUROCONTROL), 2018). Bei der Planung der Flugrouten gilt es, Aspekte wie etwa Tag und Zeit, Wettersituation, politische Gegebenheiten, militärische Zonen

zu beachten. Statische Planung erweist sich trotz Expertenwissen als recht inflexibel. Die Flugroute kann sich jederzeit ändern, beispielsweise, wenn dynamisch eine schnellere Flugroute gefunden bzw. genehmigt wurde. Maschinelle Lernverfahren können aus historischen Daten lernen und so mögliche bzw. wahrscheinliche Flugbahnen für einen anstehenden Flug vorhersagen.

Crew-Einteilung

AI-Systeme können bei der intelligenten Einteilung von Crew-Mitgliedern helfen, indem geografische und damit auch wirtschaftliche Faktoren in Betracht gezogen werden. Personal von einem zum anderen Ort zu fliegen ist neben den eigentlichen Treibstoffkosten ein weiterer kostenintensiver Vorgang, welcher sich durch vorausschauende Planung optimieren lässt. Das Planen der Crew ist unerlässlich, da eine Cockpitbesatzung immer nur für einen bestimmten oder verwandten Flugzeugtyp qualifiziert ist (Barnhart et al., 2003). Diese Flugzeugtypen formen eine Flugzeugflotte. Für jede Flugzeugflotte gilt es, die Herausforderung der Einteilung einer Crew zu lösen. Bei der Wahl der richtigen Mitarbeiter sind zudem viele Faktoren wie etwa Fluglizenzen, Wohnort, Vertragsdetails, Vorschriften wie Ruhezeiten und Pausen, Urlaub, Krankheitsstand sowie Trainingszeiten zu beachten (Barnhart et al., 2003). Ein Beispiel für ein AI-System zur Einteilung von Crew-Mitgliedern kann in *Rosterize* gefunden werden (Rosterize, 2020). Rosterize hilft bei der automatischen Erstellung von Dienstplänen, indem die zuvor genannten Faktoren analysiert werden.

Abfallmanagement

Durch das stetige Wachstum an Fluggästen sehen sich viele Flughäfen einer zunehmenden Auslastung gegenüberstellt. Die gestiegene Anzahl an Fluggästen führt jedoch zu einer größeren Menge von Abfall. Pro Jahr wächst diese beispielsweise in der UK, um drei Prozent (Pitt, Brown & Smith, 2002). Gleichzeitig muss sich die Flugindustrie mit Themen wie z.B. grüner Energie und damit mit der Verantwortung für das Klima auseinandersetzen, da eine Wiederverwendung und Recycling in der heutigen Gesellschaft einen höheren Stellenwert aufweisen als zu Beginn des Fliegens. Durch den Einsatz von KI-gestützten Prozessen kann das Abfallmanagement optimiert werden, sodass die Fluggesellschaften auf Basis von Prognosen einer steigenden Auslastungen besser gegenüberstehen und Reinigungspläne verbessert ausstellen können.

3.4.1.3 Flugzeug

Da im Bereich des Flugzeugs diverse Anwendungsbereiche für KI-Systeme gefunden werden können, finden sich diese entsprechend thematischer Cluster nachfolgend dargestellt.

Assistenzsysteme

Um die Sicherheit der Flugpassagiere sowie des Personals zu gewährleisten, müssen bei einem Flug stets die Daten des Flugzeugs und dessen Umgebung ausgewertet werden. Sensordaten liefern genaue Einblicke in den aktuellen Zustand des Flugzeugs und dessen Komponenten und erlauben so Rückschlüsse auf aktuelles bzw. zukünftiges Verhalten dieser. Die zunehmende Anzahl an unterschiedlichen Daten während eines Fluges sinnvoll auswerten zu können, stellt das Personal vor eine große Herausforderung. ML-basierte Systeme können dabei unterstützen, die relevanten Informationen zu extrahieren oder direkt Vorhersagen auf diesen zu treffen. So werden beispielsweise den Piloten nur die elementar wichtigen Informationen präsentiert.

Flottenmanagement und Operation

Bei Organisation und Management von Flugzeugen befähigen KI-basierte Anwendungen das In-Flight-Entertainment und das On-Board-Shopping, indem Kundencluster gebildet und hoch-individualisierte Angebote von Medien und Produkten erstellt werden. Dies erstreckt sich bis hin zur Speiseplanung, welche für die gesamte Passagierliste des Flugzeugs optimiert wird, aber auch auf individueller Ebene den einzelnen Fluggast berücksichtigen kann.

Da auch Wartungsprozesse durch KI-basierte Systeme angereichert werden, können Ausfälle prognostiziert, defekte Teile erkannt und bevorzugt repariert werden. Weiterhin kann der Einsatz von Experten im Bereich des *Predictive Maintenance* gesteuert werden.

Das Nachtanken und Befüllen von Flugzeugen sowie auch das Anpassen der Flugzeugkonfiguration aufgrund von Marktveränderungen basiert auf der KI-angereicherten Planung (Mousa, 2019). Insbesondere die Vorhersage von Flugpreisen ermöglicht eine effiziente Organisation der gesamten Flotte an marktüblichen Konditionen (Lu, 2018; Papadakis; Rajankar & Sakharkar, 2019; Ren; Sengupta, 2019; Tziridis, Kalampokas, Papakostas & Diamantaras, 2017).

Individualisierung des Erlebnisses im Flugzeug

Empfehlungssysteme, wie sie im E-Commerce-Umfeld Anwendung finden, werden vermehrt auch zur Optimierung des Service-Angebots eingesetzt. So wird der Reiseverkehr zunehmend digitalisiert und KI-basierte Ansätze ermöglichen eine Individualisierung von Angeboten von der Reiseplanung bis zur Umsetzung direkt im Flugzeug (Rohleder, 2018).

Bei Kundenloyalitäts-Programmen, wie z.B. „Miles & More“, lassen sich Klassifizierungsverfahren einsetzen. Bestimmte Kunden mit hoher Loyalität bekommen dabei

zum Beispiel automatisch Geschenkgutscheine. Andere, abtrünnig werdende Kunden, werden mithilfe des Machine Learning-Systems identifiziert, damit sie per Mail ermuntert werden können, doch wieder etwas zu kaufen.

Assistenzsysteme

Die Steuerung des Flugzeugs erfolgt nicht nur durch eine Ergänzung von KI-basierten Autopilotensystemen, sondern reicht bis zur Realisierung autonomer Flugzeuge. Diese starten, steuern, landen und interagieren mit der Flugbehörde eigenständig und können durch Flugpersonal begleitet werden.

Assistenzsysteme unterstützen weiterhin Sicherheitskontrollen vor und während des Flugs. Ein ML-basiertes Erkennen von auffälligem Verhalten und sicherheitsgefährdenden Personen und Gegenständen erstreckt sich somit im Flughafen, aber auch im Flugzeug.

KI-gestützte Flugzeugkonstruktion

Die Konstruktion von Flugzeugen wird befähigt durch KI-basierte Simulationen, sodass wahrscheinliche Ausfälle und Defekte erkannt und kostengünstig vor der Serienproduktion behoben werden können. Auch eine Erkennung von Teilen zur Übernahme in Neuentwicklungen kann von KI-basierten Systemen evaluiert und Anpassungen einer Konstruktion vorbereitet werden.

3.4.2 Fahrzeug

Cognitive Car

Zur Integration einer stärker menschenzentrierten Sichtweise bei der Ausgestaltung von Assistenzsystemen ist es notwendig, kognitive Fahrassistenzsysteme und ihre Akzeptanz in der Gesellschaft im Allgemeinen zu berücksichtigen (Heide & Henning, 2006). Zu diesem Zweck wird auf Ansätze des Cognitive Computing (Poli et al.) zurückgegriffen. CC ermöglicht es, Wissen aus einer hohen Anzahl von Daten zu generieren. Diese großen Datenmengen werden durch die Digitalisierung von Steuerungs- und Assistenzsystemen realisiert und schlussendlich generiert. Wenn Computer wie Menschen denken können, können auch Fähigkeiten und Kenntnisse und deren Folgen durch CC in Zukunft beeinflusst werden (Sultanow, 2017). IBM Watson, Microsoft Cognitive Services, Google DeepMind sind einige Beispiele für CC. Dabei wird erlerntes Wissen gespeichert und kann später verändert werden (Wu et al., 2014). Cognitive beinhaltet die drei Kernelemente: verstehen, argumentieren und lernen. CC wird benötigt, um die Vielzahl an entstehenden Datenmengen in Automobilen zu kontrollieren und verschie-

dene Daten, Typen und Quellen zusammen zu bringen (Matthews, 2016). Dabei werden die Daten über Signal oder Data Miner aufgenommen. Um anschließend die Daten zu analysieren, wird ein Natural Language Processor (NLP) verwendet. NLP unterstützt Computer dabei, die menschliche Sprache zu verstehen. Neben der Interaktion mit dem Nutzer steht auch das Aufnehmen von weiteren externen Daten und Informationen im Zentrum von CC. Unter Cognitive Car wird, auf den vorherigen Ausführungen aufbauend, das Fahrzeug als technologisches System verstanden, welches sich selbst und seine Umwelt, sowie das Sammeln und Strukturieren von Informationen auf autonome Weise vollzieht, um den Fahrer zu entlasten und damit einhergehend einen bestimmten Grad an Autonomie anstrebt (Heide & Henning, 2006). Damit einhergehend sind Fahrerassistenzsysteme eng mit CC verknüpft. Navigationssystem, Stop-and-Go-Automatik, adaptive Fahrtensteuerung, Spurhalteassistenten oder Kollisionswarnungssysteme sammeln und generieren isoliert Daten, die integrativ ausgewertet einen großen Schritt zum denkenden Automobil leisten können, wenn die gesetzlichen Rahmenbedingungen geschaffen werden.

Connected Car

Hinter dem Begriff *Connected Car* verbirgt sich im Allgemeinen die kabellose Internetverbindung eines Fahrzeuges. Diese technische Grundlage ermöglicht verschiedene Anwendungsformen im Bereich Automotive, die von Unterhaltung über Informationscharakter bis hin zu Produktivitätssteigerungen reichen. So beschreibt der Begriff *In vehicle Infotainment* die Möglichkeit, Informationen im Fahrzeug bereitzustellen. Diese Möglichkeit bestand bereits mit eingebauten Radios, wird jedoch durch vernetzte Fahrzeuge in großem Umfang erweitert (Stanton, Plant, Roberts, Harvey & Thomas, 2016). Eine Möglichkeit das Auto mit dem Internet zu verbinden bietet Apple CarPlay. Darüber kann ein iPhone mit dem Bildschirm eines Autos verbunden und somit Funktionalität des Autos mittels des Smartphones gesteuert werden. Dabei können Funktionen wie Telefonie, Musik oder Navigation des iPhones genutzt werden. Auch die Nutzung von Kommunikationsdiensten wie bspw. *Whatsapp* wird somit im Auto ermöglicht. Über 100 Fahrzeugmodelle von verschiedenen Anbietern unterstützen CarPlay – unter ihnen finden sich auch namenhafte deutsche Automobilkonzerne wie Audi, BMW und Volkswagen (Apple, 2020a). Die Möglichkeit während einer Fahrt das Internet im Fahrzeug nutzen zu können ist nicht nur für den Bereich des Entertainments interessant, sondern bietet dem Fahrer auch die Möglichkeit, während der Fahrt produktiv zu sein; sogenannte *In-Car Productivity*. Auch hier nutzen etablierte Unternehmen die Möglichkeit ihren Markt mit neuen Applikationen im Automobilbereich zu erweitern. So integriert beispielsweise der Automobilhersteller Volvo als erstes die Applikation *Skype for Business*, welche bereits in vielen Firmen genutzt wird, um

Gruppenmeetings online abhalten zu können. Mit Hilfe der Integration dieser Software in ein Fahrzeug ist es möglich, an eben solchen Meetings auch während einer Fahrt teilzunehmen und somit auch die Fahrzeit produktiv zu nutzen (Volvo, 2020). Neben Volvo arbeitet auch Daimler mit Microsoft zusammen. Diese Zusammenarbeit findet im Rahmen eines *In Car Office*-Projektes statt, das den Fokus vor allem auf die Integration von Microsoft Exchange legt, um Informationen wie E-Mails, Kontakte oder Termine im Fahrzeug verfügbar zu machen (Jelica, 2016). Neben *Skype for Business* oder *In Car Office* bietet Microsoft mit Windows Embedded Automotive ein Betriebssystem für Fahrzeuge an. Auf dessen Grundlage kann verschiedene Funktionalität entwickelt werden, die Kunden auch nach dem Erwerb eines Fahrzeugs über Updates in ihrem Fahrzeug nutzen können, was wiederum deren Loyalität steigern kann. Dabei funktioniert *Windows Embedded Automotive* plattformunabhängig (Microsoft, 2015). *Windows Embedded Automotive* lässt sich auf Grund seines Funktionsumfangs, der von *Vehicle Telematics* über *In-vehicle Infotainment* reicht, auch als Beispiel für den Begriff der *Location Based Information* aufführen. Unter diesem wird die Möglichkeit verstanden, mit Hilfe der Bestimmung des Standorts eines Fahrzeugs in Kombination mit der Verbindung zum Internet standortspezifische Informationen zu erhalten. So kann der Fahrer beispielsweise über das Internet suchen, welche Tankstellen oder Restaurants sich in der Nähe befinden. Neben dieser Funktion, die mitunter auch das Navigationssystem ausführen kann, lassen sich weitere Informationen über die Internetverbindung recherchieren, die über die eines Navigationssystems hinausgehen. Ein weiterer Bereich ist das *Vehicle Platooning*. Dabei werden Verkehrskolonnen auf Basis (teil-)autonomer Fahrzeuge gebildet. Dennoch kann die Anbindung der Fahrzeuge an das Internet zu dieser Nutzungsform beitragen, da die Fahrzeugkommunikation sowohl mit anderen Fahrzeugen als auch mit der Infrastruktur über diese Anbindung ermöglicht werden kann. Die weitere Verbreitung von internetfähigen Fahrzeugen kann somit genutzt werden, um zukünftig Lösungen für das *Vehicle Platooning* zu entwickeln. Die Anbindung von Fahrzeugen an das Internet bietet auch aus Geschäftsperspektive verschiedene Potentiale, die teilweise bereits genutzt werden. So wird es beispielsweise ermöglicht, Standortdaten von Fahrzeugen auf einen Server zu übertragen, die notwendig sind, damit Nutzer von CarSharing-Services ein Fahrzeug finden können. Auch für den Bereich der *Remote Diagnostics* ist die Vernetzung mit dem Internet ein Fortschritt, da gemessene Fehler am Fahrzeug nicht nur mit Hilfe einer Signalleuchte optisch an den Fahrer übermittelt, sondern die Messdaten auch direkt mit einer Werkstatt oder dem ADAC geteilt werden können.

Augmented Car

Unter Augmented Reality (AR) wird „die Anreicherung der Realität durch künstliche virtuelle Objekte“ (Dörner, Broll, Grimm & Jung, 2014) subsumiert. Diese wird als eine „logische Weiterentwicklung der virtuellen Realität“ (Tönnis, 2010) verstanden. AR in einer natürlichen Umgebung durch das Einbetten von neuen Gegenständen im Raum oder die erweiterte Informationsdarstellung zu existierenden Objekten (Tönnis, 2010). Dabei wird die Erweiterung mithilfe virtueller Objekte „nicht statisch und einmalig, sondern kontinuierlich an den aktuellen Standpunkt des jeweiligen Betrachters angepasst“ (Dörner et al., 2014), was durch die Umsetzung in Echtzeit möglich ist.

Aus technologischer Betrachtung wird zwischen Black & White Markers und dem Depth-Sensing (Markerless) unterschieden. Beide Technologien unterscheiden sich dabei in ihrer Fähigkeit, Objekte aus der realen Welt zu erkennen und als solche zu identifizieren. Während letzteres in der Lage ist, jegliche Objekte anhand von Ortungsfunktionen durch GPS oder andere Technologien wahrzunehmen und anschließend virtuellen Inhalt darauf präzise abzubilden, ist es für Applikationen, die die Variante mit Markern verwenden, nur möglich, vorher definierte Objekte zu identifizieren (Kamphuis, Barsom, Schijven & Christoph, 2014). Zum Interagieren mit den virtuellen Objekten durch AR-Technologie kommen zwei unterschiedliche Methoden zum Einsatz. Das Motion Tracking realisiert die Gestensteuerung mit FingerIO oder Soli (Brien, 2019). Letzteres ist eine Technologie von Google, die ein Radarmodul im Mikrochip benutzt und Objekte mithilfe der Kamera identifiziert, um die virtuellen Inhalte über das Übertragungsmodul zu steuern. Hierfür werden sich verändernde Wellen registriert, die bei entsprechend hoher Frequenz sehr präzise in der Lage sind, Gesten wahrzunehmen. Sonar ist dagegen eine Technologie, die die Gesten des Nutzers durch Mikrofone und Lautsprecher identifiziert, um eine Steuerung (FingerIO) des Gerätes zu ermöglichen. Zudem werden AR-Hardware und verschiedene Ansätze zur Wahrnehmung der Umgebung (*Object Recognition*) genutzt, die von der Aufnahme der relevanten Daten (Ingest) bis zu seiner Nutzung (Utilize) Anwendung finden. Bekannte Hersteller wie Google (Google Glass), Sony (SmartEyeglass) und Microsoft (Hololens) bieten ihre AR-Geräte zum Verkauf an und ermöglichen durch *Head Mounted Displays* – Geräte, die wie eine Brille über dem Kopf getragen werden, sodass der Nutzer kein weiteres Gerät in den Händen halten muss und die Sicht mit der Bewegung des Kopfes immer stabil bleibt – neben den herkömmlichen Nutzgeräten, wie Smartphones oder Tablets, einen angenehmen Weg, die erweiterte Realität zu erleben (Kamphuis et al., 2014). Um nun diese Darstellungen über die eben beschriebenen Endgeräte wahrzunehmen, benötigt es einer physischen Identifizierung der Objekte in der Realität

(*Physical Cognition*). Hierfür gibt es unterschiedliche Möglichkeiten, die je nach Nutzungsszenario implementiert werden. Das *Object Detection* scannt physische Objekte und identifiziert sie als reale Objekte und kann sie für die erweiterte Realität verwenden, indem virtuelle Inhalte darauf abgebildet werden. Beim 3D-Scan wird entweder das Objekt oder der Raum gescannt, um es als virtuellen Inhalt darzustellen. Das *Optical Character Recognition (OCR)* wird implementiert, um Zeichen wie Zahlen und Buchstaben aus der physischen Umgebung zu erfassen.

Jede Art der AR-Nutzung basiert auf dieser virtualisierten Darstellung, dennoch gibt es in ihrer Umsetzung einige Unterschiede. Während über Smartphones oder andere AR-Geräte virtuelle Objekte im Raum dargestellt werden, können mit den *Geolocation-based* AR-Diensten Szenarien umgesetzt werden, die weitreichendere Funktionen bieten. Es existieren beispielweise Anwendungen wie AED4EU, die den Nutzer per Smartphone oder Assistenzsystem und virtuellen Navigationsobjekten zum nächsten öffentlich angebrachten Defibrillator bringen. Die *Touchless Gesture Interaction* bietet eine neue Dimension der Steuerung von Endgeräten über die auf Technology-Ebene beschriebenen Möglichkeiten mittels Soli und FingerIO (Brien, 2019). Durch das *Motion Tracking* werden die Gesten des Anwenders erfasst und als Steuerungsmaßnahmen analysiert, sodass es möglich ist, Dinge ohne sie zu berühren, zu steuern. In der Praxis werden diese Techniken bereits umfassend genutzt. Auch die Automobilbranche bedient sich dieser Technologie und macht sich diese im Fahrzeuginnenraum zu Nutze und bietet seinen Kunden eine noch einfachere Bedienung der Tafelemente im Auto. Der Fahrer kann dabei durch Wisch- und Zeigegesten Funktionen wie, Telefonannahme oder Liederweitergabe steuern (BMW China, 2015).

3.4.3 Automotive

Human Machine Interface (HMI)-Lösungen im Auto

Die Schnittstelle zwischen Mensch und Maschine - im Falle der Automotive-Industrie des Fahrers und seines Fahrzeugs - wird als Human Machine Interface (Buxmann & Schmidt, 2019) bezeichnet. Diese Schnittstellen ermöglichen auf der einen Seite die Eingabe von Parametern durch den Fahrer und auf der anderen Seite die Ausgabe von Informationen für den Fahrer. Zur Ausgabe zählen sowohl visuelle Hinweise wie die Geschwindigkeit und Drehzahl des Fahrzeugs, gefahrene Kilometer oder die Tankfüllung als auch Warnungen für den Fahrer wie das Aufleuchten verschiedener Kontrollleuchten bei Problemen technischer Komponenten des Fahrzeugs. Als visuelles HMI sind *Heads-Down Displays* weit verbreitet. Ein Nachteil dieser Ausgabeschnittstellen liegt darin, dass der Fahrer seinen Blick von der Fahrbahn nach unten in den Bereich hinter seinem Lenkrad oder auf die Mittelkonsole seines Fahrzeuges abwenden muss,

um die genannten Informationen und Warnungen erfassen zu können (Meyer & Valldorf, 2011). Diese Ablenkung lässt sich durch die Nutzung von Heads-Up Displays (HUD) verringern, da sich diese Displays im direkten Sichtfeld des Fahrers befinden. Hierzu existieren verschiedene Konzepte von diversen Herstellern. So bietet beispielsweise die Firma Continental zwei unterschiedliche Bauweisen an. Zum einen das Windschutzscheiben-HUD, dessen Anzeige so konzipiert ist, dass der Fahrer diese in zwei bis drei Metern Entfernung direkt vor der Motorhaube wahrnimmt und zum anderen das Augmented Reality HUD. Windschutzscheiben-HUDs werden bereits von namenhaften, deutschen Automobilkonzernen, wie Audi, BMW, Mercedes und Volkswagen verwendet. Technisch basieren diese auf einer Projektion auf die Frontscheibe des Fahrzeugs. Im Detail nutzt beispielsweise Audi einen LED beleuchteten Flüssigkristallbildschirm, der ein farbiges Bild generiert. Zwei asphärische Spiegel vergrößern dieses Bild, lenken es auf die Scheibe um und gleichen dabei die Verzerrungen dieser aus (Audi AG, 2020). Das Verkaufsargument für den Einbau oder die Nachrüstung dieser Displays stützt sich primär auf die gesteigerte Sicherheit, da das Sichtfeld zum Erfassen der relevanten Information nicht mehr verlassen werden muss. Die Anzeige von Augmented Reality HUDs erweitert das Windschutzscheiben-HUD durch eine zusätzliche Ebene. Diese stellt Hinweise so dar, als wären sie ein Bestandteil der aktuellen Fahrsituation - wie beispielsweise Pfeile auf der Fahrbahn, die den Fahrer darauf aufmerksam machen, wenn dieser abbiegen muss (Continental Automotive GmbH, 2014).

Neben HUDs existieren weitere Möglichkeiten und digitale Neuerungen im Bereich HMI, die die Sicherheit beim Fahren erhöhen. Hinter dem Begriff *Advanced Driver Assistance Systems* (ADAS) verbergen sich verschiedene technische Komponenten, die den Fahrer bei der Steuerung des Fahrzeugs unterstützen. Im Gegensatz zu HUDs erstrecken sich ADAS über alle horizontalen Betrachtungsebenen. Um das Handeln des Fahrers unterstützen zu können, werden Kameras und andere Sensoren genutzt, welche Daten aufnehmen. Diese wiederum werden über entsprechende Applikationen und Algorithmen so analysiert und aufbereitet, dass aus den einzelnen Komponenten und Prozessschritten im Ergebnis ein Hinweis oder sogar ein konkretes Eingreifen resultieren kann, das den Fahrer unterstützt. Ein konkreter Anwendungsfall aus dem Bereich ADAS ist die Verkehrszeichenerkennung (engl. *Traffic Sign Recognition* – TSR). Mit Hilfe einer oder mehrerer Kameras werden Verkehrszeichen aufgenommen. Eine Software verarbeitet die erfassten Daten zu logischen Informationen, die den Fahrer beispielsweise auf die zulässige, maximale Höchstgeschwindigkeit hinweisen. Die von der oder den Kameras aufgenommenen Daten können jedoch zu weit mehr als nur der Erkennung von Verkehrszeichen genutzt werden. Weitere ADAS werden auf der

Grundlage zusätzlicher Sensortypen, wie Infrarotkameras, Kurz-/Mittel- und Fernbereichsradare sowie Ultraschall ermöglicht. Dazu zählen adaptive Tempomaten, Nacht-sicht-, Stau-, Park-, Totwinkel- und Spurwechselassistenten (Siebenpfeiffer, 2015). Die verschiedenen sicherheitsrelevanten Assistenzsysteme tragen durch ihren unterstützenden Charakter zur Unfallvermeidung (*Accident Prevention*) dar. Dabei werden die von verschiedenen Sensoren erfassten Daten so verarbeitet, dass das Fahrzeug in Not-situationen direkt in die Steuerung eingreift, beispielsweise durch eine Gefahrenbremsung. Eine passivere Schnittstelle, die jedoch auch zum Ziel hat Unfälle zu vermeiden, ist der so genannte *Artificial Passenger*. Dieser basiert auf dem „Conversational Inter-activity for Telematics“ (CIT) Projekt der IBM. Hier soll der Fahrer beispielsweise durch Konversation oder interaktive Spiele am Einschlafen während der Fahrt gehindert werden. Dafür werden gleichzeitig verschiedene Parameter in der Sprache gemessen, um den Grad an Fokussierung des Fahrers bestimmen zu können. Der CIT beinhaltet wiederum ein weiteres System – den Dialog Manager, der die nötigen Informationen für eine Unterhaltung aus verschiedenen Quellen bezieht, wie zum Beispiel dem Wetter oder Ortsinformationen (Gardner-Bonneau & Blanchard, 2007). Während sowohl HUDs als auch die beschriebenen Assistenzsysteme Schnittstellen zur Ausgabe darstellen, werden in Fahrzeugen auch verschiedene Eingabeschnittstellen benötigt und kontinuierlich weiterentwickelt.

Vehicular Automation

Es existieren viele unterstützende Systeme, die zur Unabhängigkeit eines Fahrzeugs von dessen Fahrer beitragen. Diesen Assistenzsystemen oder -schnittstellen liegen auf technologischer Ebene verschiedene Komponenten zu Grunde. Zwei der wesentlichen sind LIDAR und Kameras für *Computer Vision*. Light Detection and Ranging (LIDAR) sind „[...] Laser-gestützte Geräte, die in Echtzeit und im breiten Umfeld Gegenstände und ihre Entfernung zum Fahrzeug erkennen können“ (Johanning & Mildner, 2015, S. 64). Computer Vision, im deutschen auch als „maschinelles Sehen“ bezeichnet, beschreibt das technologische Zusammenspiel von Kameras und Algorithmen zur Verarbeitung der aufgenommenen Bilder. Dafür werden Techniken zur Vervollständigung unklarer Bilder (*Image Restoration*), zum Erfassen sich bewegender Objekte (*Video Tracking*) und der Abschätzung von Bewegungsvektoren (*Motion Tracking*) eingesetzt. Dabei stellen vor allem die Algorithmen einen erheblichen Kostenfaktor bei der Entwicklung autonomer Fahrzeuge dar. Tabelle 3-1 gibt eine Übersicht über die verschiedenen Aufgaben von Algorithmen im Bereich der Fahrzeugautomatisierung.

Tabelle 3-1: Anwendung von Algorithmen für autonome Fahrzeuge (eigene Darstellung auf Basis von Johanning & Mildner (2015, S. 64, 66)).

<p>Signalkonversion (Abbildung der Außenwelt auf die interne Darstellung)</p>	<ul style="list-style-type: none"> • Bildverarbeitung • LIDAR-Verarbeitung • Radar-Musterverarbeitung
<p>Signalinterpretation</p>	<ul style="list-style-type: none"> • Objekterkennung und -klassifizierung • Erkennung von Hindernissen • Erkennung der Umgebungsgeometrie und Straßenmarkierungen • Dynamische Modellierung • Erkennung von Verkehrszeichen und Ampeln • Räumliche Fahrzeugpositionierung • Fahrzeugführung
<p>Entscheidungsfindung</p>	<ul style="list-style-type: none"> • anhand der aktuellen Datenlage

Die räumliche Fahrzeugpositionierung erfolgt beispielsweise „durch Messung der Eigenbewegung eines mobilen Roboters über Radsensoren und vektorielle Addition von Wegstrecken“ (Hesse & Schnell, 2014). Von diesem und den anderen in der Tabelle dargestellten beispielhaften Anwendungsgebieten erweisen sich jene zur Entscheidungsfindung als besonders komplexe und herausfordernde Berechnungen. Bereits die Verarbeitung der Eingangsdaten zur Ermittlung von Entscheidungen während einer Autobahnfahrt ist sehr umfangreich. Die Berechnungen, die während einer Fahrt im Stadtverkehr entstehen, sind wesentlich komplexer. Zusätzliche Verkehrsteilnehmer, wie Schienenfahrzeuge oder Fußgänger sind weitere Objekte, die erkannt werden müssen (Johanning & Mildner, 2015). Wenngleich der Aufwand zur vollständigen Automatisierung von Fahrzeugen sowohl hohe technische als auch prozessuale und logische Anforderungen aufwirft, ergeben sich daraus auch viele neue Möglichkeiten der Nutzung. Ein wichtiger und häufig verwendeter Begriff im Zusammenhang von autonomen

Fahrzeugen und deren Eigenschaften ist *Vehicle Platooning*. Eine Erhöhung der Verkehrssicherheit, Einsparpotentiale von Treibstoff oder auch größere Aufnahmefähigkeiten von Autobahnen lassen sich mit dessen Hilfe erzielen. Dazu werden autonome Fahrzeuge in so genannten *Platoons* (Züge oder Kolonnen) verknüpft. Dabei werden Parameter, wie der Abstand der einzelnen Fahrzeuge computerbasiert gesteuert. Dies ermöglicht es, den sonst erforderlichen Sicherheitsabstand beim Führen eines Fahrzeugs durch einen Menschen auf ein Minimum zu reduzieren, da die Reaktionszeiten eines Computers wesentlich kürzer sind als die eines Menschen. Neben den bereits aufgeführten Vorteilen unterstützt dieses Konzept den Umgang mit weiteren Herausforderungen wie bspw. Unfälle oder Staus sowie Emission und Energieverbrauch (Kavatthekar & Chen, 2011).

Vehicle Telematics

Vehicle Telematics wird hauptsächlich dafür eingesetzt, so genannte *Intelligent Transportation Systems (ITS)* zu konzipieren und zu realisieren. Grundsätzlich verstehen sich darunter Systeme zur Verbesserung der allgemeinen Verkehrsinfrastruktur in den Bereichen Sicherheit, Effektivität und Effizienz, Beeinträchtigungen der Umwelt. Um auf die Lenkung des Verkehrs Einfluss nehmen zu können, damit beispielsweise die Staugefahr gemindert und somit den Verkehr effizienter und umweltfreundlicher zu gestalten, müssen Verkehrsteilnehmer mit Hilfe der Infrastruktur erkannt werden können. Eine Möglichkeit dazu bietet die so genannte *Inductive Loop Detection*. Mit Hilfe elektrischer Spulen, die entweder fest in einer Straße verbaut oder temporär über einer Straße installiert werden, lassen sich Fahrzeuge erkennen, die über die Spule oder unter der Spule durchfahren. Zur Messung wird hier die Änderung des Magnetfeldes der Spule genutzt. Dieses Erkennungsverfahren ist sehr genau. Eine neuere Methode – *Video Vehicle Detection* – beinhaltet die Verkehrsbeobachtung durch Videokamerasysteme. Der Vorteil an diesem Verfahren liegt darin, dass mit Hilfe von zusätzlicher Software mehr Information als nur die Anzahl von Fahrzeugen generiert werden können, wie beispielsweise Fahrzeugklassen und Geschwindigkeiten der Fahrzeuge. Die Genauigkeit der Informationen wird dabei durch folgende Parameter beeinflusst: Videoqualität der verwendeten Kameras, verwendete Analysesoftware sowie die allgemeinen Lichtverhältnisse (Mukhopadhyay & Leung, 2010). Ein wichtiger Aspekt beim Auf- und Ausbau von *ITS* ist die Reisedauer von Fahrzeugen. Diese wird zum Teil mit den bereits beschriebenen *Inductive Loops* und *Video Vehicle Detection* erfasst, was mit hohen Installations- und Wartungskosten verbunden ist und durch äußere Parameter qualitativ beeinträchtigt werden kann. Als eine kostengünstigere und qualitativ genauere Methode kann alternativ *Bluetooth Detection* eingesetzt werden. Die Erfassung von Reisezeiten der Fahrzeuge wird dabei so realisiert, dass über ein Bluetooth Protokoll

die eindeutige MAC-Adresse eines Fahrzeugs an verschiedenen Streckenpunkten kommuniziert werden kann und sich somit die Zeit zwischen zwei Punkten berechnen lässt (Araghi, Pedersen, Christensen, Krishnan & Lahrmann, 2015). Die technischen Möglichkeiten Fahrzeuge und verschiedene dazugehörige Daten erfassen zu können, bildet die Grundlage für verschiedene Applikationen im Bereich der *ITS*. Im Gegensatz zu konventionellen Mautstationen, an denen Fahrer die Gebühren manuell zahlen, sollen mit Hilfe von elektronischen und automatisierten Verfahren Usability und Komfort bei der Bezahlung von Mautgebühren erhöht werden. Weiterhin kann das Stocken des Verkehrs an diesen Stationen und damit einhergehende erhöhte Emission vermieden werden (Pickford & Blythe, 2006). Die Nutzungsmöglichkeiten von *Vehicle Telematics* im Automotive Bereich sind vielfältig. Eine Lösung der Firma Continental ist die Software TIS-Web®, welche das Management eines Fuhrparks unterstützt. Mit Hilfe von TIS-Web® werden die Fahrzeugdaten der Tachographen einer Flotte online gespeichert und ausgewertet. Neben der Sicherstellung dieser gesetzlichen Rahmenbedingungen ermöglicht die Software dem Nutzer eine effiziente Organisation der Fahrzeugflotte, in dem sie weitere Funktionen wie die Personal- und Einsatzplanung sowie Wartungspläne umfasst (VDO, 2020).

3.4.4 Logistik und Transport

Durch die Unterstützung von Produktionsbetrieben mit KI-Lösungen können sich zuverlässigere Bedarfsprognosen, flexiblere und reaktionsschnellere Lieferketten, schnellere Betriebsänderungen, genauere Planungen sowie Bestandsoptimierungen einstellen. Weitere Vorteile sind die Schaffung intelligenterer, schnellerer und umweltfreundlicherer Prozesse. Auch können diese Verbesserungen die Produktivität und Qualität erhöhen, Kosten senken sowie die Gesundheit und Sicherheit des Unternehmens erhöhen.

In der *Logistic & Transportation*-Branche gibt es lediglich den verzahnten Baustein Fleet Management/Public Transportation. Dieser beschreibt Technologien, die dafür genutzt werden, um sowohl öffentliche Transportmittel wie Busse oder Transportfahrzeuge wie LKWs besser zu managen. Die hier zum Tragen kommenden Fleet Telematics Systeme, Mobile Vehicle Systems und Fleet Communication Systems (FCS) dienen dazu die Kommunikation zwischen dem Fahrzeug und der Zentrale zu realisieren. Im Rahmen dieser Kommunikation werden Informationen, wie die aktuelle Position des Fahrzeugs und Sensordaten aus dem Laderaum weitergeleitet, um eine Echtzeitkontrolle zu gewährleisten. In diesem Zusammenhang beschreibt das Fleet Management Tracking die genaue Ortung des Fahrzeugs zur besseren Planung und Überwachung der eingesetzten Transportmittel.

Predictive Logistics

Ein Schlüsselbereich der KI-Anwendung in der logistischen Abwicklung ist die adaptive Planung von Lieferungen und die Routenplanung der Fahrzeuge. Fortschrittliche Logistik- und Lieferketten werden mit Hilfe von Expertensystemen zur Entscheidungsunterstützung erstellt. Bestellungen können durch optische Fahrerassistenz- und automatisierte / robotische Systeme effizienter transportiert werden. Dies hat den Transport weniger anfällig für Störungen durch Wetter, Verkehr und unnatürliche Ereignisse gemacht. Einen Überblick zum Einsatz von ML in der Logistik ist in Abbildung 3-3 zu finden.

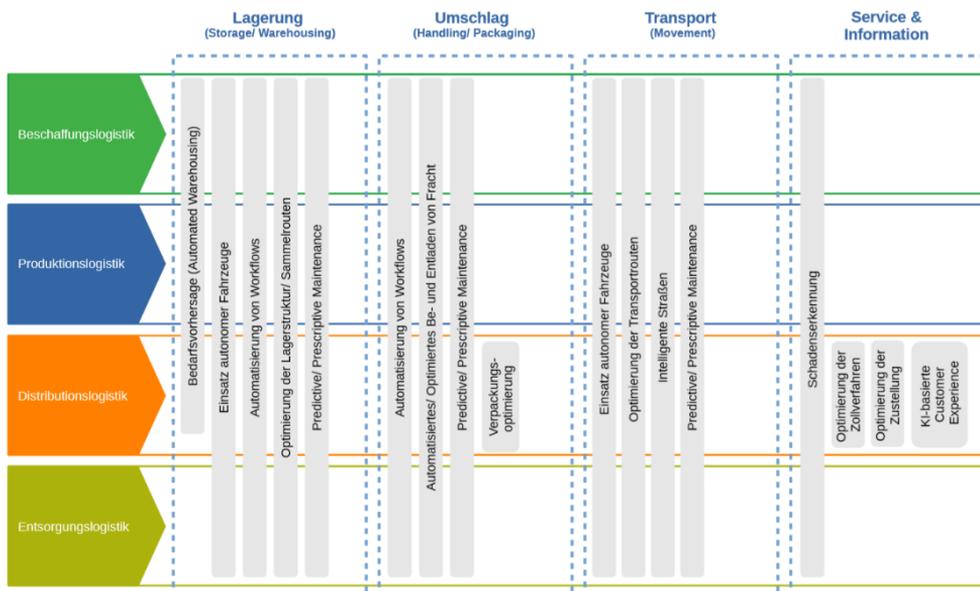


Abbildung 3-3: Übersicht ML in der Logistik (Eigendarstellung).

Klassisches Tracking von Waren ermöglicht die Lokalisierung des aktuellen Aufenthaltsorts. Für eine gut funktionierende Lieferkette ist es jedoch auch hilfreich zu erfahren, wann Waren am Zielort ankommen werden. Hierfür wird auf Sensoren und das Internet der Dinge sowie deren Kombination mit maschinellen Lernverfahren gesetzt, sodass *Predictive Logistics* ermöglicht wird. Predictive Logistics ermöglicht die Vorhersage von Ankunftszeiten auf Basis von Liefer-, Entlade- sowie Liegezeiten, indem die großen entstehenden Datenmengen mittels ML analysiert werden.

Car Platooning

In der Logistikbranche können Flotten von Lastkraftwagen als Verbund mittels Car Platooning möglichst effizient Waren transportieren. Aus diesem Verwendungszweck heraus ergeben sich weitere technische Anforderungen, wie beispielsweise die dynamische Anordnung innerhalb einer Fahrzeugkolonne. Diese sollte zum einen möglichst effizient sein und zum anderen Sicherheitsaspekte berücksichtigen. Tabelle 3-2 zeigt verschiedene Platooning Projekte und verdeutlicht im Vergleich die Heterogenität des Anwendungsbereichs.

Die Effizienz lässt sich beispielsweise anhand unterschiedlicher Fahrzeuggrößen steigern, indem kleine Fahrzeuge weiter hinten in der Kolonne angeordnet werden, um den Windschatten vorausfahrender größerer Fahrzeuge auszunutzen und entsprechend Einsparungen beim Treibstoff zu erzielen. Gleichzeitig müssen sicherheitsrelevante Parameter ebenfalls in die Berechnung der optimalen Anordnung einfließen, wie das Fahrzeuggewicht und der technische Zustand der Bremsen eines Fahrzeugs zur Berechnung des jeweiligen Bremswegs. Weitere zukünftige Herausforderungen sind Aspekte wie die Verknüpfung von Fahrzeugen verschiedener Hersteller, die Möglichkeit für Fahrzeuge anderer Logistikunternehmen sich im Laufe einer Route einem Platoon anzuschließen und dazu im Vorfeld mit diesem in Kontakt zu treten, um es finden zu können (Meyer & Beiker, 2015).

Tabelle 3-2: Vergleich verschiedener „Platooning“-Projekte (eigene Darstellung auf Basis von Bergenhem, Shladover, Coelingh, Englund und Tsugawa (2012, S. 6) und Dadrás, Gerdes und Sharma (2015)).

Projekt-namen	Fahr-zeugtyp	Kontrolle	Anforderun-gen einer Inf-rastruktur	Integra-tion des Verkehrs	Sensoren	Ziele
SARTE	Gemischt	Längen- und Breiten-grad	Keine	Autobahn, gemischt	Produktion	Komfort, Sicherheit, Stau, Energie
PATH	Schwer	Längen- und Breiten-grad	Referenzmar-kierungen auf der Straße	Ausgewie-sene Fahr-bahnen	Gemischt	Erhöhter Durchsatz per Bahn, Energieein-sparung
GCDC	Gemischt	Längengrad	Augmentiertes GPS	Gemischt	SoA und Pro-duktion	Beschleunigte Ent-wicklung kooperativer Fahrzeugsysteme
Energy-ITS	Schwer	Längen- und Breiten-grad	Fahrbahnerken-nung	Ausgewie-sene Fahr-bahnen	SoA	Verminderung von mangelnden Fahr-competenzen
SCANIA	Schwer	Längengrad	Keine	Autobahn, gemischt	Keine, Auto-zu-Auto-Kommuni-kation in Stufe 1	Kommerzielles Flot-ten- und Energiema-nagement

Infrastrukturmanagement

KI kann die Effizienz des Betriebs im Reise- und Transportsektor verbessern, indem es durch Systeme zur Strukturüberwachung und zum Infrastrukturdatenmanagement für mehr Sicherheit sorgt. Dies kann sich in Form von reduzierten Reparatur- und Rekonstruktionskosten oder Echtzeitinformationen in der Routenberechnung auszahlen, wodurch die Transparenz steigt und der Energieverbrauch und Emissionen reduziert werden.

3.5 Alltag und Entertainment

Während eine KI zunächst im wirtschaftlichen Bereich, im Unternehmenskontext und in technisch hochversierten Systemen eingesetzt wurde, finden KI-Ansätze auch in alltäglichen Bereichen und im Entertainment-Bereich immer mehr Einzug. So finden KI-Ansätze sich inzwischen auch in akkubetriebenen Staubsaugern, welche anhand der KI-basierten Erkennung der Bodenbeschaffenheit die Laufleistung verlängern, Rasier-

apparate, welche anhand der KI-basierten Erkennung die Bartbeschaffenheit identifizieren und die Rasierleistung verbessern, oder in der Navigation von Saug- und Mährobotern. Weitere Beispielbereiche finden sich nachfolgend zusammengefasst.

3.5.1 Alltag

Intelligente Haarbürste

Um Tipps für gesünderes Haar bieten zu können, wird mit einer intelligenten Haarbürste das Kämmverhalten von Nutzern aufgezeichnet. Mit Hilfe verschiedener Sensoren wie einem Gyroskop, Mikrofon oder Leifähigkeitssensor werden Charakteristiken bestimmt und ausgewertet, sodass die Pflegegewohnheit verbessert und helfende Produkte empfohlen werden können (L'Oréal, 2017).

Augmented Reality (AR) Emojis

Normale Smartphone-Emojis sind statisch oder haben eine fest definierte Animation. AR-Emojis werden erstellt, indem der Gesichtsausdruck des Nutzers in ein animiertes 3D-Modell gespiegelt wird (Palladino, 2018). KI-Techniken helfen hier den Gesichtsausdruck des Nutzers zu erkennen und in das digitale Abbild des Nutzers zu überführen. Samsung vermarktet diese Technologie als AR-Emojis (Samsung, 2019), während Apple diese als *Memojis* bezeichnet (Apple, 2019).

3.5.2 Gaming

Im Bereich der Videospiele wird KI in vielen verschiedenen Bereichen eingesetzt. Darunter das Steuern von nicht-spielbaren Charakteren (NPCs), die Generierung von prozeduralen Spielwelten bis hin zu automatischen Anti-Cheat-Systemen.

Anti-Cheat-Systeme

Das Ziel eines Anti-Cheat-Systems ist es, mitlaufende Cheat-Software aufzudecken und den Spieler für sein unfaires Verhalten entsprechend zu bestrafen. Ein Beispiel für die Verwendung von Deep Learning in Bezug auf Anti-Cheat-Systeme kann in Valves-Anti-Cheat-System (VAC) gefunden werden (Gieselmann, 2018). Bislang setzte Valve in Spielen wie etwa Counter Strike auf Peer-Beobachtung. Mittels einer Aufzeichnung des gemeldeten Spielers konnte der Spieler durch andere Spieler überführt werden. Diese über die Jahre gesammelte Aufzeichnung konstituiert durch die Peer-Beobachtung letztlich ideal gelabelte Trainingsdaten. Mittels dieser wurde ein neuronales Netz trainiert, welches zukünftig bei der Klassifizierung unterstützt.

Verbesserte grafische Darstellung

Jeder Videospiele-Fan kennt dieses Problem: Aus einiger Distanz sehen den eigenen Charakter umgebene Gebäude oder Wälder optisch ansprechend aus. Begibt sich der Spieler jedoch näher an diese, so leidet sehr schnell die Immersion. Unschärfe Texturen, sowie eine zu geringe Anzahl an Polygonen bestätigen die Tatsache, dass es eben doch nur ein Videospiele ist. Eine Lösung, insbesondere ältere Videospiele zu verschönern, kann in KI-Upscaling gefunden werden. Dabei wird ein Eingangsbild mit einer niedrigen Auflösung mittels ML-Algorithmus in eine höhere Auflösung konvertiert. Details, welche zuvor nicht oder nur sehr grob sichtbar waren, können somit besser zum Vorschein kommen. Deep-Learning-Ansätze können dabei helfen, die beste Farbe für ein Pixel zu finden, um skalierte oder gezoomte Bilder zu verbessern (Hoef, 2018).

3.5.3 E-Commerce

Die Einsatzmöglichkeiten von maschinellem Lernen im E-Commerce sind vielfältig, da gerade in diesem Bereich eine große Menge an Daten anfällt, die genutzt werden kann, um den Kunden ein individualisiertes Käuferlebnis zu ermöglichen. Dabei sind die potenziellen Einsatzgebiete bereits heutzutage wesentliche Erfolgsfaktoren von Onlineshops (Holtforth, 2018): Empfehlungen von Produkten sowie Nutzung von Cross- und Upselling-Potenzialen, indem Nutzungsverhalten, Suchverlauf und ggf. bereits vorhandene Kundenstammdaten kombiniert und analysiert werden. Weiterhin können Werbeanzeigen durch die Analyse der Daten personalisiert werden, sodass diese für den Kunden einen höheren Relevanzgrad aufweisen. Weitere Anwendungsbereiche im E-Commerce sind dynamische Preisfestsetzung, Betrugsvermeidung durch Erkennung von Mustern bei Bestellungen, Predictive Analytics zur Einschätzung des Wettbewerbs als auch Nachfrageveränderungen, Vorhersage von Lieferzeiten oder der Einsatz von Chatbots bei der Beantwortung von Routinefragen.

Stellvertretend ist Syte (2020) ein Beispiel für den Einsatz von KI zur visuellen Unterstützung der Kundenerfahrung auf der Reise bis zum Kauf eines Produkts. Dabei werden Empfehlungen auf Basis der Suchnavigation in einem Onlineshop.

3.6 Produktfertigung

Koordinierte und traditionelle Tätigkeiten der Produktfertigung bieten ein breites Anwendungsfeld für den Einsatz von ML-Ansätzen. Es zeigen nachfolgende Beispiele einen Auszug von teilweise überraschenden Anwendungsbereichen, die durch intelligente Verfahren verbessert werden können.

3.6.1 Betrieb

Kartierung der globalen Fischereitatigkeiten

Die Fischbestande einiger Arten sind bereits um 90% gesunken. Da aktuell die Ernahrung von uber einer Milliarde Menschen auch von gefangenen wilden Fischen abhangig ist, ist es essenziell, illegales Fischen zu unterbinden und Regelungen fur eine nachhaltige Fischerei aufzustellen. Da jedes Schiff seine Positionsdaten an das „Automatic Identification System“ schickt, konnen mit Hilfe von maschinellem Lernen der Schiffstyp, das Fanggerat sowie Bewegungsmuster identifiziert werden (Sullivan, 2016).

Verbesserte Reinigung in der Lebensmittelindustrie

Im Bereich der Lebensmittelindustrie werden viele Gerate, Werkzeuge und Gefae zur Verarbeitung und Aufbewahrung von Lebensmitteln genutzt und mussen gereinigt werden. Dabei konnte es sich zum Beispiel um Besteck oder auch leere Flaschen handeln. Klassischerweise landet alles in Geschirrspulern oder ahnlichen Geraten. Mit dem Ziel, dass nach der Reinigung samtliche Verschmutzungen beseitigt sind, laufen die Gerate auf hoher Stufe, um auch starke Verschmutzungen effektiv zu reinigen. Dabei wird deutlich mehr Zeit, Energie und Wasser benotigt als es fur manche Verschmutzungen notwendig ware. Stattdessen man die Reinigungsgerate mit Sensoren aus, welche den Grad der Verschmutzung scannen, lasst sich die Reinigungszeit laut Forschern der Universitat Nottingham um bis zu 50% reduzieren und somit die gesamte Produktivitat erhohen (Walker, 2019).

Landwirtschaft

Die Landwirtschaft ist ein weiterer Sektor, der von intelligenten Losungen profitieren kann, indem er intelligentere Produktions-, Verarbeitungs-, Lager-, Verteilungs- und Verbrauchsmechanismen nutzt. Des Weiteren konnen KI-Losungen, durch standortspezifische und aktuelle Daten uber Pflanzen, dazu beitragen, dass die Anwendung geeigneter Substanzen wie Dungemittel und Chemikalien optimiert wird (PwC, 2017).

Konsumguter und Dienstleistungen

Konsumguter und Dienstleistungen waren einer der ersten Bereiche der Einfuhrung von KI in Indien und machen derzeit einen erheblichen Teil der Anwendung im privaten Sektor aus. Um es den Verbrauchern zu ermoglichen, bessere Produkte zu niedrigen Preisen zu finden, werden Algorithmen fur maschinelles Lernen eingesetzt, um das Angebot besser an die Nachfrage der Verbraucher anzupassen.

Online-Shopping-Portale nutzen in groem Umfang Prognosefunktionen, um das Interesse der Verbraucher an Produkten zu messen, indem sie ein gezieltes Verstandnis

der Präferenzen durch die Erfassung von Browsing- und Click-Stream-Daten aufbauen und Kunden über einen mehrkanaligen Ansatz effektiv ansprechen und binden.

3.6.2 Wartung

Vorausschauende Wartung (Predictive Maintenance)

Um Ausfällen vorzubeugen werden Produktionsmaschinen in regelmäßigen Abständen gewartet. Dabei gilt es, die Personalkosten für die Wartung sowie gegebenenfalls die Kosten für den Stillstand der Maschine während der Wartung gegen die Kosten bei Ausfall der Maschine abzuwägen. Um diese Entscheidung zu optimieren und genau dann Wartungen durchzuführen, wenn ein Ausfall bevorsteht, hat beispielsweise Thyssenkrupp eine Software entwickelt, welche die Daten aller Aufzüge des Unternehmens analysiert und Muster erkennt, wann ein Ausfall bevorsteht. Eine solche Software kann auch bei Produktionsmaschinen im Bereich Industrie 4.0 Ausfälle verhindern und Kosten sparen. Dort sind jedoch viele Ausfälle auf menschliche Fehler zurückzuführen, die derzeit nicht ausreichend in Algorithmen berücksichtigt werden (können) (Müller, 2018).

Audit Services

Unternehmen setzen KIs ein, um weitestgehend automatisiert Dokumente zu Durchsuchen und Schlüsselbegriffe zu identifizieren. Dies war bisher ein zeitaufwändiger manueller Prozess. Die maschinelle Verarbeitung natürlicher Sprachen liest wichtige Punkte in den Dokumenten. Die Technologie des maschinellen Lernens ermöglicht es, dieses System anhand einer Reihe von Beispieldokumenten zu trainieren, damit es lernt, wie Informationen automatisiert identifiziert und extrahiert werden.

3.7 Finanzen, IT und Software-Engineering

Auch wenn sich zahlreiche unabhängige Anwendungsbeispiele für KI in den drei Bereichen der 1) Finanzen, 2) IT und 3) Software-Engineering finden lassen, werden diese nachfolgend gebündelt dargestellt, da es primär um das Anwenden von ML-Verfahren auf digitalen Gegenstandsbereichen betrifft.

3.7.1 Prämien

Echtzeit-Prämien

Klassischerweise beruhen Versicherungsprämien auf der Risikobewertung historischer Daten. Dabei können aktuelle Trends entweder gar nicht oder durch menschliche Arbeit berücksichtigt werden. In Zeiten von immer größerer Datenverfügbarkeit ist es

möglich, Versicherungsprämien in Echtzeit auf Basis einer Vielzahl von Parametern anzupassen. So kann zum Beispiel der Beitrag einer Gesundheitsversicherung steigen, wenn man eine Wildwasserrafting-Tour geplant hat oder sinken, wenn man sich für einen Yoga-Kurs anmeldet. Beschränkungen für Echtzeit-Versicherungsprämien existieren derzeit primär beim Datenschutz und der Akzeptanz von Kunden, technisch können ML-basierte Algorithmen bei entsprechenden Rahmenbedingungen Versicherungsprämien bereits in Echtzeit berechnen (Miessler, 2018).

Financial Services

Zu den wichtigsten Anwendungsbereichen von KI im Bank- und Finanzdienstleistungssektor gehören die Früherkennung von Finanzrisiken und Systemfehlern sowie die Automatisierung zur Reduzierung Störungen in Finanzsystemen. Oft werden diese auch als *Smart Financial Services* bezeichnet. Beispiele sind in Marktmanipulationen, Cyber-Betrug, auf Anomalien fokussiertes Trading und die Reduzierung von Marktvolatilität und Handelskosten zu finden.

3.7.2 Entwicklung

Statische Codeanalyse

Quellcode ist eine Sammlung von Anweisungen und Funktionen, die von einem Programmierer geschrieben wurde. Die Analyse des Quellcodes kann entweder statisch oder dynamisch erfolgen (Rouse, 2010).

Die statische Codeanalyse, die oft als „White Box“-Testing bezeichnet wird, ist eine Auswahl von Algorithmen und Techniken, die zur Analyse von Quellcode verwendet werden. Sie arbeitet automatisch in einer Nicht-Laufzeitumgebung, um Schwachstellen wie versteckte Fehler oder schlechten Quellcode während des Entwicklungsprozesses aufzudecken (Nadeem, 2015). Auf diese Weise kann die statische Codeanalyse die Kosten für die Behebung von Sicherheitsproblemen reduzieren (DuPaul, 2017). Der beste Zeitpunkt für den Einsatz statischer Werkzeuge ist während der frühen Entwicklung, um Codequalität und Sicherheit zu gewährleisten (Cornell, 2008b). Abbildung 3-4 zeigt die Zuordnung der Methoden für die statische und dynamische Codeanalyse.

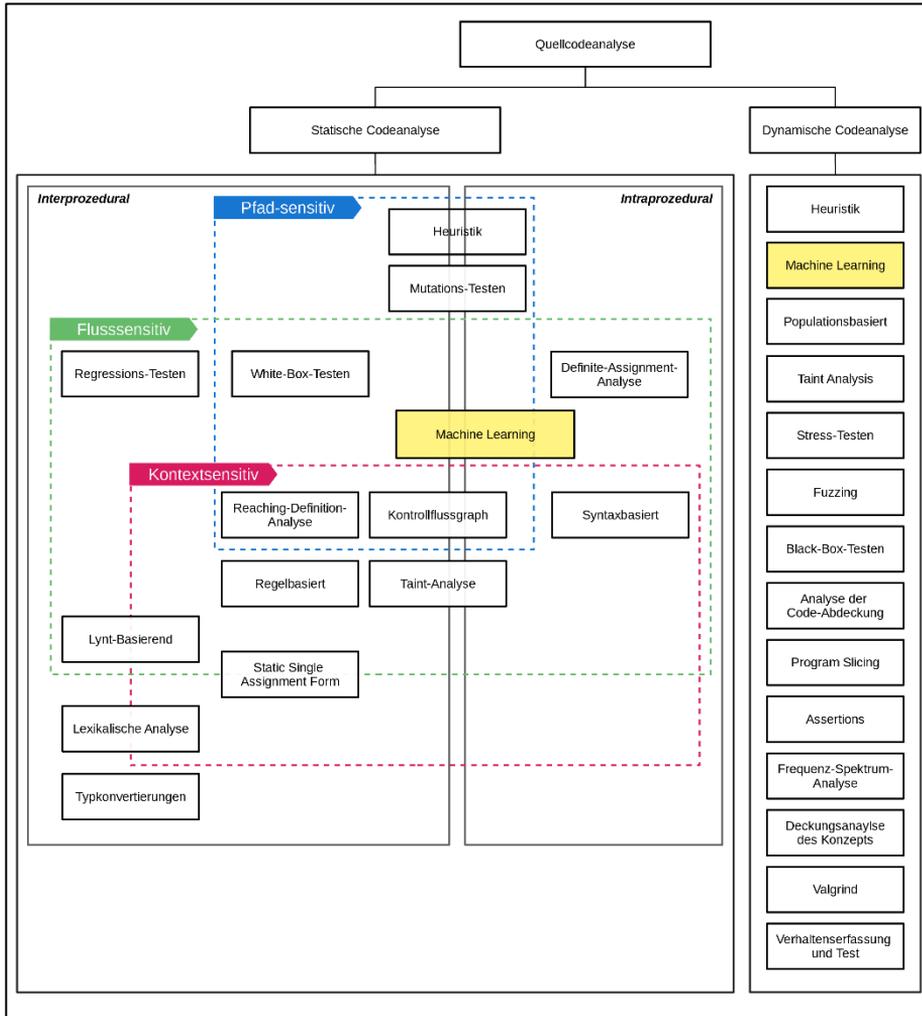


Abbildung 3-4: Codeanalyse – Zusammenhang statischer und dynamischer Codeanalysemethoden (Eigendarstellung).

Zum Beispiel kann die Methode „Reaching definitions“ entweder als interprozedural oder als fluss-, kontext- und pfadsensitiv klassifiziert werden. Die Methode Lint als weiteres Beispiel ist sowohl interprozedural als auch fluss-sensitiv und kann als kontextsensitiv, aber auch als nicht kontextsensitiv angesehen werden (daher wird Lint am Rand dargestellt). Heuristisches oder maschinelles Lernen verwenden Methoden sowohl aus der statischen als auch aus der dynamischen Codeanalyse. Maschinelle Lernverfahren sind inter- und intraprozedural, flussabhängig und können pfadsensitiv sein. Heuristik hingegen ist inter- und intraprozedural und pfadsensitiv.

Dynamische Codeanalyse

Die dynamische Codeanalyse (DCA) verfolgt den umgekehrten Ansatz und analysiert das Programm während des Betriebs. Funktionen werden mit Werten in den Variablen aufgerufen, während jede Codezeile überprüft wird und mögliche Verzweigungsszenarien erraten werden (DuPaul, 2017). Sicherheit kann durch gute Testszenarios angenommen werden, was beim Refactoring besonders wichtig ist. Ein weiterer Vorteil ist, dass die Dynamische Analyse Sicherheitslücken wie verbotene Speicherzugriffe und Pufferüberläufe behebt (ein voller Puffer führt dazu, dass Daten unbeabsichtigt überschrieben werden) (Neumann, 2015). Die Dynamische Analyse sollte während und nach der Bereitstellung verwendet werden, um die Live-Performance zu berücksichtigen oder potenzielle Probleme zu erkennen (Cornell, 2008a).

4 Auswahlmethodik von ML-Tools

Umgeben von einer Vielzahl von ML-Tools findet sich der Anwender schnell im Zielkonflikt von zeitaufwändigen Tool-Analysen über vielfältige Funktionsbereiche, der pragmatischen Auswahl des geeignetsten Tools, z.B. auf Empfehlung von Mundpropaganda oder Firmenpolitiken, sowie der Abwägung von Standardsoftware versus einer aufwändigen Individualentwicklung. Während letztere zumeist eigenen Ansprüchen genügt, stehen diesem der hohe Entwicklungsaufwand und die eingeschränkte Wiederverwendung gegenüber. In dieser Situation ist es empfehlenswert, sich einer Auswahlmethodik zu bedienen, die möglichst vielfältige Analyse Kriterien verschiedener Bereiche berücksichtigt, um zu einer bestmöglichen Toolauswahl für ein oder mehrere spezifische Probleme zu gelangen.

Einflussbereiche einer Auswahl von ML-Tools beziehen sich einerseits auf die unternehmensspezifische Situation und strategische Ziele aus der Unternehmensführung; andererseits auf das spezifische Projekt, in welchem Tools zur Entwicklung von ML-Systemen zum Einsatz kommen sollen. Weiterhin schränken die mehr oder weniger konkret spezifizierbaren Algorithmen, bzw. deren Verfügbarkeit, den Auswahlprozess weiter ein – eine Einschränkung die angesichts der hohen Zahl algorithmischer Ansätze schnell intransparent, komplex und nicht-nachvollziehbar wird.

Aus diesem Grund orientiert sich die Auswahlmethodik und deren Vorgehen an diversen Einflussbereichen, welche in nachfolgenden Unterkapiteln einzeln betrachtet werden. Ziel ist es somit nicht, eine umfassende Sammlung an Kriterien sowie deren konkrete Ausgestaltung zu geben, sondern Bereiche hierfür aufzuzeigen und mit Beispielen zu verdeutlichen. Das Kapitel ist somit wie folgt strukturiert.

In einem ersten Unterkapitel erfolgt die Darstellung einer unternehmensorientierten ML-Tool-Auswahl. Von dieser werden unternehmensspezifische Anforderungen für die Auswahl von ML-Tools abgeleitet. Im zweiten Unterkapitel erfolgt die Darstellung der projektorientierten-ML-Systementwicklung, von welcher phasenspezifische Anforderungen für die Auswahl von ML-Tools abgeleitet werden. Im dritten Unterkapitel erfolgt die Darstellung von ML-relevanten Algorithmen, welche als Algorithmen spezifische Anforderungen für die Auswahl von ML-Tools eingesetzt werden. Schließlich erfolgt die Bündelung von Anforderungen in Unterkapitel vier, sodass die Erfüllung von diversen Anforderungskategorien als Analyse Kriterien zur Auswahl von ML-Tools herangezogen werden können. Die Erarbeitung eines Vorgehensmodells zur Auswahl von ML-Tools vervollständigt das Kapitel und zeigt, wie das geschaffene Auswahlwerkzeug zum Einsatz kommen soll.

4.1 Unternehmensorientierte ML-Tool-Auswahl

Basierend auf der jeweiligen Ausgangssituation eines Unternehmens sind Einschätzungen eines ML-Tools zu relativieren. Bestehen beispielsweise bereits Verträge mit einem ML- und KI-Tool-Hersteller, oder sind bereits Erfahrungen im Einsatz bestimmter Tools im Unternehmen vorhanden, so ist unter Umständen die Wahl eines nach Funktionalität attraktiveren Tools nachrangig. Eine Argumentation sollte hier jedoch stets sachlogisch motiviert und nicht subjektiv erfolgen, sodass sich die Relativierung am Zweck des Vorhabens orientiert. Somit sind Einflüsse einer Unternehmung auf die Tool-Auswahl in einem Verfahren zur ML-Tool-Auswahl individuell zu berücksichtigen. Weiterhin werden von der Unternehmung unternehmensspezifische Anforderungen für die Auswahl von ML-Tools abgeleitet.

Auch wenn eine unternehmensspezifische Relativierung im Rahmen der unternehmensübergreifenden Marktstudie in Kapitel 5 keine Berücksichtigung finden kann, kann die hier vorgestellte Einschätzung als Grundlage dienen. Um den Entscheidungsprozess insbesondere für unerfahrene Anwender zu vereinfachen, wird der Entscheider anhand einer Reihe von Teilentscheidungen geleitet, um den komplexen Entscheidungsvorgang einer Relativierung auszuführen. Während Abbildung 4-1 einen Entscheidungsbaum für die unternehmensspezifische ML-Tool-Relativierung darstellt, werden die Einzelentscheidungen nachfolgend erläutert. Es sei an dieser Stelle angemerkt, dass die Reihenfolge für jedes Unternehmen individuell aufgebaut werden kann. Der Entscheidungsbaum, welcher somit aus einer beliebig großen Anzahl von Ebenen bestehen kann, resultiert in einer Auf- oder Abwertung für betrachtete Tools. Somit kann der Entscheidungsbaum also zur objektiven Begründung einer Entscheidung herangezogen werden.

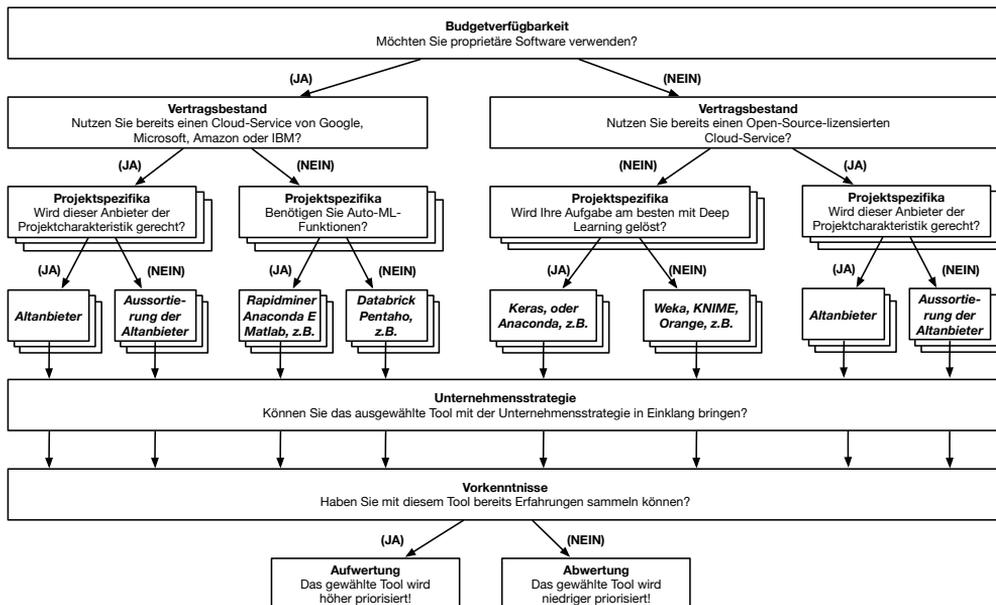


Abbildung 4-1: Beispiel eines Entscheidungsbaums für die Auswahl von ML-Systemen (Eigendarstellung).

Budgetverfügbarkeit

Erstens sollte entschieden werden, ob das Unternehmen ein Budget für den Kauf einer neuen maschinellen Lernsoftware hat, oder ob das Data-Science-Team die Auswahl von Open-Source-Plattformen fokussieren sollte.

Vertragsbestand

Zweitens ist es, wenn das Unternehmen bereits Cloud-Services nutzt und Lizenzen für Statistik- bzw. Visualisierungssoftware gekauft hat, sinnvoll, für diese bereits erworbenen Produkte zu prüfen, inwiefern Projekte auch mit diesen realisiert werden können, ggf. inklusive projektspezifischer Anpassungsimplementierungen.

Weiter fällt, wenn die Organisation plant mit einem ML-as-a-Service-System zur Analyse zu arbeiten, die Gestaltung einer Projektarbeit leichter sofern derselbe Anbieter sowohl für die Speicherung als auch für die maschinelle Auswertung gewählt wird. Dies reduziert z.B. den Zeitaufwand für die Konfiguration einer Datenquelle drastisch.

Projektspezifika

Drittens lassen sich einige Probleme mit Deep Learning besser lösen, andere benötigen klassisches ML. Deep Learning eignet sich besser, wenn die Funktionen in einem Datensatz nicht gut definiert, nicht unterscheidbar und zahlreich sind. Bei Problemen

wie Audioerkennung, Bilderkennung und Videoerkennung liegen die Features zahlreich oder in Form von Pixeln vor. Diese Features sind nicht klar definiert, nicht unterscheidbar und zahlreich. Für diese Probleme haben sich neuronale Netze besser als andere Methoden des maschinellen Lernens erwiesen.

Unternehmensstrategie

Viertens lassen sich strategische Einflüsse in die Toolauswahl integrieren. Beispiele hierfür sind in Tabelle 4-1 ersichtlich und stellen keine abgeschlossene Sammlung dar.

Tabelle 4-1: Unternehmensstrategische Einflüsse auf die KI-Toolauswahl.

Handlungen (Initiativen)	Bedingungen und Einflüsse
Käufe, Verkäufe	Wirtschaftsordnung, Verfassung
Preispolitik	Marktbedingungen, Konjunkturlage, Wettbewerbsordnung
Produktpolitik	Verwaltungsvorschriften
Werbemaßnahmen	Gesellschaftliche Faktoren
Service	Kulturelle Faktoren
Konditionen	Preisbindung, Lizenz-Gestaltung

Vorkenntnisse

Fünftens lassen sich die Kompetenzen und Erfahrungen der Unternehmensmitarbeiter berücksichtigen. Von erfahrenen Anwendern mit Programmierkenntnissen wird erwartet, dass sie Werkzeuge nutzen, an die sie bereits gewöhnt sind. Sofern das Unternehmen beschließt, ein neues ML-Tool auszuwählen, oder bis dato keine Erfahrung im Umgang mit ML hat, da es bspw. als Start-up noch keine Software eingeführt hat, ist es zielführender, die Einführung spezialisierte Software zu fokussieren.

Weitere Kriterien

Eine Reihenfolge der aufgeführten Kriterien ist veränderbar und um beliebig viele Teilscheidungen erweiterbar, sodass die unternehmensspezifischen Anforderungen eine Berücksichtigung finden können. Die initial gezeigte Abbildung stellt somit lediglich ein Beispiel zur Verdeutlichung des Vorgehens und möglicher Kriterien dar.

4.2 Projektorientierte ML-Systementwicklung

Die Auswahl von ML-Systemen sollte in jedem Fall an Ansprüchen orientiert werden, die sich über die gesamte Einsatzdauer ergeben. Als maßgebend stellen sich sämtliche

Phasen des Einsatzes eines ML-Systems heraus, welche zum einen im Folgenden erklärt und zum anderen zur Ableitung von Anforderungen herangezogen werden. Ausführungen beziehen sich auf den in Abbildung 4-2 dargestellten projektorientierten ML-Tooleinsatz.

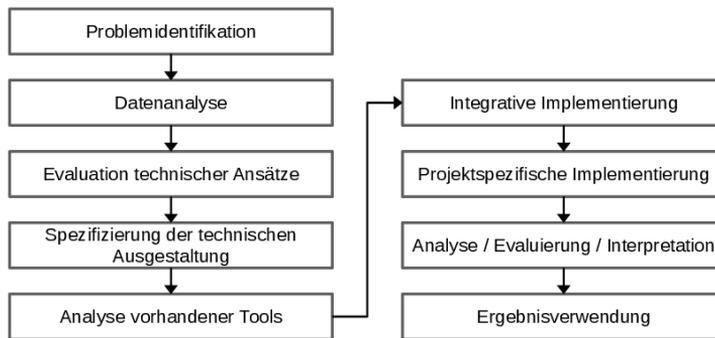


Abbildung 4-2: Projektorientierte ML-Systementwicklung (Eigendarstellung).

Obwohl sämtliche Phasen im Zuge der ML-Systementwicklung durchlaufen werden und eine stringente Abarbeitung phasenspezifischer Aktivitäten einer ML-Systemkonstruktion zu empfehlen ist, verlaufen ML-Projekte in den seltensten Fällen beim ersten Durchlauf erfolgreich, sodass eingangs formulierte Zielkriterien erfüllt werden. Vielmehr werden erarbeitete Inhalte solange einer Überarbeitung unterzogen, bis Zielkriterien erfüllt werden können. In Abgrenzung zu einem Trial-and-Error-basierten Vorgehen erfolgt eine Anpassung jedoch zielorientiert, sodass im Übergang von Phase zu Phase eine kontrollierte Rückführung von Inhalten und deren Überarbeitung möglich sind.

Problemidentifikation

Zunächst wird in der ersten Phase festgestellt, mit welchem Ziel und zu welchem Zweck ein ML-Projekt durchgeführt werden soll. Hierbei definieren Zielkriterien die erfolgreiche Realisierung des Projekts und dienen als Abbruchkriterien eines iterativen Vorgehens. Als Beispiel dient im Folgenden die Vorhersage der Traktion eines PKWs mit einer Abweichungsgenauigkeit von 3% auf nasser Fahrbahn.

→ Eine Auswahlmethodik von ML-Tools soll die systematische Möglichkeit einer Zielerfassung und Beurteilung der Zielerreichung thematisieren können.

Datenanalyse

Im Sinne eines datenbasierten Vorgehens folgt die Sichtung und Charakterisierung der Datenlage. Hierzu zählen das Feststellen des im Zuge des Projekts erfassbaren Datensatzes, die Auswahl der Lernmethode (unsupervised, supervised, semi-supervised, reinforced), sowie die Definition von Qualitätsanforderungen an diese. Weiter werden die zu erhebenden Daten, welche im Rahmen einer ML-Systementwicklung notwendiger Weise noch erhoben werden müssen, geplant und zur Realisierung vorgemerkt. Beispielsweise müssen neben dem Reifendruck, des Profilizustands und der Fahrweise des Fahrers Wetterinformationen, Temperatur- und Feuchtigkeitsparameter oder der Gummi-Art des aktuellen Reifensatzes erhoben werden.

→ Eine Auswahlmethodik von ML-Tools soll die systematische Erfassung der Datenlage adressieren und zwischen geplanten und bereits vorhandenen Daten unterscheiden.

Evaluation technischer Ansätze

Aus dem State-of-the-Art werden relevante technische Ansätze gewählt und konkrete ML-Algorithmen bzgl. der Eignung für das aktuelle ML-Projekt evaluiert. Bspw. wird ein Clustering-Ansatz gegen einen Deep-Learning-Ansatz gehalten, und analysiert, wie auf Basis der vorhandenen Datenlage Projektziele bestmöglich erreicht werden können. Ziel ist es, vorhandene Ansätze in einer Art Ranking zu charakterisieren und beste Kandidaten für eine iterative Realisierung vorzuhalten. Während beste Kandidaten für nachfolgende Phasen ausgewählt werden, stehen nicht realisierte Ansätze somit für spätere Iterationen zur Verfügung. Beispielsweise stellt sich der Deep-Learning-Ansatz als bester Kandidat heraus und wird in folgenden Phasen weiter spezifiziert.

→ Eine Auswahlmethodik von ML-Tools soll die systematische Evaluation von ML-Systemen thematisieren können sowie sich auf den State-of-the-Art beziehen.

Spezifizierung der technischen Ausgestaltung

Nachdem der grundlegende technische Ansatz gewählt wurde, erfolgt die Ausgestaltung der konkreten Technik. Beispielsweise sei dies bei der Wahl der Deep-Learning-Technik die Wahl der konkreten Art des Neuronalen Netzwerks, sowie dessen konzeptionelle Vorbereitung zur Verwirklichung. Beispielsweise erfolgt hier die Wahl einer spezifischen Netzwerkarchitektur.

→ Eine Auswahlmethodik von ML-Tools soll das Spezifizieren technischer Ausgestaltungen thematisieren können und sich auf den State-of-the-Art beziehen.

Analyse vorhandener Tools

Auf Basis konzeptioneller Anforderungen, welche in vorherigen Phasen abgeleitet wurden, erfolgt eine Überprüfung am Markt verfügbarer Tools. Weiter wird die Frage beantwortet, inwiefern ein einzelnes Tool oder sogar eine Kombination von Tools einen möglichen Implementierungsaufwand reduzieren können. Im Rahmen eines Rankings können beste Tools gewählt und nicht gewählte Tools für spätere Iterationen vorge­merkt werden. Beispielsweise wird eine Programmbibliothek gewählt, die spezifische Netzwerkarchitekturen und ein Datenmanagement-Tool vorhält, welches das Bereini­gen und Vorbereiten von Daten unterstützt, sowie ein Experiment-Management-Assis­tent, über welchen verschiedene Trainingsläufe systematisiert abgewickelt und Ergeb­nisse übersichtlich dargestellt werden können.

→ Eine Auswahlmethodik von ML-Tools soll die Möglichkeit der Evaluation gängiger ML-Tools sowie deren Kombination und Integrationsfähigkeit thematisieren können.

Integrative Implementierung

In dem wahrscheinlichen Fall, dass sich für das konkrete ML-Projekt eine Tool-Kombi­nation als attraktiv herausstellt, deren einzelnen Bestandteile eine aufwändige Imple­mentierung bedürfen würden, folgt die Zusammenführung. Im Ergebnis entsteht bei­spielsweise eine Implementierung, welche die LSTM-fähige Programmbibliothek mit einem Daten- und Experimentmanagement bereitstellt und für die Ausgestaltung erar­beiteter Konzepte vorbereitet ist.

→ Eine Auswahlmethodik von ML-Tools soll die Unterstützung einer integrativen Im­plementierung thematisieren können.

Projektspezifische Implementierung

Der Tooleinsatz erfolgt nun und erlaubt das projektspezifische Implementieren des konzeptionell designten ML-Systems. Hierzu zählt beispielsweise das Aufsetzen eines Algorithmus zur Datenbereinigung, Trainingsvorbereitung, das strukturelle Aufsetzen des technischen ML-Ansatzes, das Durchführen von Trainings- und Generalisierungs­aktivitäten sowie die Ergebnisdarstellung.

→ Eine Auswahlmethodik von ML-Tools soll die Unterstützung projektspezifischer Fra­gestellungen thematisieren können.

Analyse und Evaluation

Anschließend werden die Ergebnisse angesichts der initial festgestellten Problemfor­mulierung anhand von qualitativen und quantitativen Zielkriterien analysiert. Dabei wird evaluiert, inwiefern die Projektzielstellung erfolgreich realisiert werden konnte,

oder ob weitere Iterationen erforderlich sind und auf Basis, der bis dahin gemachten Erfahrungen Anpassungen erforderlich sind. Eine Interpretation fokussiert die Übertragung von ML-System-Outputs auf die originale Umwelt und ermöglicht es, Rückschlüsse auf die Funktionsfähigkeit des erstellten ML-Systems zu ziehen.

→ Eine Auswahlmethodik von ML-Tools soll Analyse, Evaluation und Interpretation der Daten unterstützen. Typische Kriterien sind hier bspw. das automatische Erstellen von Visualisierungen, das systematische Erfassen von KPIs und projektspezifischer Fragestellungen.

Ergebnisverwendung

Nach erfolgreicher Realisierung eines ML-Systems erfolgt die Übertragung in den originalen Kontext. Beispielsweise wird das ML-System in einen PKW transferiert und in den operationalen Einsatz gebracht. Weiterhin werden Maintenance- und Anpassungsarbeiten im Sinne einer kontinuierlichen Verbesserung berücksichtigt.

→ Eine Auswahlmethodik von ML-Tools soll die Unterstützung des Ausrollens eines ML-Systems betrachten, sowie bewerten, ob ein ML-Tool Maintenance- und Anpassungsarbeiten im operativen Betrieb unterstützt.

4.3 Algorithmische Verfügbarkeit

Kriterien einer algorithmischen Verfügbarkeit betreffen die in Kapitel 2 thematisierten Ansätze. Eine Auswahlmethodik von ML-Tools sollte somit die Verfügbarkeit von diversen Ansätzen und Methoden des ML betrachten, sodass Tools zielgerichtet bzgl. ihrer Funktionalität ausgewählt werden können. Weiter sollte die Auswahlmethodik die Verfügbarkeit von diversen KI-Methoden und -Ansätzen adressieren, sodass Tools entsprechend ihrer zugrunde gelegten KI ausgewählt werden können. Wie Abbildung 4-3 verdeutlicht, kann die Auswahl von relevanten Algorithmen mit Hilfe einer ML-Algorithmus-Taxonomie erleichtert werden, da zunächst Anforderungen auf einer abstrakten Ebene definiert und über beliebig viele Detaillierungsstufen bis auf die einzelnen Algorithmen heruntergebrochen werden kann. Somit gestaltet die in der Abbildung visualisierte Taxonomie lediglich ein Beispiel und realisiert den Anspruch auf Vollständigkeit, Aktualität und des Vollumfangs nicht.

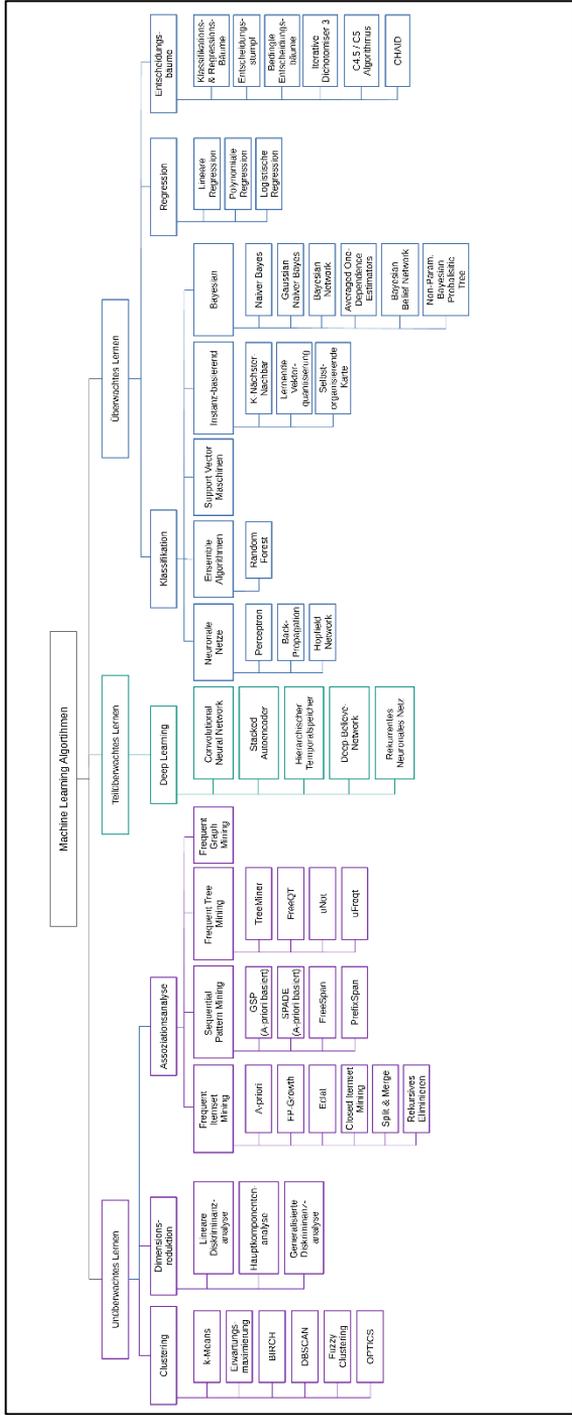


Abbildung 4-3: Beispiel eines taxonomischen ML-Algorithmus (Eigendarstellung).

4.4 Ebenen einer ML-Auswahlmethodik

Auf Basis der in den vorangegangenen Unterkapiteln dargestellten Einflussbereiche einer Auswahl von ML-Tools erfolgt in diesem Unterkapitel die Zusammenführung einzelner Analyse Kriterien und Anforderungen in ein einheitliches Auswahlwerkzeug. Auch wenn die unternehmensspezifische Relativierung von jedem Unternehmen individuell und im Nachgang vollzogen werden muss, so stellt der im nachfolgenden Kapitel erzeugte Marktüberblick von ML-Tools eine gute Ausgangsbasis für die Durchführung von ML-Projekten dar. Zur transparenten Darstellung werden identifizierte Kriterien hier in Ebenen aufgeteilt, die in Abbildung 4-4 visualisiert werden.

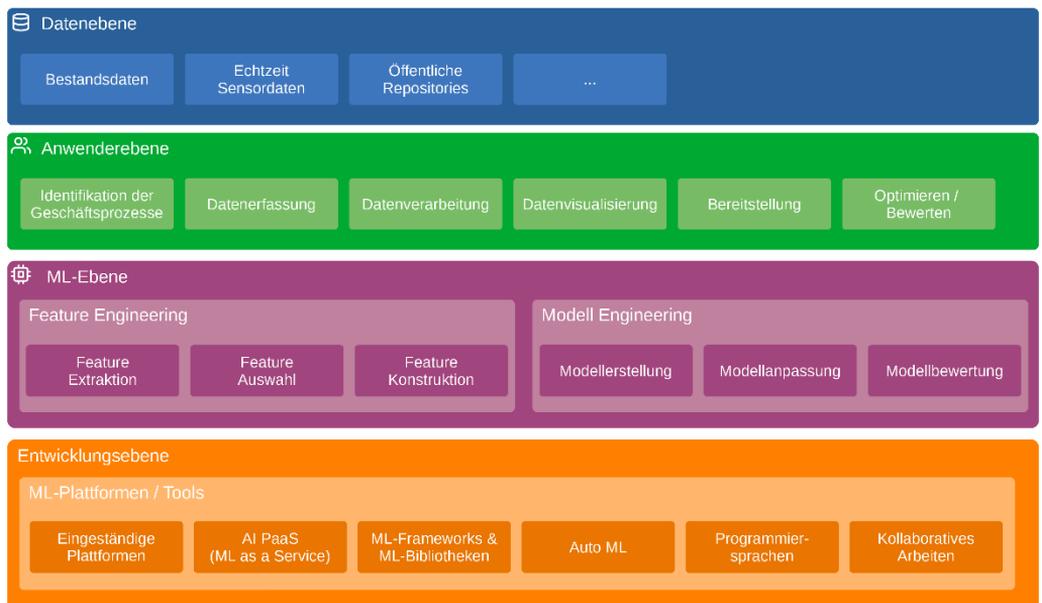


Abbildung 4-4: Ebenen-orientierte ML-Darstellung (Eigendarstellung).

In der Abbildung 4-4 werden insgesamt vier Ebenen deutlich, deren Unterbereiche nachfolgend erläutert werden. Die hier verwendete Farbgebung identifizierter Ebenen wird in nachfolgenden Kapiteln wiederverwendet und soll so stets eine Orientierung zur Zuordnung von Kriterien und Tools zu einzelnen Ebenen bieten.

Daten

Die Datenebene fokussiert Einflussfaktoren, die eine konkrete Ausgestaltung von Informations- und Rechtemanagement betreffen, sowie sich auf die in Kapitel 3 gereichten branchenspezifischen Lösungen und Anwendungsbereiche beziehen. Diese wurden

anhand der Tool-inhärenten Ausgestaltungsmöglichkeiten kategorisiert, sodass die Möglichkeit einer Arbeit mit „Bestandsdaten“, „Sensordaten“ in Echtzeit, sowie die Integration „Öffentlicher Repositories“ erfasst wird.

Anwenderebene

Die Anwenderebene fokussiert Einflussfaktoren, die aus dem Bereich des projektspezifischen Vorgehens kommen und in Kapitel 4.2 thematisiert wurden. Diese wurden anhand der Einsatzzwecke „Identifikation der Geschäftsprozesse“, „Datenerfassung“, „Datenvisualisierung“, „Bereitstellung“ und „Optimieren / Bewerten“ als Cluster dargestellt.

ML-Ebene

Die ML-Ebene fokussiert Einflussfaktoren der Verfügbarkeit spezifischer Algorithmen und Funktionen, welche in Kapitel 4.3 adressiert wurden. Entsprechend ihres Einsatzbereichs zum „Feature Engineering“ oder „Model Engineering“ und den entsprechenden Einsatzzwecken („Feature Extraktion“, „Feature Auswahl“, „Feature Konstruktion“ sowie „Modellerstellung“, „Modellanpassung“, „Modellbewertung“) werden diese im Folgenden als Cluster dargestellt.

Entwicklungsebene

Die Entwicklungsebene fokussiert Einflussfaktoren, welche die Charakterisierung der unternehmensspezifischen Ausgangssituation betreffen und in Kapitel 4.1 thematisiert wurden. Diese wurden entsprechend des Einflussbereichs als Cluster „Eigenständige Plattformen“, „AI PaaS“, „ML-Frameworks & ML-Bibliotheken“, „Kollaboratives Arbeiten“, „AutoML“ und „Programmiersprachen“ zusammengefasst.

4.5 Vorgehensmodell zum Einsatz des ML-Auswahltools

Die Anwendung des Ansatzes zur Auswahl von ML-Tools folgt einem stufenweisen Vorgehen. Während einzelne Stufen sequenziell realisiert werden können, sind Rückschlüsse zu bereits erarbeiteten Ergebnissen und Rücksprünge zu vorherigen Stufen möglich. Ein Rücksprung ist insofern sogar empfehlenswert, als dass z.B. Erkenntnisse aus der Evaluation einzelner Tools eine erneute Überprüfung bereits evaluierter Tools erfordert und der Gesamteinschätzung, im nächsten Kapitel als Marktüberblick bezeichnet, erträglich sind.

Phase 1: Anpassung an die unternehmensspezifischen Anforderungen

Das hier gereichte ML-Auswahltool ist zunächst an die unternehmensspezifische Ausgangssituation anzupassen. Gegebenenfalls erfordern einige Anforderungen die Ergänzung von Entscheidungsebenen (vgl. hierzu Kapitel 4.4), die Hinzunahme weiterer Einflussbereiche oder Einsatzzwecke (vgl. hierzu Kapitel 4.1 und Kapitel 4.2).

Phase 2: Toolergänzung und individuelle Toolüberprüfung

In dem in Kapitel 5 gereichten Marktüberblick müssen neben bereits analysierten Tools ggf. weitere Tools entsprechend den Bedürfnissen, Wünschen und Vorlieben des Unternehmens ergänzt werden. Die aktuelle Betrachtung nahe des State-of-the-Art erfordert somit eine fortwährende Aktualisierung.

Phase 3: Analyse und Vergleich ausgewählter Tools

Ausgewählte Tools werden nun anhand des ML-Auswahltools individuell analysiert und mit anderen Tools verglichen. Eine übergreifende Analyse über sämtliche Tools, entsprechend der gesetzten Analysebereiche inklusive des vergleichenden Abwägens von Vor- und Nachteilen auf Kriterienbasis, unterstützt hierbei die objektive Auswahl von ML-Tools.

Phase 4: Erstellung eines Rankings

Für eine Gegenüberstellung ist es empfehlenswert, Punkte zu vergeben, welche aggregiert und für einen Vergleich herangezogen werden können. Auf Basis der Punktvergabe kann somit ein Ranking erstellt werden, welches es ermöglicht, das attraktivste Tool auswählen zu können. Es ist hier unbedingt empfehlenswert, die Auswahl eines Tools nicht nur für ein spezifisches Projekt zu treffen, sondern es empfiehlt sich eine Szenario-basierte Berücksichtigung, welche eine nachhaltige und zukunftsgerechte Toolauswahl fördert.

Phase 5: Auswahl des besten ML-Tool

Einem Punkteranking folgend kann eine Toolauswahl systematisch und möglichst objektiv erfolgen. Es ist zu empfehlen, eine Auswahl von den besten Tools zu treffen und in Pilotprojekten zunächst zu überprüfen, inwiefern initiale Erwartungen an ausgewählte Tools erfüllt werden und ein ausgewähltes Tool Teil des Unternehmensstandards werden kann.

5 Marktübersicht

Um der stets wachsenden Anzahl von ML-Tools gerecht zu werden, welche ML-Grundlagen und KI-Methoden in variierender Bandbreite darstellen und eine individuelle, projektspezifische Ausgestaltung ermöglichen, sowie ML-Anwendungsbereiche mehr oder weniger spezialisiert berücksichtigen und Anforderungen individueller ML-Projektvorhaben erfüllen, wird im Folgenden ein Marktüberblick gegeben. Dabei sollen Anwendungseinblicke die Bereiche für eine Anwendung von ML in verschiedenen Kontexten, Anwendungsgebieten und Domänen sowie dessen Funktionsfokus gegeben werden.

Es folgt zunächst die Charakterisierung des methodischen Vorgehens zur Identifikation von ML-Tools. Die Realisierung des Marktüberblicks folgt im zweiten Teilkapitel und bietet einen Überblick der wichtigsten ML-Tools sowie deren Abgrenzung untereinander. Weiterhin verfolgt dies das Ziel die projekt- und unternehmensspezifische Auswahl von KI-Verfahren und ML-Tools entsprechend der im Kapitel 3 gezeichneten Auswahlmethodik vorzubereiten. Insbesondere API-basierte KI-Interfaces stellen einen spezialisierten Fokus dar.

5.1 Vorgehen zur Identifikation von Tools

Es geben Marktstudien eine überblicksartige Momentaufnahme eines betrachteten Marktes wieder. Demgemäß wurden im angestrebten Kontext zunächst die vorhandenen ML-Werkzeuge systematisch identifiziert. Dabei wurde der Suchraum hinsichtlich praktisch anwendbarer ML-Algorithmen spezifiziert und ermittelt, welche früheren Umfragen und ähnliche Untersuchungen als Grundlage hierfür dienen können. Zur Identifikation der Tools wurde eine Literatur- und Internetrecherche durchgeführt.

Nach der initialen Identifikation potenziell relevanter Tools wurde gemäß des PRISMA-Ansatzes (Moher, Liberati, Tetzlaff, Altman & Group, 2009) eine Vorauswahl durchgeführt, bei der einerseits Dubletten eliminiert und andererseits der Literaturkörper entsprechend der Abstracts – wenn vorhanden – oder dem manuellen Überfliegen des Textes gescreent wurde. Anschließend wurden die Volltexte hinsichtlich Eignung gründlich gelesen, sodass die finale Menge von 89 unterschiedlicher ML-Tools resultierte. Die berücksichtigten Tools wurden auf Grundlage ihrer Dokumentationen, Benutzerhandbücher und Demovideos entsprechend dem Klassifizierungsschema beschrieben, das auf den Schritten des ML-Workflows (siehe Kap. 4) basiert.

Fokale Aspekte der Toolbetrachtung und damit konkrete Ausprägungen der in Kapitel 4 gereichten Bereiche sind:

- Die Ausprägung des Produkttyps als Framework oder Bibliothek,
- der Anwendungsbereich der möglichen Softwareprojekte,
- technische Merkmale, wie z.B. das Betriebssystem, Lizenz, Interface-Verfügbarkeit, oder die zur Bedienung des Tools benötigten Programmiersprache,
- die Art der Methoden, Algorithmen und Modelle, die vom Tool unterstützt werden,
- die Benutzerfreundlichkeit, wie z.B. Visualisierungsmöglichkeiten, Rückmeldungen hinsichtlich einer Modifikation verschiedener Parameter, um die Präzision des Vorhersagemodells zu erhöhen, die Notwendigkeit von Programmierkenntnissen sowie vorhandene Hilfsmittel,
- die Kompatibilität, wie zum einen die Exportfunktionalität zu anderen Anwendungen sowie zum anderen die Möglichkeit das Projekt mit weiteren Benutzern teilen bzw. kollaborieren zu können.

Dem PRISMA-Ansatzes folgend wurden von der Liste Tools ausgeschlossen (Moher et al., 2009), die weniger als 1% der Stimmen erhielten. Als Teil dieser Recherche wurde die 20. KDNuggets-Umfrage aus dem Jahr 2019 analysiert (Piatetsky, 2019). Darin wurde bei den Teilnehmern abgefragt, welche Analyse-, Data-Science- und ML-Lernsoftware diese im vergangenen Jahr verwendet haben (Piatetsky, 2019). Es beteiligten sich über 1.800 Teilnehmer, wobei diese im Durchschnitt ca. sechs verschiedene Tools nutzten. Die zuvor ausgewählte Datenbasis wurde um aus Sicht der Autoren wichtige Tools aus der KDNuggets-Umfrage ergänzt bzw. aktualisiert. Tabelle 5-1 zeigt wie der Toolbestand der KDNuggets-Umfrage hierfür erweitert, angereichert, aktualisiert und abgewandelt wurde.

Tabelle 5-1: Abweichungen und Ergänzungen gegenüber KDNuggets-Umfrage (Piatetsky, 2019).

	Toolauswahl	Begründung
Tools, die gegenüber KDNuggets entfernt wurden.	CatBoost, LightGBM, Microsoft SQL Server, SQL on Hadoop tools, Unix shell/awk, XGBoost	Tools sind zu allgemeingültig, zu klein oder zu spezifisch auf ein Problem fokussiert
Tools, die mittels Domänenexperten ergänzt wurden.	AccordNet, Apache Heron, Apache Mahout, Apache Singa, DyNet, ELKI Framework, Gluon, Google Colab, Ilastik, Kubeflow, Massive Online Analysis, Metaflow, ML.NET, MIPack, ONNX, PyCaret, PyXLL, RapidMiner, SAS Viya, Shogun, SPMF, Tribuo, Turi Platform	Wichtige, teils sehr aktuelle Tool-Novizen auf dem Markt
Tools, die umbenannt, konkretisiert oder aktualisiert wurden.	R Language → R Platform; Solver → Analytic Solver; Flink → Apache Flink; Weka → Weka Machine Learning Workbench; QlikView → QlikSense; MLlib → Apache Spark MLLib; Altair/Datawatch/Angoss → Altair Knowledge Works;	Aktualisierung bzw. Transformation von Tools
Tools, die zusammengefasst wurden.	Hadoop Commercial Tools und Hadoop Open Source Tools → Apache Hadoop; Tibco Data Science und Spotfire → Tibco Data Science / Spotfire;	Zusammenfassung von Tools

Neben den Änderungen und Neuerung aus Tabelle 5-1 wurden die allgemeingültigen Programmiersprachen (z. B. Python, JavaScript, ...) die in der KDNuggets-Umfrage aufgelistet wurden, nicht in die Toolauswahl einbezogen, da sie keine konkreten ML-Tools oder ML-Frameworks darstellen. Vielmehr können diese Sprachen innerhalb oder im Zuge unterschiedlicher ML-Tools, ML-Frameworks und ML-Bibliotheken eingesetzt werden.

5.2 Toolübersicht

Während die in der Marktübersicht analysierten ML-Tools und Frameworks in Abbildung 5-1 bis Abbildung 5-4 als platzsparende Übersichten visualisiert werden, adressiert das hier gezeigte Kapitel den Gesamteindruck, der auf den detaillierten Toolbeschreibungen nachfolgender Kapitel basiert. So zeigen die Übersichtsabbildungen sämtliche ML-Tools und Frameworks im Vergleich und die in Kapitel 4.4 geschaffenen Ebenen einer ML-Auswahlmethodik finden sich in den Abbildungen auf den einzelnen Zeilen wieder. Eine Farbgebung in den Übersichten folgt dem in Abbildung 4-4 eingeführten Farbschema und dient der Zuordnung von Kriterien zu einzelnen Ebenen. Jedes Kriterium ist hierbei mit einer eigenen Zeile repräsentiert.

Spalten in der Abbildung repräsentieren die ausgewählten Tools. Die Erfüllung einzelner Kriterien für ein Tool wird mit einem grauen Balken hervorgehoben, sodass mittels der Markierungen das Erfüllen des jeweiligen Kriteriums für ein Tool ersichtlich wird. Einzelne Tools können teilweise über verschiedene Cloud- bzw. Hosting-Dienstleister zur Verfügung gestellt werden können (siehe IaaS-Ebene). Eine genauere Betrachtung der Bereitstellungsmöglichkeiten ist abhängig vom Projektkontext und damit dem Ziel des Einsatzes. Da Bestandsdaten, Sensordaten sowie öffentliche Daten von dem Kontext der zu entwickelnden Anwendung spezifiziert werden (siehe Daten-Ebene), sind die betreffenden Kriterien im Überblick allgemein gehalten.

Im Folgenden werden einige, aus Sicht der Autoren, nennenswerten Tools herausgestellt, wobei eine Nennung keine Aussage über die Relevanz des jeweiligen Tools treffen soll. Dies ist aufgrund der unterschiedlichen Einsatzzwecke und Projektziele nicht zielführend, jedoch als Orientierung dienlich. Weiterhin ist die Reihenfolge der Tools in den Abbildungen 5-1 bis 5-4 nicht von Bedeutung. Die Tools wurden im Rahmen der Recherche platziert.

In den in Abbildung 5-1 definierten Tools finden sich gängige Größen im Bereich des ML wie TensorFlow, Keras, Scikit-learn, PyTorch. Darüber hinaus sind auch essenzielle Umgebungen wie Anaconda oder MATLAB vertreten. Zudem sind große Plattformen

wie Microsoft Power BI, SAS Base, die IBM SPSS Produktfamilie sowie Microsoft Azure ML vorhanden.

In Abbildung 5-2 sind die ML-Frameworks FastAi, Apache MxNet, Microsoft CNTK hervorzuheben. Als Plattformen sind der SAS Enterprise Miner, Amazon SageMaker, BigML, Dataiku, IBM DSX, Pentaho sowie Google Auto ML zu nennen, die in einem näheren Blick vielversprechend erscheinen.

Angesichts der in Abbildung 5-3 dargestellten Tools und Frameworks kann im Bereich der ML-Frameworks DyNet sowie ML.NET herausgestellt werden. Mit ONNX findet sich ein noch recht neues, aber vielversprechendes Format zu Austausch von Deep-Learning-Modellen in der Übersicht. Im Bereich der Plattformen sind unter anderem der Microsoft ML Server, TIBCO Data Science und Spotfire sowie Sas Viya und Looker zu finden. Weiterhin finden sich elementare Bausteine wie Mathematica oder die R-Plattform in der Übersicht.

Mit Metaflow findet sich ein vielversprechendes ML-Framework aus dem Hause Netflix in Abbildung 5-4. Als neue Plattformen finden sich Kubeflow für das Betreiben von ML-Projekten auf Kubernetes und Google Colab für das kollaborative Arbeiten an Data-Science-Projekten in der Übersicht.

In nachfolgenden Kapiteln werden die in Abbildung 5-1, Abbildung 5-2, Abbildung 5-3 sowie Abbildung 5-4 gezeigten ML-Tools, Plattformen und Frameworks im Detail beschrieben. Während breite Plattformen oftmals diverse Bereiche abdecken und somit komfortable Lösungen bieten, gehen diese eventuell mit einem Vendor-Lock-In-Effekt einher. ML-Tools und Plattformen, die sehr neu sind, sind zu Teilen sehr vielversprechend. Allerdings muss von Fall zu Fall abgewogen werden, ob ein produktiver Einsatz bereits sinnvoll ist. Generell ist ersichtlich das viele elementare ML-Tools und Frameworks als Open-Source-Lösungen zur Verfügung stehen, die durch eine aktive Entwicklergemeinde stetig weiterentwickelt werden. Die ebenenorientierten ML-Auswahlmethodik dient als Übersicht, wie und welche Tools für welchen Bereich bei der Erstellung und dem Betrieb von eigenen KI-Lösungen verwendet werden kann. Um eine transparente Darstellung der Tools zu gewährleisten, werden diese anhand der allgemeinen KI-Methoden und -Ansätze von Kapitel 2 gruppiert dargestellt.

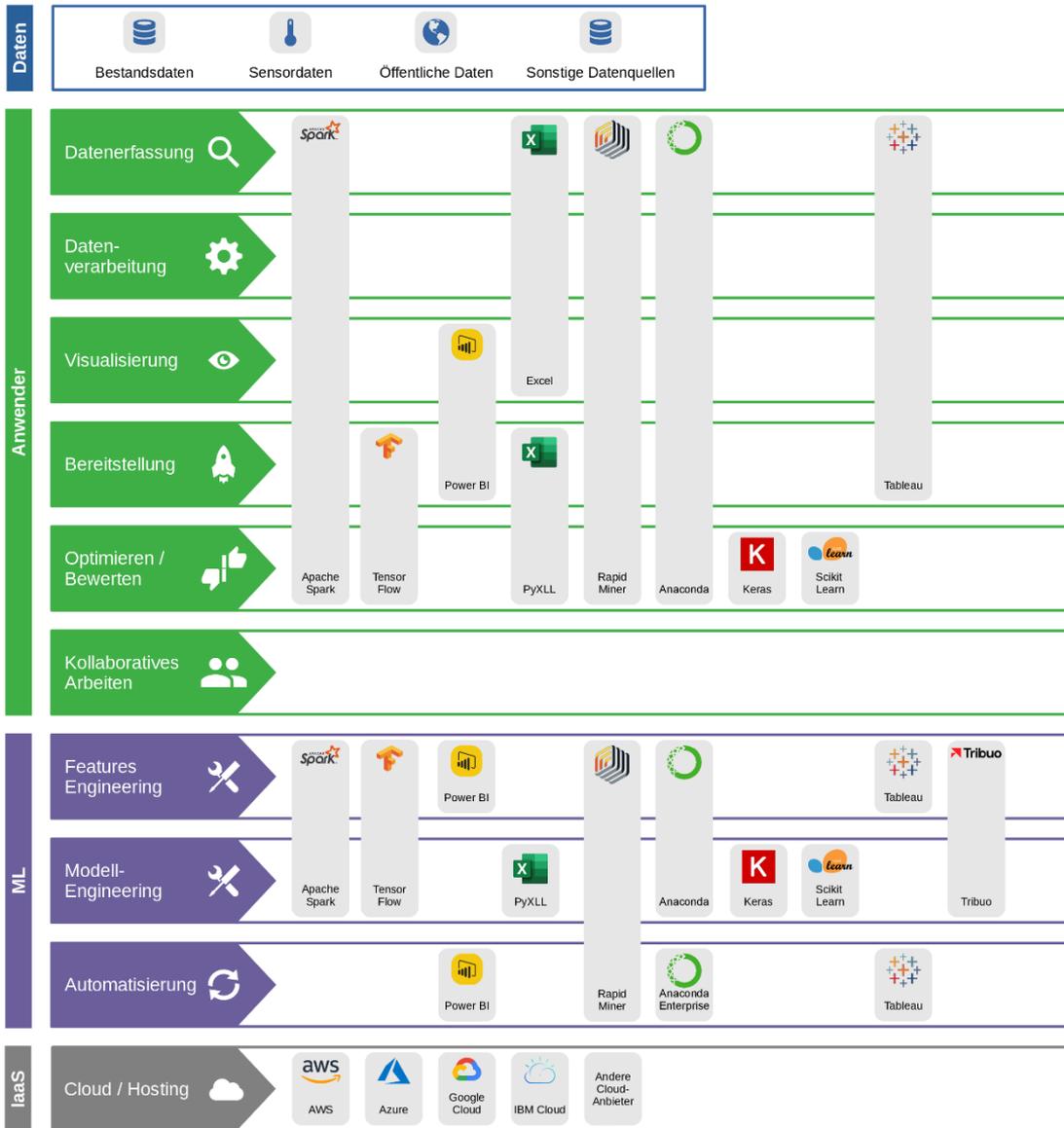


Abbildung 5-1: Übersicht ML-Frameworks Teil 1 (Eigendarstellung).

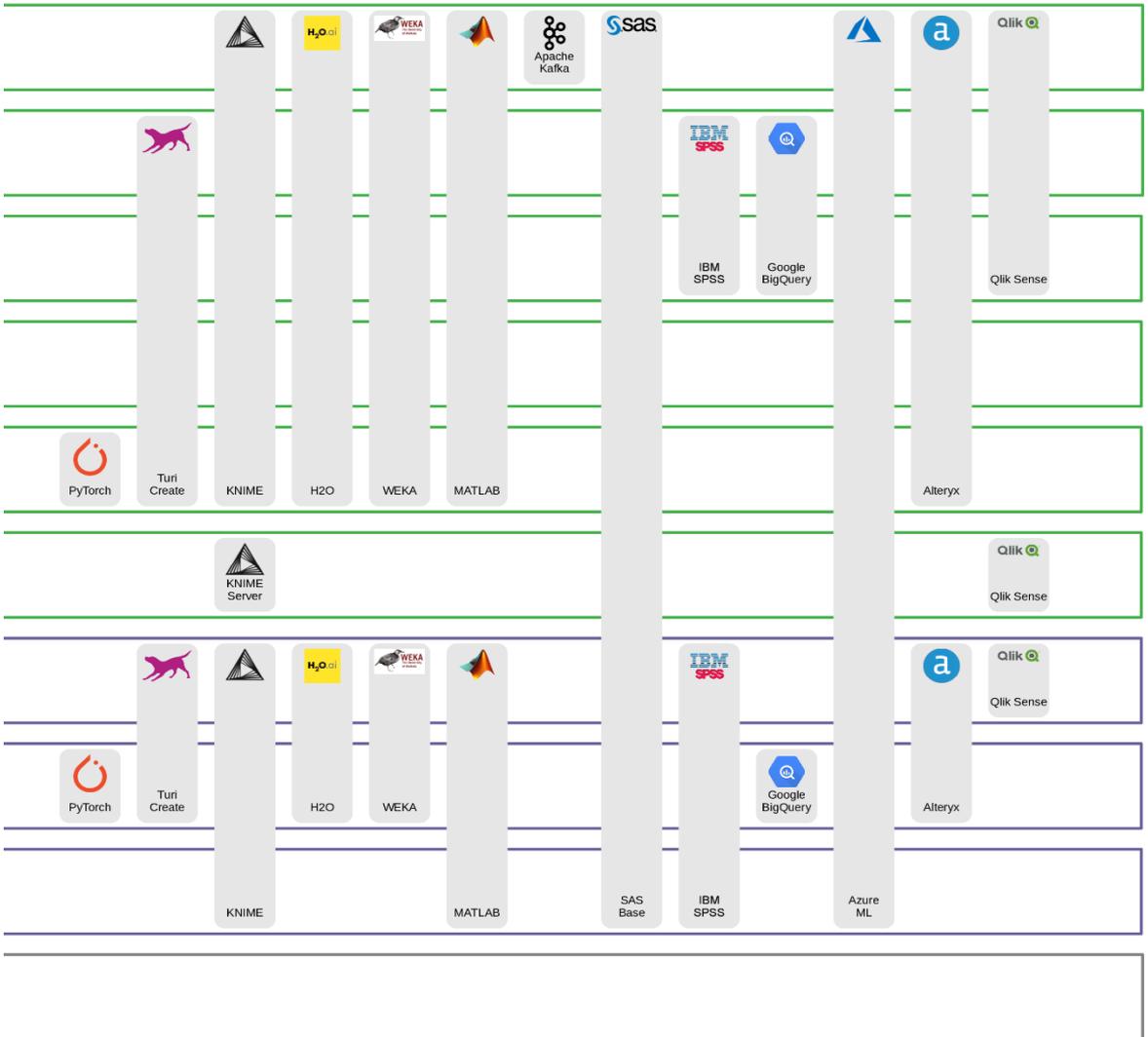


Abbildung 5-1: Übersicht ML-Frameworks Teil 1, Fortführung (Eigendarstellung).

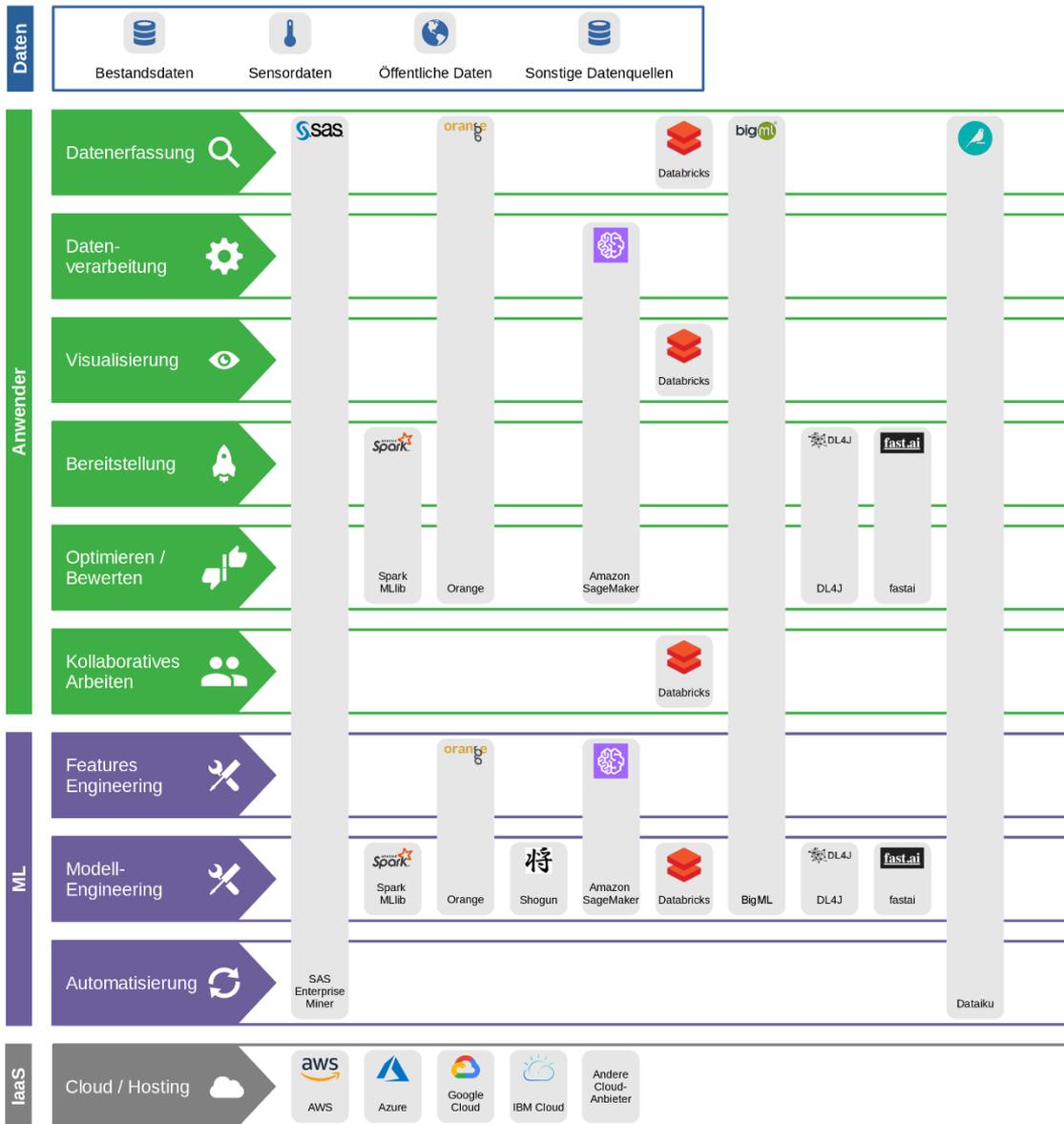


Abbildung 5-2: Übersicht ML-Frameworks Teil 2 (Eigendarstellung).

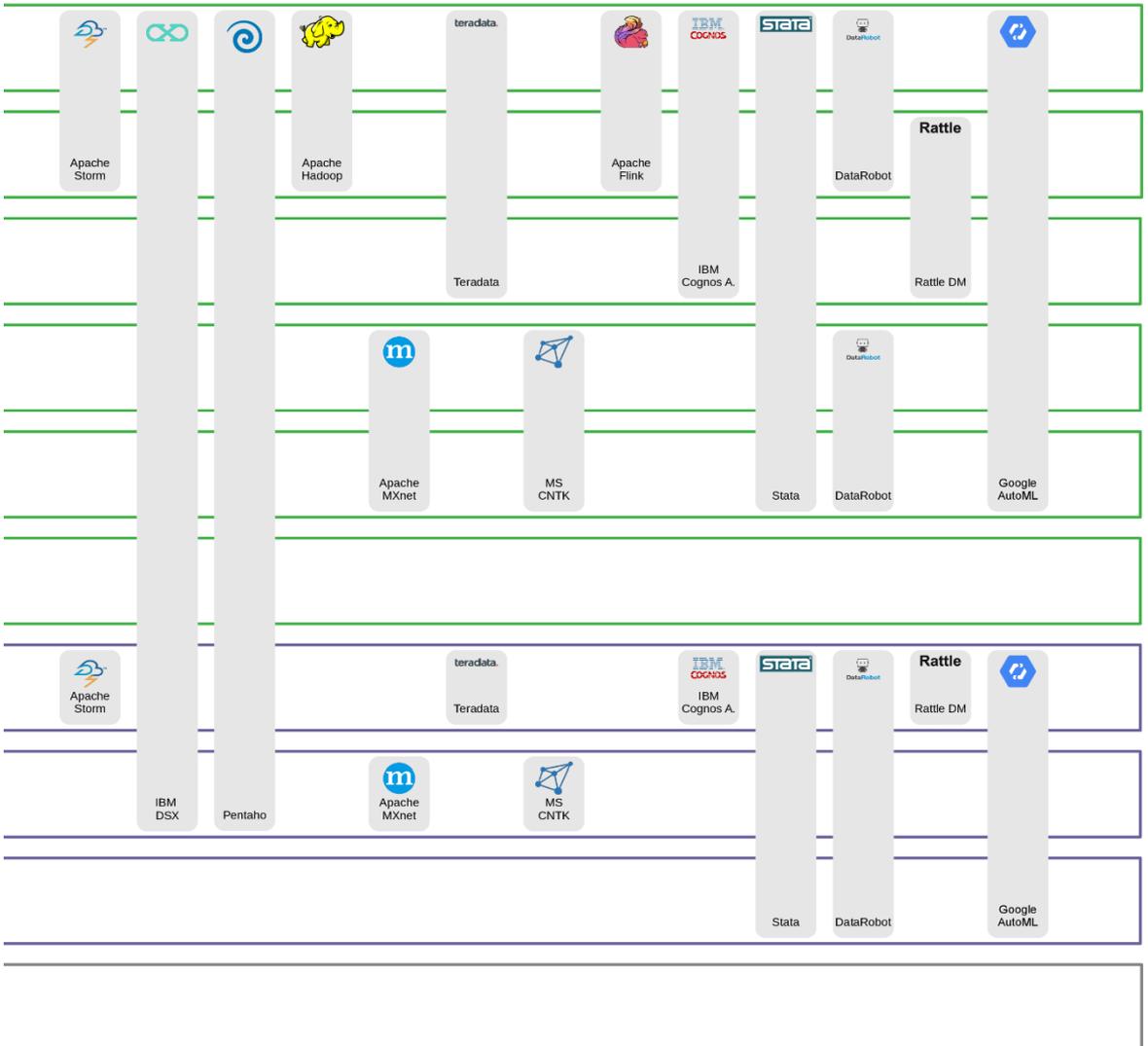


Abbildung. 5-2 Übersicht ML-Frameworks Teil 2, Fortführung (Eigendarstellung).

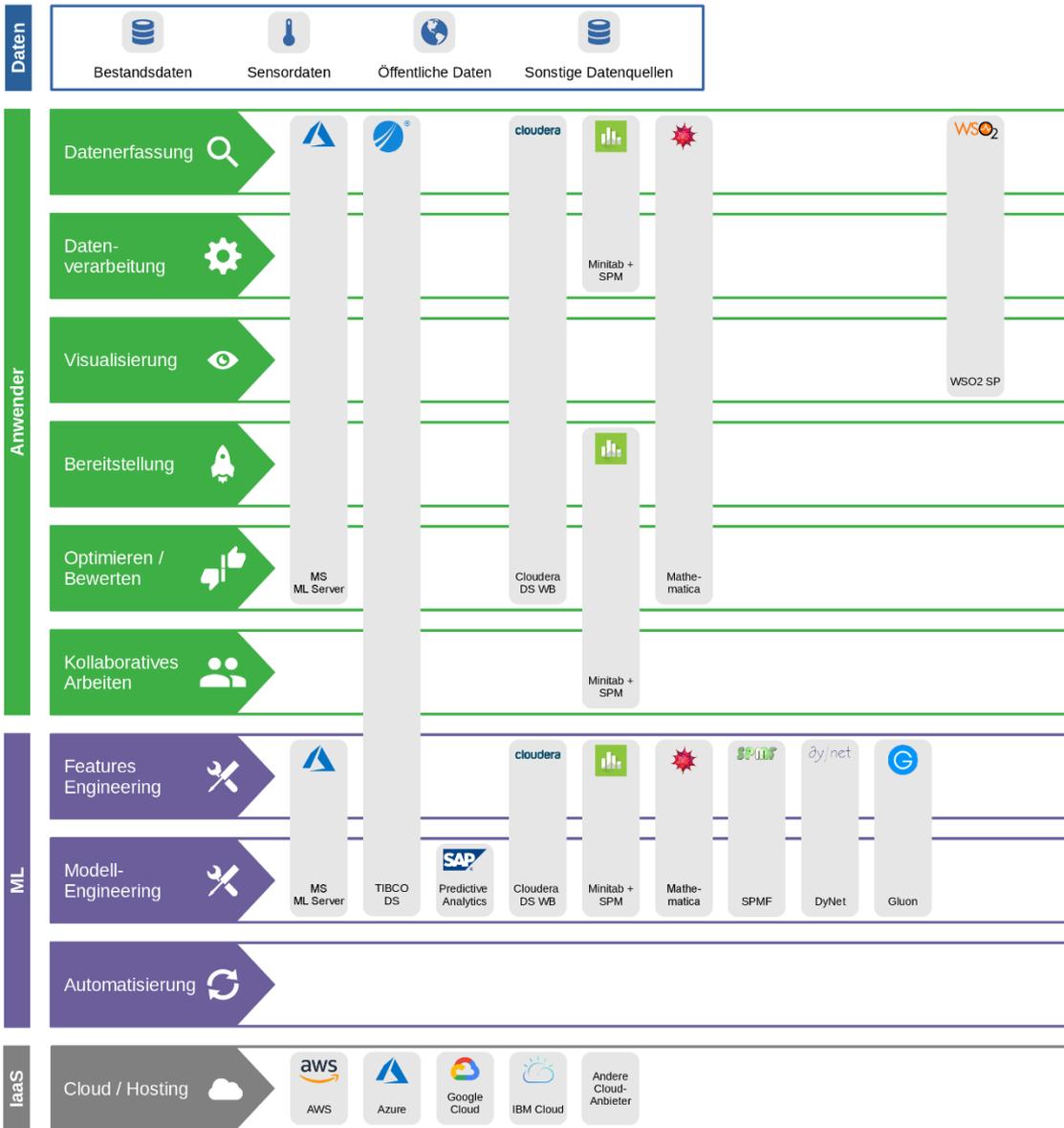


Abbildung 5-3: Übersicht ML-Frameworks Teil 3 (Eigendarstellung).

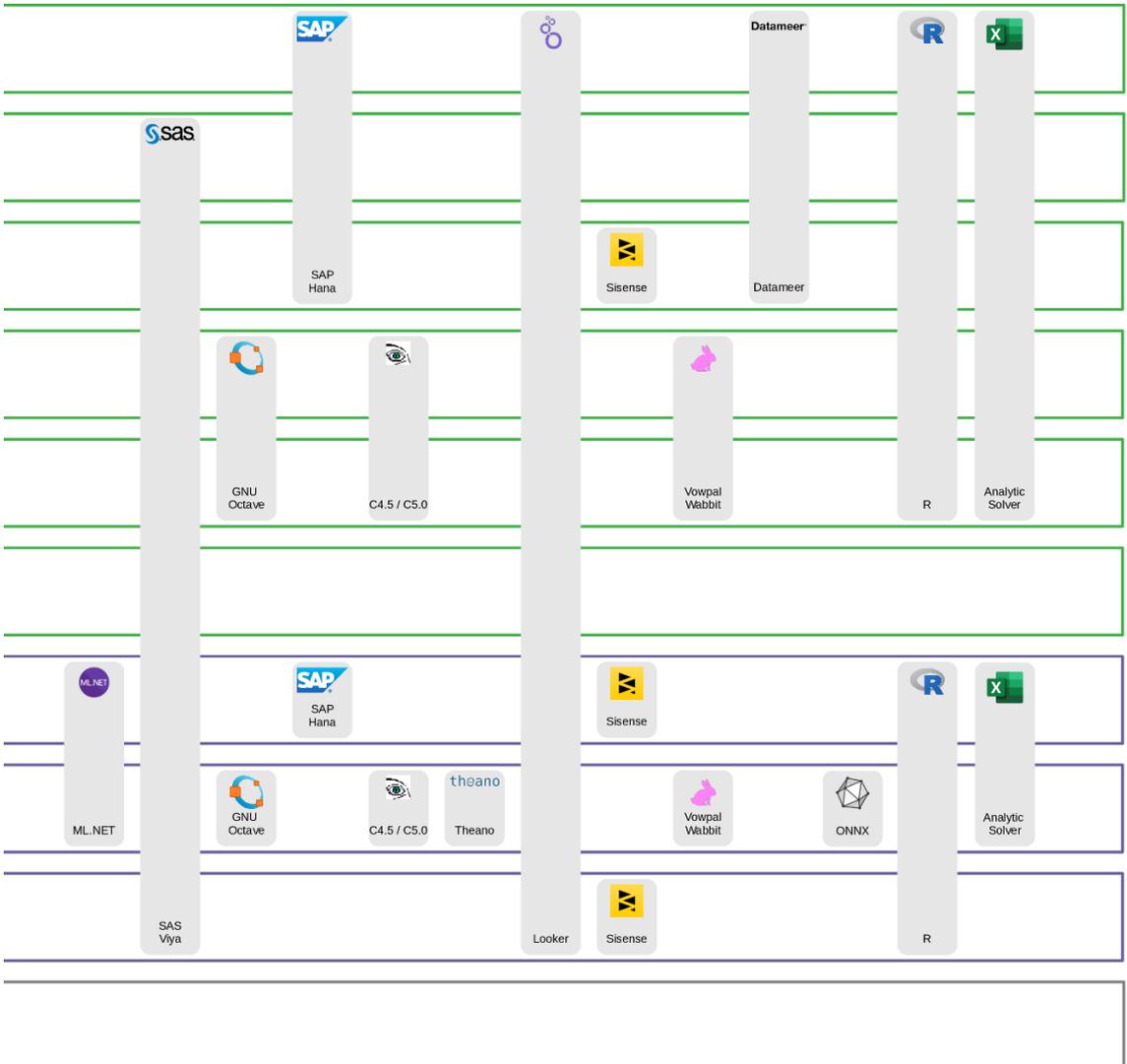


Abbildung 5-3: Übersicht ML-Frameworks Teil 3, Fortführung (Eigendarstellung).

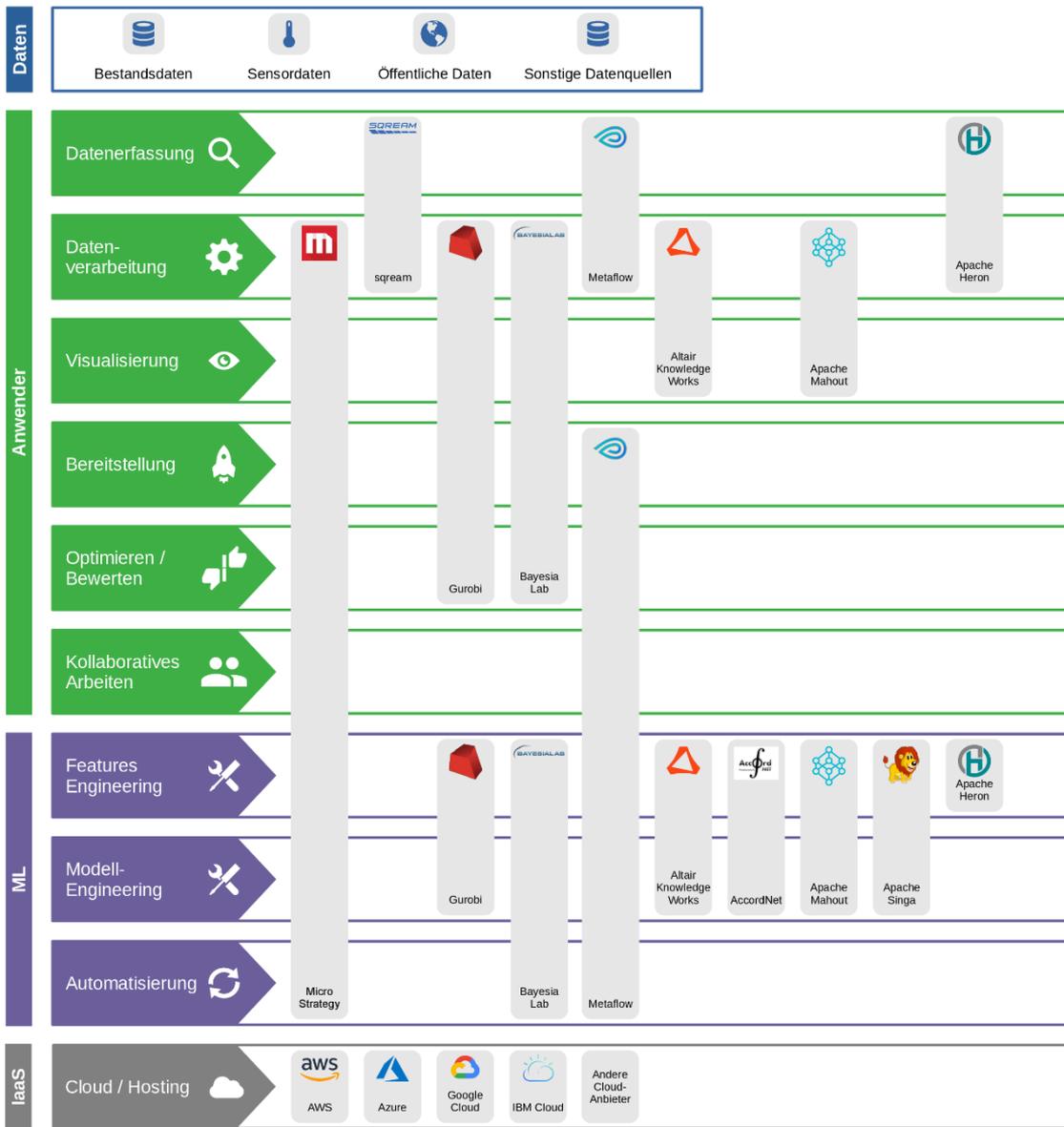


Abbildung 5-4: Übersicht ML-Frameworks Teil 4 (Eigendarstellung).

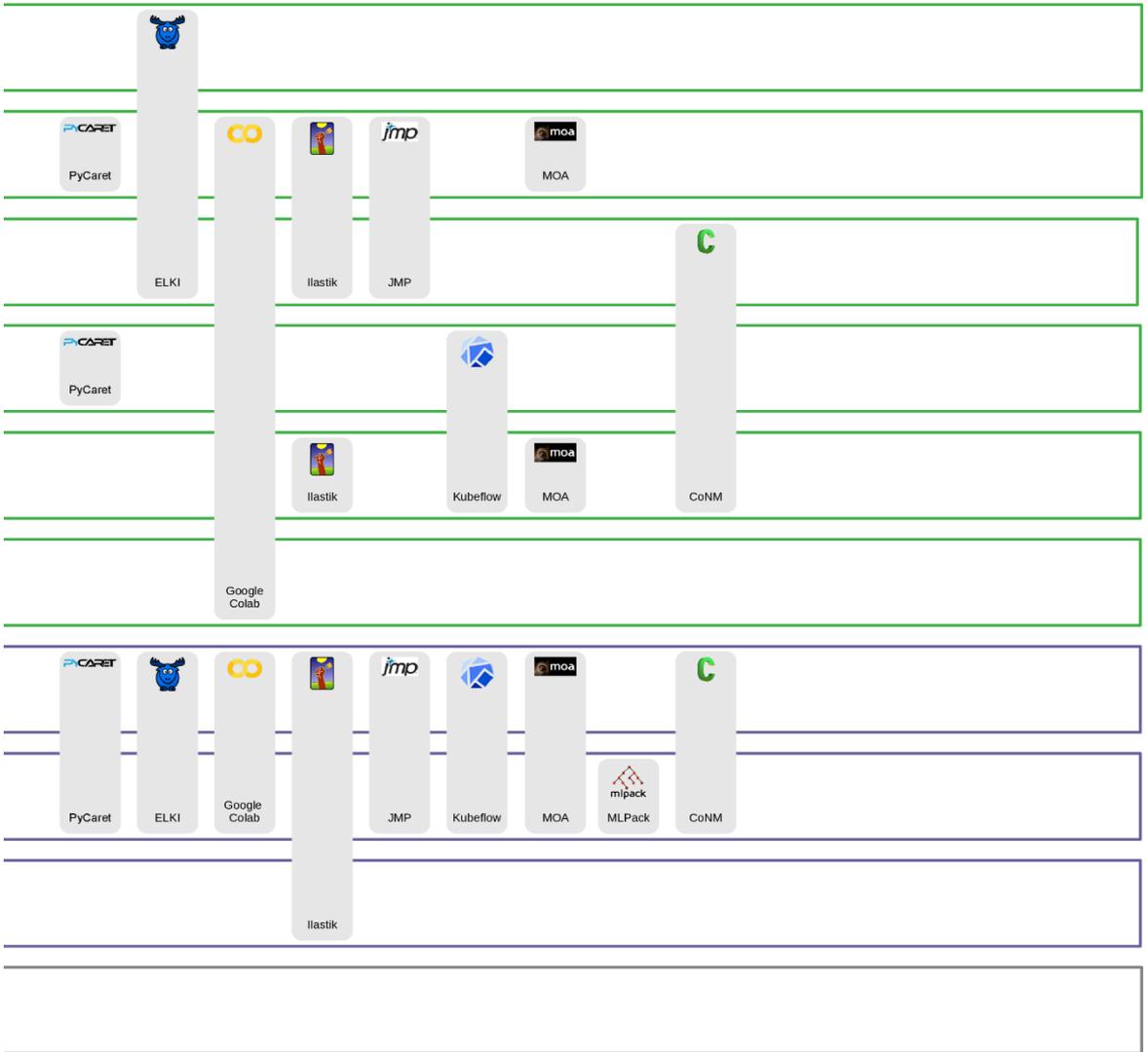


Abbildung 5-4: Übersicht ML-Frameworks Teil 4, Fortführung (Eigendarstellung).

5.3 Tools des Maschinellen Lernens im Detail

AccordNet

AccordNet ist ein quelloffenes C# .NET ML-Framework. Die Einsatzzwecke reichen dabei von Computer Vision, über Signalverarbeitung bis hin zu statistischen Auswertungen. Genauer werden unter anderem Algorithmen für die Klassifikation, die Regression, das Clustering, für Wahrscheinlichkeitsverteilungen sowie die Bild- und Audioverarbeitung bereitgestellt (Souza, Kirillov, Catalano & Accord.NET contributors, 2014). Eine Vielzahl unterschiedlicher Demo-Anwendungen auf der Herstellerseite (Accord.NET contributors, 2017) erleichtert den Einstieg in das Framework und zeigt die Anwendungsmöglichkeiten. An dieser Stelle sei jedoch erwähnt, dass das Projekt nicht mehr weiter entwickelt wird (Accord.NET contributors, 2020). Bevor ein neues Projekt auf Basis dieses Frameworks initiiert wird, sollte also grundlegend der Status von AccordNet überprüft und Risiken abgewogen werden.

Altair Knowledge Works

Unter dem Namen Altair Knowledge Works vereint Altair zwei Produkte: Den Altair Knowledge Seeker und das Altair Knowledge Studio. Beim Altair Knowledge Seeker handelt es sich um eine Desktop-Anwendung zur Visualisierung, Datenmodellierung, Datentransformation sowie zur Erstellung analytischer Modelle. Des Weiteren unterstützt diese bei der Identifizierung von prädiktiven Attributen und ermöglicht das Verwenden der Sprachen R, Python und Java (Altair Engineering Inc., 2020b). Eine Stärke der Anwendung liegt zudem in dem Erstellen von Ensemble-Entscheidungsäumen, also der Kombination von unterschiedlichen Arten einen Entscheidungsbaum zu erstellen. Dabei können diese mit den Algorithmen „Bagging“, „Boosting“ und „Random Forest“ automatisiert erstellt werden.

Altair Knowledge Studio ist eine Plattform zur prädikativen Analyse sowie für maschinelles Lernen. Die Plattform unterstützt viele unterschiedliche ML-Themenbereiche wie etwa Sentiment-Analyse, Scoring-Modelle, Neuronale Netze (Altair Engineering Inc., 2020a). Dabei muss sich nicht für eine Programmiersprache entschieden werden, vielmehr können alle unterstützten Sprachen verwendet werden. Für das Altair Knowledge Studio wird zudem auch eine Integration in Apache Spark angeboten.

Alteryx

Alteryx ist eine Analyse-Plattform, die Daten entdecken, vorbereiten und analysieren kann. Die Plattform hat ihre Stärken vor allem in der Visualisierung und Aufbereitung der Daten. Mit Alteryx Workflow lassen sich Daten aus unterschiedlichen Quellen wie etwa Hadoop oder Excel zusammenführen (Alteryx, 2020). Die Erstellung kann dabei

dank dem Alteryx Designer auch gänzlich ohne das Schreiben von Code erfolgen. Mittels vorhandenen Integrations- und Transformationstools kann der entstehende Datensatz aufbereitet werden. Weiterhin lassen sich über die Plattform statistische Prognosen und Analysen erstellen, welche entweder direkt zusammengebaut oder mit der Sprache R implementiert werden können. Der entstehende Analyseprozess kann schlussendlich mit anderen Nutzern geteilt werden. So können diese den Prozess für sich nutzen. Neben eigenen Visualisierungsmöglichkeiten wird auch ein Export in andere etablierte Visualisierungslösungen, wie etwa Tableau, Qlik oder Microsoft Power BI, unterstützt (Alteryx, 2020).

Amazon Machine Learning / SageMaker

Amazon bietet mit seiner Cloud-Produktpalette eine Vielzahl von KI-Optionen an. Mit den KI-Services existieren vordefinierte Lösungen für Computer Vision, Sprache, Empfehlungen, sowie von ML-basierten Vorhersagen (Amazon Web Services, 2020c). Unter Amazon Sage Maker vermarktet Amazon eine Plattform, die es ermöglicht, auf einfache Art und Weise ML-Modelle unterschiedlicher Größenordnungen zu erstellen, zu trainieren und bereitzustellen. Bei der Erstellung kann auf einen Pool an vordefinierten und damit optimierten Algorithmen zurückgegriffen werden, für die Amazon eine schnelle Laufzeit verspricht. Amazon SageMaker Ground Truth unterstützt insbesondere beim Labeling-Prozess von Eingangsdaten. Training und Bereitstellung des Modells können gänzlich automatisch erfolgen (AutoML). Mittels des Amazon SageMaker Neo kann das Modell einmal trainiert und auf unterschiedlicher Hardware ausgeführt werden. Die Plattform sowie die dort erstellten Modelle sind nahezu beliebig skalierbar. Zudem ermöglicht der Bereitstellungsprozess automatisierte Tests und das automatische Allokieren neuer Ressourcen, wenn das Modell skalieren muss. Insofern bestehende (Open-Source)-Frameworks genutzt werden sollen, kann auf die AWS Cloud zurückgegriffen werden. Amazon bietet unter anderem etablierte Frameworks, wie z.B. TensorFlow, PyTorch und Apache MXNet. Diese Frameworks lassen sich entweder in den Amazon SageMaker integrieren oder mittels des AWS Deep Learning AMIs (Amazon Machine Images) nutzen. Zusätzlich zu all den hier beschriebenen KI-Lösungen lassen sich auch bestehende AWS-Produkte, wie z.B. für ML optimierte Amazon EC2-Instanzen oder auch S3-Objektspeicher zur Speicherung der Daten, in das Gesamtsystem einbetten (Amazon Web Services, 2020c).

Anaconda

Anaconda ist eine Data-Science-Plattform, die das Entwickeln von ML-Algorithmen sowie ML-Modellen erleichtert. Die Plattform kann auf allen gängigen Betriebssystemen betrieben werden und unterstützt die Programmiersprachen Python und R

(Anaconda, 2020). Mit nur wenigen Klicks können mittels des Anaconda Navigators (grafische Desktop-Anwendung) über 1500 Python- und R-Pakete heruntergeladen und verwendet werden. Hierzu wird auf den hauseigenen Open-Source-Paketmanager Conda zurückgegriffen (Anaconda, 2017). ML-Modelle lassen sich unter anderem zum Beispiel mit sickit-learn und TensorFlow erstellen. Weiterhin stehen Analysetools wie Numba oder Dask zur Verfügung, um Laufzeiten zu optimieren. Zuletzt lassen sich die Ergebnisse dann mit Bibliotheken wie Matplotlib, bokeh, Datashader oder HoloViews visualisieren. Während Anaconda in der kostenfreien Version eine Desktop-Anwendung darstellt, kann in Anaconda Enterprise eine skalierbare, kollaborative Plattform für ML gefunden werden (Anaconda, 2020). Unter Verwendung des zuletzt genannten Tools kann ein ML-Modell kollaborativ entwickelt, verbessert und bereitgestellt werden. Zudem ist es möglich, das Repository von Anaconda in Form eines Mirrors zu betreiben, um ML-Modelle unter den Auflagen unternehmensweiter Richtlinien entwickeln zu können.

Analytic Solver

Das Tool Analytic Solver ging aus dem Tool XLMiner hervor und ist als Add-in für Excel ein leichtverständliches und weit verbreitetes ML-Tool. So kommt es bspw. bei der NASA, der US Army oder der FDA zum Einsatz (Frontline Systems, 2020a). Neben dem Angebot von ML-Algorithmen für Neuronale Netzwerke, Klassifikation, Regressionsbäumen, logistische Regression, Bayes-Klassifikator, K-nächste Nachbarn, Clusteransätzen und Hauptkomponentenanalysen (Frontline Systems, 2020a), finden sich hier auch klassische Berechnungsansätze für statistische Analysen, wie z.B. der t-Test, Kovarianz- und ANOVA-Berechnungen (Microsoft, 2020c). Als Versionen werden weiterhin Themen der Optimierung (Premium Solver), der Simulation (Risk Solver), Data Mining (XLMiner) und Cloud-basierte Berechnungen (AnalyticSolver.com) unterstützt sowie dedizierte Softwareversionen für Entwickler (Solver SDK und XLMiner SDK) bereitgestellt (Frontline Systems, 2020b).

Apache Flink

Basierend auf dem Projekt *Stratosphere* (Alexandrov et al., 2014) ist Apache Flink ein quelloffenes Stream-Processing-Framework mit dem Ziel, enorme Datenmengen sowohl in Datenströmen als auch in Form von Stapeln zu verarbeiten. Im Falle der Verarbeitung von Datenströmen werden eintreffende Daten direkt verarbeitet. Typische Beispiele für solche Datenströme sind Finanztransaktionen oder auch Sensordaten. Um bei den immensen Daten nicht einzubrechen, verfügt Apache Flink über diverse Cluster, wie etwa ein Kubernetes-Cluster oder ein eigenes Flink-Cluster zur Skalierung. Ein

weiterer nennenswerter Punkt ist die eingebaute Fehlertoleranz. Flink kann kontinuierlich Speicherpunkte (Checkpoints) erzeugen (Apache Software Foundation, 2020a). Diese sind leichtgewichtig und erlauben im Falle eines Ausfalls einer Komponente die Rekonstruktion des Datenstroms, indem zu einem vorher getroffenen Checkpoint zurückgerollt werden kann. Diese Eigenschaft qualifiziert Flink dazu, auch in einem Transaktions-kritischen Umfeld wie etwa dem Finanzsektor eingesetzt zu werden.

Apache Hadoop

Apache Hadoop ist eine der weitverbreitetsten Framework-Lösungen zur Verbreitung von BigData. Das Framework basiert auf Java und setzt auf ein eigenes Filesystem namens Hadoop Distributed File System (HDFS). HDFS ist ein fehler-tolerantes robustes und verteiltes Dateisystem, wodurch Hadoop über mehrere Cluster-Nodes skaliert werden kann (Apache Software Foundation, 2019b). Mittels der Hadoop-Implementierung des Map-Reduce-Algorithmuses von Google (Dean & Ghemawat, 2008) ist das Framework in der Lage große strukturierte und unstrukturierte Datenmengen (mehrere Terabyte) parallel zu verarbeiten (Apache Software Foundation, 2019c). Der Ressourcen-Manager YARN kümmert sich dabei um die Vergabe von Ressourcen, deren Limits pro Anwendung und Nutzer konfigurierbar sind. Auch die Anbindung von Objektspeichern ist möglich. Die Implementierung für Hadoop ist unter Ozone bekannt und erweitert Hadoop um einen skalierbaren, verteilten und fehler-toleranten Objektspeicher (Apache Software Foundation, 2018c). Alternativ lassen sich auch andere Objektspeicher einbinden. Das Hadoop-AWS Modul erlaubt beispielsweise die Integration von AWS S3. Cloudanbieter bieten Lösungen zum Betrieb von Apache Hadoop in ihrer Cloud an. So auch Microsofts Azure-Cloud, welche unter dem Namen „HDInsight“ eine Managed-Cluster-Lösung für Apache-Projekte des Hadoop-Ökosystems wie Hadoop, Spark, Kafka, Storm, etc. anbietet (Amazon Web Services, 2020a). Somit kann Apache Hadoop, neben anderen Apache Projekten, in der Cloud betrieben werden.

Apache Heron

Apache Heron ist Twitters Nachfolger zu Apache Storm. Einige operative Herausforderungen im Umgang mit Apache Storm (ehemals Twitter Storm) führten zu einer kompletten Neuentwicklung unter dem Namen Heron (Ramasamy, 2015). Zunächst startete das Projekt auch unter dem Namen Twitter Heron, worunter heutzutage noch einige Links zu finden sind. Anfang 2018 wurde auch das Heron-Projekt an die Apache Software Foundation übergeben (Masong, 2018). Das Framework ist ebenfalls ein verteiltes, fehlertolerantes skalierbares Framework zur Echtzeitverarbeitung von Datenströmen. Eine Migration von Storm auf Heron ist ebenfalls spielend leicht möglich, da die Heron API kompatibel mit Apache Storm ist.

Apache Kafka

Apache Kafka ist eine verteilte Streaming-Plattform, die es erlaubt, einen Stream an Daten anzubieten, zu abonnieren und zu verarbeiten (Apache Software Foundation, 2020b). Die Plattform wird vor allem für Echtzeitanwendungen eingesetzt, die Daten zwischen Systemen oder Anwendungen sicher übertragen und auf diese reagieren möchten. Durch die verteilte Architektur gehen eine hohe Fehlertoleranz und Skalierbarkeit einher. Diese Kerneigenschaften sind insbesondere für Big-Data-Systeme interessant, da das Kafka-Cluster zwischen Sendern und Empfängern vermittelt. Beim Eintreffen der Nachrichten des Senders können diese zunächst verarbeitet werden, z.B. durch eine Aggregation der Nachrichten, bevor sie dem Empfänger zugestellt werden. Eintreffende Informationen bezeichnet Kafka als „Topics“. Im Bereich des ML kann Apache Kafka beim Erfassen und Verarbeiten von Daten sowie beim Überwachen in Echtzeit unterstützen. Offiziell werden nur Scala und Java als Programmiersprachen unterstützt. Zahlreiche Open-Source-Projekte ermöglichen allerdings die Integration in andere Sprachen durch eigene Kafka-Clients (Apache Software Foundation, 2019a).

Apache Mahout

Apache Mahout ist ein Framework zur Entwicklung von verteilten Algorithmen der linearen Algebra. Darüber hinaus werden die Bereiche Vorverarbeitung, Regressionsanalyse, Clustering und Empfehlungssysteme bedient. Eine Stärke von Apache Mahout ist es, dass das Framework mit Big-Data-Systemen wie etwa Apache Spark zusammenarbeitet. Dabei werden die Daten mit parallelisierbaren Abfragen verarbeitet, gesteuert über Apache Mahout. Diese Eigenschaft macht das Framework insbesondere mit dem Umgang von großen Datenmengen interessant. Bei der Entwicklung mittels Apache Mahout wird auf eine eigene, auf der Programmiersprache SKALA basierende, Sprache zurückgegriffen (Apache Software Foundation, 2020c).

Apache MXnet

Apache MXnet ist eine Deep-Learning-Bibliothek für das Entwickeln und Trainieren von neuronalen Netzen. MXnet unterstützt die Gluon-Schnittstelle, einer in Zusammenarbeit von Amazon und Microsoft entwickelte Deep-Learning-Schnittstelle zur standardisierten Entwicklung von neuronalen Netzen. Ziel der Gluon-Schnittstelle ist es, schnellere und zugänglichere Entwicklung sowie standardisiertes Training zu erreichen. Die Stärken von MXNet liegen in einem breiten Anwendungsspektrum von Cloud bis hin zur Mobile-Plattform sowie in einer ausgeprägten Skalierbarkeit. Das Framework unterstützt die symbolische Programmierung sowie die imperative Programmierung. Somit wird eine Vielzahl an Programmiersprachen, darunter z.B. Py-

thon, Scala, C++, Java, R, Go, Javascript, unterstützt. Durch das breite Anwendungsspektrum wird das Framework auch von den großen Cloudanbietern wie Amazon AWS oder der Microsoft Azure Cloud unterstützt.

Apache Singa

Die Deep-Learning-Bibliothek Apache Singa ist spezialisiert auf verteilte und skalierbare maschinelle Lernanwendungen. Dabei ist der Software-Stack von Singa in die Architekturschichten Python-Interface, Backend und Hardware unterteilt. Das Python-Interface liefert abstrakte Komponenten zur Erstellung der ML-Modelle (Apache Software Foundation, 2020d). Die Backend-Komponenten bieten grundlegende Datenstrukturen sowie Hardware-Abstraktionen für die Planung und Ausführung an. Apache Singa ermöglicht demnach die Erstellung von ML-Modellen sowie das Trainieren über ein Cluster. Damit ist Apache Singa fähig über mehrere Grafikkarten zu skalieren, was insbesondere das Trainieren von komplexeren ML-Modellen ermöglicht.

Apache Spark

Die Apache Software Foundation definiert Apache Spark als universelle Analyse-Engine für die Verarbeitung von großen Datenmengen (Apache Software Foundation, 2018b). Apache Spark kann als ein Schlüsseltool von Big-Data-Verarbeitung eingestuft werden. Das Framework ermöglicht die analytische Auswertung, Transformation oder Fusion von Daten. Damit Spark ausreichend Rechenleistung für diese Aufgaben zur Verfügung hat, setzt es auf eine verteilte Architektur. So können unter der Verwendung des Cluster-Modus bei Bedarf weitere Instanzen bereitgestellt werden, um die Datenverarbeitung auch bei Datenmengen bis in den Petabyte-Bereich performant aufrecht erhalten zu können. So arbeitet jedoch nicht nur die Architektur verteilt, auch die Daten liegen verteilt vor. Dies ermöglicht ein paralleles Lesen sowie Arbeiten und sorgt für mehr Robustheit in der Datenverarbeitung. Neben dem Spark Core, welcher die grundlegende Operation wie Transformations-Funktionen übernimmt, bietet Spark weitere Module an: Spark Streaming ermöglicht die Verarbeitung von Datenströmen. GraphX ist eine API für die Berechnung von Graphen und mit SparkML können ML-Algorithmen eingesetzt werden.

Apache Spark MLlib

MLlib ist die hauseigene ML-Bibliothek von Apache Spark. Diese liefert Unterstützung etablierter ML-Algorithmen für dein Einsatz mit Apache Spark. Dabei werden unter anderem Lösungen für Aufgaben der Klassifikation, Regression, Implementierung von Entscheidungsbäumen sowie Empfehlungssysteme und Clustering-Aufgaben bereitgestellt (Apache Software Foundation, 2018a). MLlib verwendet die Spark API und kann

mit der populären NumPy-Bibliothek in Python sowie mit Bibliotheken in R zusammenarbeiten. Auch die Integration in Hadoop-Workflows ist durch die Unterstützung von zum Beispiel HDFS einfach möglich. Im Fokus der Entwicklung steht vor allem die Performance der Algorithmen und damit ein bestmögliches Zusammenspiel mit Spark.

Apache Storm

Apache Storm ist ein verteiltes, fehlertolerantes und skalierbares Framework zur Echtzeitverarbeitung von Datenmengen in einer Big-Data-Größenordnung. Ehe das Framework der Apache Software Foundation übergeben wurde, war es als Twitter Strom bekannt. Storm kann zeitgleich eine Datenbank in Echtzeit aktualisieren, während im Hintergrund weiterführende Berechnungen laufen (Continuous Computation). Darüber hinaus werden Remote Procedure Calls unterstützt. Laut Apache ist es mit Storm möglich, über eine Millionen Tupel pro Sekunde und Node zu verarbeiten.

BayesiaLab

BayesiaLab ist eine grafische Desktop-Anwendung für Windows, Mac und Linux. Wie der Name es schon vermuten lässt, liegen die Stärken der Anwendung in der Ausarbeitung von Bayesschen Netzen. Hierbei wird der Anwender von Modellerstellung, Analyse, Simulation bis hin zu Optimierung des Netzes unterstützt. Die Visualisierung der Bayesschen Netze ist somit ein Grundbaustein der Anwendung. Sie erlaubt einen visuellen Modellierungs- und Analyseprozess des erstellten Netzes. Mit dem Zusatz BayesiaLab WebSimulator können die erstellten Modelle ohne notwendige Zusatzsoftware bereitgestellt werden (Bayesia, 2020). Ist das Modell einmal erstellt, kann es von den zugangsberechtigten Nutzern ohne Einschränkungen verwendet werden. Mit dem VR-Modul von BayesiaLab ist sogar eine Visualisierung in der virtuellen Realität möglich, womit nahezu endlose Netzwerkgraphen dargestellt werden können.

BigML

Das Ziel von BigML ist der einfache Zugang zu maschinellem Lernen für Jedermann. Die Plattform stellt einige der bekanntesten und robusten ML-Algorithmen und Frameworks bereit. Dem Anwender stehen sowohl eine umfassende Weboberfläche sowie eine REST API zur Interaktion mit der Plattform zur Verfügung. Der Einstieg in die Plattform ist dabei komplett kostenlos, die Verarbeitung größerer Datenmengen setzt dann einen Bezahlaccount voraus. Die erstellten ML-Modelle können visualisiert und exportiert werden. So wird auch eine Offlinenutzung der Modelle möglich. Auch das Teilen der ML-Modelle über die Plattform ist unproblematisch. Über Team- und Projektstrukturen lassen sich die verwendeten Projektdaten ganz einfach mit anderen Mit-

gliedern teilen. Über diverse quelloffene Bindings und Bibliotheken kann über die unterschiedlichsten Programmiersprachen wie Python, JavaScript, Java, Conjure, Swift, C# und PHP auf alle BigML-Ressourcen zugegriffen werden. In Bezug auf die Automatisierung kann BigML ebenfalls punkten (BigML, 2020). Mit nur wenigen Handgriffen lässt sich das Training oder die Evaluierung von ML-Modellen automatisiert durchführen.

C4.5/C5.0/See5

Der C4.5 ist ein, von Ross Quinlan entwickelter, Algorithmus zur Generierung von Entscheidungsbäumen. Dabei können die generierten Bäume als Klassifizierer eingesetzt werden. Im Jahr 2008 wurde der Algorithmus zu den Top-10 Algorithmen im Bereich Data-Mining von der IEEE International Conference on Data Mining gezählt (Wu et al., 2008). C5.0 ist die Weiterentwicklung des Algorithmus. Diese äußert sich in genaueren, schnelleren und speicher-effizienteren Variante gegenüber der beliebten Vorgängerversion (Rulequest, 2017). Darüber hinaus wurde der Algorithmus auch um neue Datentypen erweitert. Eine große Neuerung ist auch der Support für die Boosting-Technik also dem Kombinieren mehrere Klassifizierer. Damit wird die Genauigkeit der Vorhersage nachhaltig verbessert.

Caffe

Das ursprünglich am Berkeley Artificial Intelligence Research Lab entwickelte Framework Caffe ist ein Deep-Learning-Framework, welches Algorithmen für die Klassifikation und Clusteranalyse von Bilddaten bereitstellt. Es wird als Community-Projekt auf Github weiterentwickelt. Eine Besonderheit an Caffe sind die individuellen Optimierungen für die im Projekt vorliegende Hardware. So liegt das Framework beispielsweise in einer optimierten Version für Intel-Xeon-Prozessoren vor. Durch die Unterstützung seitens NVIDIA mithilfe der cuDNN-Bibliothek ist es möglich, die Grafikprozessor-basierte Beschleunigung für die Verarbeitung der Bilddaten einzusetzen und damit über 60 Millionen Bildern pro Tag verarbeiten zu können (Berkeley Artificial Intelligence Research, 2020; Shelhamer, 2014). Während die ursprüngliche Version als Open-Source-Projekt kaum noch Aktivität zeigt, hat Facebook im April 2017 Caffe2 vorgestellt. Ende März 2018 wurde Caffe2 in PyTorch integriert und wird fortan innerhalb des Projekts weiterentwickelt. Aus diesem Grund ist Caffe nicht als eigenständiges Framework in den Übersichtsbildern aufgeführt (vergleiche Abbildung 5-1, rechte Seite).

Cloudera Data Science Workbench

Die Cloudera Data Science Workbench ist eine Plattform für kollaboratives Arbeiten zwischen Data Scientists und ML-Entwicklern. Ähnlich wie unter der Verwendung von Jupyter Notebooks liefert die Plattform eine browserbasierte Web-DIE (Cloudera Inc., 2020a). Direkt im Webbrowser können Bibliotheken und Frameworks für Python, R und Scala ausprobiert werden. Über den gestellten Cloudspeicher kann die Arbeit dann von überall aus aufgegriffen werden. Die Plattform ermöglicht sowohl das Entwickeln und Testen von ML-Modellen, als auch deren endgültige Bereitstellung z.B. mittels REST-APIs (Cloudera Inc., 2020b). Daten für die Erstellung der ML-Modelle können effizient aus bestehenden Hadoop- oder Spark-Systemen bezogen werden. Durch einen einheitlichen Ablauf sowie eine Protokollierung der Schritte wird eine einfache Reproduzierbarkeit sichergestellt.

CoNM

Das CoNM ist eine Plattform, die primär den Einsatz von KNN fokussiert und den Umgang mit KNN mittels einer graphischen Modellierungssprache steuerbar gestaltet. Während hierfür Programmierbibliotheken wie z.B. PyBrain oder TensorFlow zum Einsatz kommen, und die graphische Modellierung z.B. mittels Prozessmodellierungstools wie dem Modelangelo oder Draw.io bzw. Diagrams.net realisiert werden, erfolgt eine Interpretation der Modellierung mittels des CoNM, sodass das neuronale Verhalten transparent wird und ein direkter Eingriff in neuronale Vorgänge ermöglicht wird (Grum, 2020).

Databricks Unified Analytics Platform

Die Databricks Unified Analytics Plattform ist eine weitreichende Plattform zum Erstellen, Experimentieren, Testen, Trainieren und gemeinsamen Arbeiten an ML-Modellen. Mittels kollaborativer „Notebooks“ lassen sich ähnlich zu „Jupyter-Notebooks“ kollaborativ ML Modelle in Scala, R, Python und Java realisieren (Databricks, 2020a). Mit nur wenigen Klicks ist es möglich, ML-Ablaufumgebungen mit vordefinierten Vorlagen bereitzustellen. Eine aufwändige Konfiguration und die Integration der notwendigen ML-Frameworks entfallen somit gänzlich. Diverse Report und Share-Funktionalitäten ermöglichen das Teilen der Analysen, Ergebnisse und das Bereitstellen der fertigen ML-Modelle (Databricks, 2020b).

Dataiku

Dataiku ist eine kollaborative Data-Science-Plattform, welche sowohl auf eigener Hardware als auch in als Cloudlösung verfügbar ist. DSS ermöglicht es, unterschiedli-

che Datenquellen zu verbinden und mit diesen kollaborativ ML-Modelle zu entwickeln. ML-Bibliotheken wie Scikit-Learn, MLlib, etc. können für die Erstellung von ML-Modellen verwendet werden und mittels Python oder R kann der ML-Algorithmus individuell angepasst werden (Dataiku, 2020). Mit den vorhandenen Reporting-Möglichkeiten lassen sich fertige Modelle analysieren und Daten evaluieren. Individuell einrichtbare Dashboards ermöglichen das Teilen von Projektdaten mit anderen Nutzern.

Datameer

Datameer ist ein Produkt des gleichnamigen Unternehmens, zur Aufbereitung und Analyse von Datenmengen in Big-Data-Größenordnungen. Mit über 70 verschiedenen Quellen und unterschiedlichen Formaten können die Daten zusammengetragen werden. Hierbei ist es sowohl möglich, selbstständig Daten zu importieren, als auch Datameer damit zu beauftragen, Daten aus einer Quelle zu laden. Mithilfe von vordefinierten Funktionen können die Daten sowohl analysiert, als auch für zukünftige Verwendung vorbereitet werden. Unterstützend können die Daten auch visualisiert werden. Datensicherheit und Konsistenz adressiert Datameer auch. Es können Zugriffsrechte gesteuert, Daten verschlüsselt und Änderungen verfolgt werden (Datameer, 2020). Darüber hinaus erlaubt Datameer die Automatisierung von Datenfunktionen. So können bei neuen Daten ganz automatisch neue Reports generiert werden. Eine gestellte REST-API ermöglicht den Zugriff auf die Daten durch Drittsysteme.

DataRobot

DataRobot ist eine Auto-ML-Plattform, die dabei hilft, ein ideales prädiktives ML-Modell für vorliegende Daten zu finden. Das allgegenwärtige Ziel von DataRobot ist Auto-ML. Die Plattform automatisiert nicht nur die Auswahl von Modellen, sondern auch die Vorverarbeitung der Daten, das Erstellen und Finden von Features sowie das Verbessern des ML-Modells. Mit nur wenigen Klicks können hunderte unterschiedlicher ML-Algorithmen parallel durchlaufen werden. Features zur Auswertung erlauben das Finden des besten ML-Modells für die verwendeten Daten. Ist das passendste Modell gefunden, so kann es zudem über die Plattform bereitgestellt werden. Neben Auto-ML liefert DataRobot auch speziell die Möglichkeit automatisiert Zeitreihen zu verarbeiten und ML-Modelle in Produktion zu verwalten (DataRobot, 2020).

DeepLearning4J

DeepLearning4J ist ein Open-Source-Framework zur Erstellung verteilter Deep-Learning-Projekte in Java, Scala, Kotlin sowie Clojure. Während die JVM-basierten Sprachen zur grundlegenden Implementierung der Projekte dienen, sind die zugrundeliegenden Berechnungen mittels C, C++ und Cuda innerhalb des Frameworks realisiert.

Deeplearning4j kann laut den Entwicklern auch beim Einsatz von Multi-GPU-Systemen mit dem Framework Caffe in Bezug auf Performanz mithalten (DL4J, 2020). Das Framework ermöglicht den einfachen Zugang zu ML in der java-lastigen Enterprise-Umgebung. Um dennoch auf etablierte Standards im Bereich KI setzen zu können, können Python-Bibliotheken mithilfe von Keras angebunden werden. In Bezug auf mögliche neuronale Netze ist das Framework ebenfalls gut ausgestattet. So stehen beispielsweise CNNs, RNNs oder auch RBMs zur Auswahl. Enterprise-Anwendungen haben mit Deeplearning4J die Möglichkeit, um maschinelles Lernen erweitert werden zu können. Wie relevant dies ist, zeigt sich daran, dass Deeplearning4J laut Kdnuggets zu den Top 10 Deep-Learning-Frameworks auf Github gehört (Mayo, 2016).

DyNet

DyNet ist eine quelloffene und kostenfreie Bibliothek für das Realisieren von Neuronalen Netzen. Dieses kann wahlweise in C++ oder Python-basiert geschehen. Die Bibliothek selbst ist in C++ geschrieben und bietet sowohl CPU- als auch GPU-Unterstützung an. Gegenüber CNTK oder TensorFlow verwendet DyNet keine vorangestellte symbolische Beschreibung eines auszuführenden Modells. Stattdessen setzt DyNet auf eine dynamische Deklarationsstrategie, die es erlaubt für jede Trainingsinstanz eine unterschiedliche Netzwerkstruktur zu verwenden (Neubig et al., 2017). Dies erzeugt einen Overhead, der mit DyNet aber sehr gering ausfällt (Neubig et al., 2017). Am Beispiel der Sprach- bzw. Satzerkennung wird deutlich, aus welchen Gründen es sinnvoll sein kann, auf eine dynamische Deklarationsstrategie zu setzen. Typischerweise werden bei verschiedenen langen Sätzen die kürzeren Sätze mithilfe der Padding-Strategie aufgefüllt. Diese ergänzenden, semantisch unbedeutenden, Informationen sind notwendig, damit die Satzlänge überall gleich lang ist und somit das Netz auf diesen Lernen kann. Mithilfe einer dynamischen Deklarationsstrategie ist das Anwenden der Padding-Strategie nur noch je Batch notwendig.

ELKI Framework

Elki ist ein in Java geschriebenes quelloffenes Data-Mining-Framework. Die Schwerpunkte des Frameworks sind unüberwachte Lernmethoden, die Clusteranalyse sowie das Erkennen von Ausreißern (Elki-project, 2020). Dabei stellt ELKI eine Implementierung für eine Vielzahl an relevanten Algorithmen wie etwa K-Means oder Nächste-Nachbar-Klassifikatoren bereit. Diese sind so konzipiert, dass sie sich leicht erweitern lassen, sodass neue Varianten erstellt werden können.

Fastai

Das Ziel von Fastai steckt schon im Namen der beliebten Python-Bibliothek. Die auf Deep-Learning spezialisierte Bibliothek ermöglicht es, komplexe ML-Modelle mit wenigen Zeilen Quelltext zu erstellen. Dabei setzt Fastai auf homogene Schnittstellen, die in hierarchisch organisierten Ebenen lose gekoppelte abstrakte Deep-Learning-Muster vereinen (Howard & Gugger, 2020). Dies hat zur Folge, dass die Entwicklung eines ML-Modells für Computer Vision genauso funktioniert wie für ein NLP-Modell. Dabei fußt Fastai auf PyTorch und nimmt viel Boilerplate-Quelltext weg. Ein Port namens SwiftAI bringt die Vorteile von Fastai in die Programmiersprache Swift (fastai, 2020).

Gluon

Die Gluon-API-Spezifikation versteht sich als High-Level-API Aufsatz für Apache MXNet und ist durch eine Kooperation von Microsoft und Amazon entstanden. Ziel des Projekts ist es Deep Learning noch zugänglicher zu machen, indem eine imperative, leicht verständliche, API vor Apache MXNet gestellt wird. Die Idee ist damit vergleichbar mit Keras oder Fastai. Aus einer PR-Meldung Ende 2017 von Microsoft geht hervor, dass die Unterstützung von Gluon auch für CNTK angedacht ist (Microsoft, 2017a). Leider steht die Integration immer noch aus, wie ein eingestellter Issue im CNTK Github Repository zeigt. Daher ist es fraglich, ob und wann eine Integration erfolgt.

Gnu Octave

Gnu Octave ist eine kostenfreie und quelloffene Alternative zu Matlab. In erster Linie ist Gnu Octave eine Programmiersprache zur Lösung numerischer Probleme. Typische Anwendungsgebiete sind hierbei lineare- und nichtlineare Gleichungen oder auch statistische Auswertungen (Octave, 2020). Mittels einer mitgelieferten grafischen Entwicklungsumgebung kann die Programmiersprache ohne große Einstiegshürden verwendet werden. Auch im Bereich des ML kann Octave für erste prototypische Entwicklungen eingesetzt werden. Hier finden die Stärken von Octave wie etwa Vektor- und Matrixberechnungen ihren Einsatz. Über zusätzliche Pakete kann der Funktionsumfang von Gnu Octave erweitert werden. Lösungen sind größtenteils zu Matlab kompatibel (Eaton, 2020).

Google BigQuery

Googles BigQuery ist ein Managed-Datwarehouse-Angebot, welches eine voll-funktionsfähige Analyseplattform für enorm große Datenmengen bereitstellt. Dabei agiert die Plattform aus Sicht des Kunden serverlos und kann auch für große Projekte einfach skalieren. Basierend auf ANSI-SQL können Abfragen nicht nur schnell, sondern auch

ohne operativen Aufwand durchgeführt werden, selbst dann, wenn der Umfang der angefragten Daten bis in Größen von Petabyte reicht. Mithilfe von BigQuery ML können ML-Modelle mit SQL-Abfragen erstellt und ausgeführt werden. Ein entscheidender Vorteil ist es, dass die Daten zur Erstellung eines Modells nicht verschoben werden müssen, stattdessen kann das Modell direkt auf der Plattform entwickelt werden (Google Cloud, 2020b).

Google Cloud AutoML

Das Ziel hinter Googles Plattform: Google Cloud AutoML ist es, auch Unerfahrenen oder Entwicklern mit geringen ML-Kenntnissen zu ermöglichen, ML-Modelle zu trainieren. So gibt es eine grafische Weboberfläche zu Einrichtung und Training von ML-Modellen sowie automatisierte Abläufe zur Label-Erstellung von Trainingsdaten (Google Cloud, 2020a). Google Cloud AutoML liefert unter anderem Möglichkeiten zur Verarbeitung von natürlicher Sprache, der Erstellung von Übersetzungsmodellen, der Auswertung von Bilddaten sowie für den Einsatz von strukturierten Daten.

Google Colab

Google Colab(boratory) ist eine kostenfreie Cloud-Anwendung zum Ausführen von Jupyter-Notebooks im Browser (Google, 2020). Mit wenigen Klicks lässt sich Google Drive anbinden, um Daten mittels Python analysieren und auswerten zu können. Durch die Integration von gängigen ML-Bibliotheken wie etwa Keras, TensorFlow oder scikit-learn können ML-Modelle erstellt, trainiert sowie angewandt werden. Eigene oder nicht vorhandene Bibliotheken lassen sich ebenfalls über Google Drive einbinden. Ein wesentliches Feature ist die Möglichkeit des kollaborativen Arbeitens. So können gemeinsam Notebooks entwickelt oder abgelaufene Notebooks und deren Ergebnisse per Direktlink geteilt werden. Den Möglichkeiten sind kaum Grenzen gesetzt und mit 12GB Ram lassen sich auch größere Projekt gut meistern. Eine Colab-Pro-Version mit mehr Rechenleistung steht aktuell nur US-Kunden zur Verfügung. Es ist sicherlich aber nur noch eine Frage der Zeit bis auch Kunden aus anderen Ländern eine Pro-Version erwerben können.

Gurobi

Das gleichnamige Unternehmen veröffentlicht unter dem Namen Gurobi eine kostenpflichtige Solver Lösung, die unter anderem lineare- sowie quadratische Programmierung, gemischte-ganzzahlige Programmierung, quadratisch beschränkte Programmierung und gemischt-ganzzahlige quadratisch-beschränkte Programmierung ermöglicht (Gurobi, 2020b). Der Gurobi Optimizer ist dabei in C implementiert und bietet Schnittstellen zu nahezu allen gängigen Sprachen wie C++, Java, .NET und Python sowie eine

Integrationsmöglichkeit in MATLAB. Im Rahmen von Gurobi Cloud stellt das Unternehmen die Anwendung auch als Cloud-Lösung bereit (Gurobi, 2020a).

H2O.ai

H2O ist eine quelloffene, verteilte, in-memory ML-Plattform die ML-Ansätze zum überwachten und unüberwachten Lernen unterstützt. Eine Vielzahl von unterschiedlichen Algorithmen steht hierbei für die Programmiersprachen Python oder R zur Verfügung. Mithilfe von AutoML ist das automatisierte Erstellen, Trainieren und Auswerten für ein konfigurierbares Zeitlimit möglich. Durch die Möglichkeit einer verteilten Architektur können auch große ML-Herausforderungen gemeistert werden. Die H2O Compute Engine umfasst das Laden der Daten, die Analyse und damit verbundene Feature-Extraktion, das Trainieren von unüberwachten und überwachten ANNs sowie das anschließende Exportieren eines fertig trainierten Modells (H2O.ai, 2020). Als Datenquellen werden marktübliche Vertreter wie beispielsweise der Objektspeicher S3, das Dateisystem von Apache Hadoop HDFS sowie Netzwerkspeicher unterstützt. Durch den In-Memory-Ansatz können die Daten schnell verarbeitet werden.

IBM Cognos Analytics

Unter dem Namen IBM Cognos Analytics betreibt der Konzern IBM eine cloud-basierte Plattform für die Visualisierung, Analyse und Weitergabe von Daten (IBM, 2020a). So lassen sich aus Daten wertvolle Informationen durch die visuelle Darstellung sowie durch das Erstellen von Berichten extrahieren. Neben der Auswertung ist es zudem möglich, Daten zu bereinigen und diese schlussendlich mit anderen über die Plattform zu teilen. IBM Cognos Analytics erleichtert das Verständnis vorliegender Daten und ermöglicht die Datenvorverarbeitung für die weitere Verarbeitung durch beispielsweise ML-Modelle.

IBM DSX (bekannt als Watson Studio)

Das Produkt namens „IBM Data Science Experience (DSX) platform“, welches auch als „Watson Studio“ bekannt ist, ist eine kostenpflichtige Online-Plattform von IBM für die Analyse von Daten sowie das manuelle oder automatisierte Erstellen von ML-Modellen (IBM, 2020b). Durch die Integration von Jupyter können direkt im Browser interaktive Daten vorbereitet sowie ML-Modelle erstellt, trainiert und ausgeführt werden. Mithilfe des SPSS Modeler, welcher in das Watson Studio integriert werden kann, können analytische Aufgaben per Drag-and-Drop-Funktionalität durchgeführt werden (IBM, 2020c). Auch Auto-ML-Features sind Bestandteil von Watson Studio. Auch hier können fertige Bausteine per Drag-and-Drop-Funktionalität integriert werden. Eine visuelle Programmierung von ML-Modellen ist somit möglich.

IBM SPSS Modeler

Der IBM SPSS Modeler ist eine kommerzielle Data-Mining- und Text-Analyse-Software, die sowohl als Desktop-Anwendung als auch innerhalb von IBM DSX (Watson-Studio) angeboten wird (IBM, 2020c). Damit reiht sich der SPSS Modeler in die Produktfamilie von IBM ein. Die Idee hinter SPSS Modeler ist das Ermöglichen eines schnellen und zielorientierten Prozesses von den Daten angefangen bis hin zum fertigen ML-Modell. Dabei unterstützt die Anwendung viele unterschiedliche Datenquellen, ist in der Lage automatisch Daten aufzubereiten, diese automatisch zu visualisieren sowie ML-Modelle zu konstruieren und zu evaluieren. Es werden Entscheidungs-bäume, neuronale Netze sowie Regressions-Modelle unterstützt (IBM, 2020c). Darüber hinaus ist der Einsatz diverse Programmiersprachen wie R oder Python sowie Tools wie Apache Spark und Hadoop oder Jupyter möglich.

IBM SPSS Statistics

IBM SPSS Statistics ist ein kostenpflichtiges Programmpaket zur statistischen Auswertung von Daten, welches mit dem Watson Studio integriert werden kann. Mit diesem lassen sich gängige statistische Verfahren einsetzen (IBM, 2020d). Mithilfe von Zusatzmodulen sind weitere Funktionen wie etwa eine Zeitreihenanalyse oder der Einsatz von Entscheidungsbäumen abgedeckt. Des Weiteren wird ein Zusatzmodul für neuronale Netze angeboten.

Ilastik

Die modulare quelloffene Anwendung Ilastik ermöglicht eine einfache Klassifikation und Segmentierung von Bilddaten (Sommer, Straehle, Köthe & Hamprecht, 2011). Mithilfe einer benutzerfreundlichen UI schafft Ilastik ein einheitliches und einfaches Bedienkonzept für Konzepte wie der Pixel- und Objekt Klassifikation, dem Extrahieren von Features sowie der Verfolgung und dem Zählen von Objekten. Die Anwendung konnte bereits in etlichen wissenschaftlichen Veröffentlichungen Anwendung finden (Berg et al., 2019).

JMP

JMP ist eine Software zur statistischen Datenanalyse. Der Funktionsumfang umfasst hierbei unter anderem das Erfassen sowie Bereinigen von Daten sowie die Analyse und Visualisierung der Daten. Mithilfe statistischer Modelle können weitere Informationen aus den Daten gewonnen werden (SAS, 2020b). So lassen sich etwa Trends und Muster in den Daten erkennen und für weitere Analysen berücksichtigen. Unter dem Namen JMP Live stellte SAS im Jahr 2019 ein Produkt vor, welches es ermöglicht, Analyse von JMP direkt online zu teilen und somit kollaborativ auszuwerten (SAS, 2020a).

Keras

Keras ist eine der wohl bekanntesten Open-Source-Bibliotheken für die Erstellung von neuronalen Netzen und sieht sich selbst als eine API für neuronale Netze. Der Fokus der Bibliothek liegt klar auf einer nutzerfreundlichen Verwendungsmöglichkeit, sowie auf einem modularen und erweiterbaren Aufbau. Keras liefert alles, was ein neuronales Netz ausmacht: Implementierungen von Optimizern, Aktivierungs-Funktionen, Layern und weitere ML-typische Bausteine. Mit nur wenigen Codezeilen kann ein eigenes ANN erstellt werden. Der Erfolg von Keras ging auch an Google und damit an TensorFlow nicht vorbei. Bereits im Dezember 2018 verkündete man, dass fortan Keras als High-Level-API in TensorFlow 2.0 integriert wird (Gupta, Gordon & Allison, 2018). Seit der zweiten Version von TensorFlow ist Keras unter dem Paket „tf.keras“ integriert.

KNIME

KNIME Analytics Plattform ist eine quelloffene und kostenfreie Software zur Datenanalyse, welche Integrationsmöglichkeiten für maschinelles Lernen bietet. KNIME setzt dabei auf ein Workflow-System als Arbeitsweise (Knime, 2020). Über eine grafische Oberfläche können verschiedene Module zusammengebaut werden, welche sich um die Modellierung, Bereinigung, Analyse und Visualisierung von Daten kümmert. Sollten bestehende Module für einen Anwendungsfall nicht ausreichen, so lassen sich eigene Module in den Programmiersprachen R und Python entwickeln. Ein weiteres nennenswertes Feature ist das Konzept der Connector. Mit diesen ist es möglich, verschiedene Dienste, wie zum Beispiel Datawarehouses, Datenbanken, Objektspeicher uvm. zu verbinden (Knime AG, 2020). Connector sowie weitere Module können dann in einen Workflow gegossen werden. In der kostenfreien Version muss ein Prozess immer durch einen Klick angestoßen werden. Ist es das Ziel Aufgaben und Prozesse zu planen, so kann auf KNIME Server geupgradet werden. Instanzen hierfür können beispielsweise bei AWS bestellt werden (Amazon Web Services, 2020b).

Kubeflow

Bei Kubeflow handelt es sich um ein quelloffenes System, mit welchem ML-Workflows ausgeliefert werden können, die von TensorFlow als Container bereitgestellt werden. Der Entwicklungsprozess wird insbesondere durch Wiederverwendbarkeit und Kombinationsmöglichkeiten dieser ML-Workflows charakterisiert, sodass mit Kubeflow eine Lösung zur effizienten Erstellung und Deployment sowie Verwaltung gefunden werden kann (Lucks, 2020). Eine technische Fundierung erfahren die Workflows durch die Orchestrierung von Containern über Kubernetes (Joos, 2020). Ein Dashboard erleichtert das Management sowie den Überblick zu den Bereitstellungen.

Als Open-Source-Framework wird es insbesondere von Google und Microsoft unterstützt. Aufgrund der Integration von Komponenten wie Jupyter Notebooks, TFJob und PyTorch sowie der Bereitstellung mittels „kfctl“ ist eine breite Unterstützung des Projekts an Entwicklern sowie ML- und KI-Anwendungsentwicklern sichergestellt.

Kubeflow ist ein Open Source-Projekt zur schnellen und effektiven Bereitstellung von Containern über Kubernetes. Im Fokus stehen dabei Container in denen Workflows laufen, die für ML oder KI genutzt werden. An dem Projekt arbeiten unter anderem auch Google und Microsoft mit. Aktuell steht die Version 1.0 zur Verfügung, bei der nahezu alle Komponenten als stabil gelten.

Looker

Als ein direkter Konkurrent von Tableau (Conrad, 2020) adressiert Looker Unternehmen aller Größen und Industrien. Es integriert sich in die Google Cloud Plattform und weißt Ansätze für Datenvisualisierung, Analyse, OLAP, Dokumentenmanagement sowie zur Unterstützung von Entscheidungen auf. Weiterhin lassen sich weitere Systeme und Datenquellen integrieren sowie Big Data-Ansätze ansprechen. Mithilfe von Integrationen in Google BigQuery, Jupyter oder auch R-Studio lassen sich maschinelle Lernmodelle realisieren (Looker Data Sciences Inc., 2020).

Obwohl Looker eine deutlich geringere Verbreitung als die AWS oder Microsoft Azure Cloud hat, gilt es als die moderne Version dieser Cloudalternativen, da es von Beginn an für die Cloud konzipiert wurde und ein Standard in der Verwendung von Google Big Query gilt (Berlinger, 2019). Die Plattform wurde im Jahr 2019 ein Teil von Google und ist damit fester Bestandteil der Google Cloud (Alphabet, 2019).

Massive Online Analysis (MOA)

MOA präsentiert sich als das populärste Open-Source-Framework für Daten-Stream-Mining (MOA, 2020a), welches ML-Funktionen der Klassifikation, Regression, Clustering sowie Ausreißerdetektion und Empfehlungssysteme bereitstellt. Weiterhin werden zahlreiche Tools zur Evaluation von KI-Ansätzen gereicht. Obwohl sie Java-basiert ist, skaliert sie selbst bei anspruchsvollen algorithmischen Problemen (Bifet, Holmes, Kirkby & Pfahringer, 2010) und erfreut sich an einer wachsenden Community (MOA, 2020b). Da MOA sich darauf fokussiert, Datenströme in Echtzeit zu analysieren, hilft es Unternehmen, Wissen von der Analyse der aktuellen Situation abzuleiten, schnell auf auftretende Probleme zu reagieren, Trends zu entdecken und die Leistung zu verbessern (Bifet, Gavaldà, Holmes & Pfahringer, 2018).

Mathematica

Als Teil der Klasse der Mathematikprogramme ist Mathematica so konzipiert, dass Anwender die Algorithmen zur Lösung mathematischer Probleme zur exakten bzw. numerischen Lösung nicht kennen müssen. Neben vielen vordefinierten Berechnungsfunktionen stehen zusätzlich Erweiterungspakete zur Verfügung, welche den Einsatz in spezialisierten technischen, natur- und wirtschaftswissenschaftlichen Gebieten ermöglichen (Benker, 2016).

Da eine eigene Programmiersprache in Mathematica integriert ist, können Anwender eigene Programme erstellen und den Funktionsumfang des Tools leicht erweitern. Somit werden auf diversen Mathematik-Foren, Universitätshomepages oder durch Dritthersteller Plug-Ins für Spezialprobleme bereitgestellt. Diese erweitern den grundsätzlichen Funktionsumfang von Mathematica der Statistik, des ML, Datenanalyse und Visualisierungen zuverlässig.

MATLAB

MATLAB zeigt sich als ein Matrix-orientiertes Softwaresystem, dessen Name für Matrix Laboratory steht (Stein, 2001). Ursprünglich wurde es in den 70ern aus den Softwarepaketen Linpack und Eispack zur Lösung linearer Gleichungssysteme und algebraischer Eigenwertprobleme entwickelt. Inzwischen wurden durch die Beiträge vieler Nutzer und der Professionalisierung des Tools viele weitere Module zur Simulation, Datenauswertung und Visualisierung entwickelt. Somit dient es nicht nur an universitären Forschungseinrichtungen als zuverlässiges Werkzeug, sondern findet Einsatzbereiche in Industrie und Praxis. Es stellt einen Fokus in der hier gezeigten Toolsammlung dar, da es eine spezialisierte Komponente für Statistik und ML als eigene Toolbox bereitstellt.

Metaflow

Metaflow ist ein Data-Science-Framework, das von Netflix für die Unterstützung der eigenen Infrastrukturteams beim Verwalten von ML-Lösungen entwickelt wurde. Ende 2019 wurde aus dem Projekt ein Open-Source-Projekt. Metaflow verfolgt das Ziel, die Interessen von Data-Scientists und Infrastrukturteams gleichermaßen zu vertreten. Data-Scientists entwerfen ML-Modelle, die durch das Voranschreiten schnell komplex werden können. Dabei wird lokales Verwalten von Modellen und Daten unübersichtlich. Demnach ist es ein Wunsch, diese Aufgaben auf die Ebene der Server auszulagern, um zudem die Modelle allgemein zugänglich zu machen. Metaflow ermöglicht es, Modelle und Daten in Form von Containern auf AWS bereitzustellen, sodass die containerisierten Modelle von unterschiedlichen Businessanwendungen eingesetzt werden können. Neue Modelle sowie Änderungen an bestehenden Modellen können

fortan einfach mittels neuer Container oder neuen Versionen bestehender Container bereitgestellt werden. Anders als andere ML-Frameworks und Plattformen vertraut Netflix nicht auf eine GUI-Oberfläche (Metaflow, 2020c). Als Grund hierfür nennt das Unternehmen die schlechte Skalierbarkeit solcher Systeme. Um das Verwenden des Frameworks dennoch einfach zu ermöglichen, vertraut Metaflow auf die beliebten Programmiersprachen Python und R. Dabei kann auf eine Vielzahl an etablierten Bibliotheken wie PyTorch, TensorFlow, oder SciKit Learn zurückgegriffen werden (Metaflow, 2020a). Auch das Thema der Kollaboration wird von Metaflow aufgegriffen. So ist es möglich, bestehende Modelle und ML-Algorithmen anderer nicht nur zu nutzen, sondern auch auf diesen aufzubauen. Da die Infrastruktur von Netflix hauptsächlich auf AWS vertraut, ist der Einsatz von AWS ein elementarer Bestandteil des Frameworks. So werden beispielsweise AWS-Batch-Instanzen für das Verarbeiten der Daten sowie S3-Buckets für das Speichern von ML-Algorithmen und deren Ergebnisse gewählt. Ein Speichern von Snapshots geschieht dabei ganz automatisch, ohne Eingriff der Entwickler. Auch wenn Metaflow technisch gesehen von jedem Cloud-Provider betrieben werden könnte, ist die Anbindung an AWS ein zentraler Bestandteil. Daher wird aktuell nur AWS als Cloud-Provider fokussiert (Metaflow, 2020b).

Microsoft Azure Machine Learning / ML Studio

Azure Machine Learning ist Microsofts Cloud-ML-Plattform, mit welcher die Entwicklung und Bereitstellung von ML-Modellen von Anfang bis Ende durchgeführt werden kann. Die Entwicklung von ML-Modellen kann sowohl per Code als auch per Drag-and-Drop-Designer, sowie schlussendlich vollautomatisch in Form von Auto-ML erfolgen (Microsoft, 2020a). Von der Erstellung bis hin zu finaler Bereitstellung können ML-Pipelines etabliert werden, die sich um das stetige Bauen und Bereitstellen der Modelle kümmern. Bei der Entwicklung von ML-Modellen kann auf gängige Frameworks und Tools wie beispielweise TensorFlow, PyTorch, scikit-learn, uvm. zurückgegriffen werden. Über das Security-Center können Unternehmensrichtlinien etabliert werden, um so Ressourcen und Daten zu schützen oder freizugeben.

Microsoft Cognitive Toolkit

Hinter dem Namen Microsoft Cognitive Toolkit verbirgt sich ein frei verfügbares Deep-Learning-Framework zur Entwicklung von neuronalen Netzen. Es kann als direkter Konkurrent zu Googles TensorFlow gesehen werden. Als Programmiersprachen werden Python, C++ sowie C# unterstützt (Microsoft, 2017b). Es werden alle gängigen und bekannten Netzwerk-Arten unterstützt darunter beispielsweise CNNs und RNNs. Das Microsoft Cognitive Toolkit kann zudem Gebrauch von mehreren GPUs sowie Servern machen, um die Erstellung von ML-Modellen zu beschleunigen (Microsoft, 2017b).

Zum Austausch der ML-Modelle wird das ONNX-Format unterstützt. Mehr zu diesem Format findet sich in diesem Abschnitt unter ONNX.

Microsoft Machine Learning Server

Bei dem Tool "Microsoft Machine Learning Server" handelt es sich um eine fortschrittliche Softwaresammlung, welche es ermöglicht mit diversen Bibliotheken, Basisdistributionen und Deployment-Mechanismen, die die Erstellung von KI-basierten Anwendungen unterstützt. Zum einen werden attraktive KI-Algorithmen hierfür bereitgestellt. Beispiele belaufen sich auf R-basierte oder Python-basierte Distributionen. Zum anderen können erstellte Softwarekomponenten leicht als Web-Service und On-Premises- und Cloud-basierte Kanäle für Endkunden bereitgestellt werden (Microsoft, 2019b).

Da die "Microsoft Machine Learning Server"-Lösung nahtlos in bestehende IT-Hardware- und Anwendungslandschaften integriert werden kann, können KI-Algorithmen effizient an Datenquellen platziert und Lösungen schnell an Datensinken eingespielt werden (Microsoft, 2020b).

Microsoft Power BI

Microsoft Power BI ist eine Plattform zur Analyse und Visualisierung von Daten aus unterschiedlichen Datenquellen. Durch kollaboratives Arbeiten lassen sich eingehende Daten sowohl transformieren, visualisieren als auch aufbereiten. Mithilfe individueller Dashboards und Auswertungen können sie zudem präsentiert und ausgewertet werden. Durch die von Microsoft bereitgestellten Apps können die erstellten Auswertungen auf unterschiedlichen Endgeräten betrachtet werden.

MicroStrategy Analytics Platform

Bei „MicroStrategy Analytics Platform“ handelt es sich um eine Plattform zur Datenerfassung und -analyse. Es lassen sich auch hier Daten unterschiedlichen Quellen bündeln und Erkenntnisse direkt zum Ort des Geschehens einspielen. Die Software vom gleichnamigen Hersteller ist in Analytics, Mobile, Usher und Cloud unterteilt. Während „Analytics“ die Analyse der Daten unterstützt, können mit „Mobile“ die Analysen in mobile Anwendungen für die Plattformen iOS und Android integriert und mit „Cloud“ über On-Premises-Berechnungen zur Verfügung gestellt werden (MicroStrategy, 2020). In Form von Reports, Dashboards werden Kunden in die Lage versetzt adhoc Analysen zu erstellen und Ergebnisse zu teilen (Moraschi, 2013).

Minitab Salford Predictive Modeler

Der „Salford Predictive Modeler“ von Minitab ist eine Software die diverse ML-Ansätze zur Analyse und Vorhersage bereitstellt. Es werden Algorithmen für die Bereiche Klassifikation, Regression, Lebensdaueranalyse, Analyse fehlender Werte, Einteilung der Daten (Data Binning) sowie Clustering und Segmentierung bereitgestellt (Minitab, 2020). Ein Fokus liegt hier insbesondere auf der hochpräzisen und sehr schnellen Verarbeitung von Datenanfragen (Additive, 2020).

ML.NET

ML.NET bringt ML auf die .NET-Plattform. Dabei deckt ML.NET das Laden und Speichern von Daten, die Datenvorverarbeitung, das Training von ML-Modellen sowie die Modellverwendung ab. So lässt sich die Bibliothek für eine Vielzahl an unterschiedlichen Problemstellungen einsetzen. Darunter sind: Klassifizierung, Regression, Zeitreihen, Anomalieerkennung und vieles mehr (Microsoft, 2019a). Für eine Verwendung von ML.NET spricht zudem die Einbindung von Microsoft in hauseigenen Produkten wie Power BI, Windows Defender oder der Azure-Cloud-Plattform (.Net Foundation, 2020). ML.NET liefert ein hohes API-Level für .NET-Entwickler, während es gleichzeitig Anbindungen an die TensorFlow-.NET-Bibliothek besitzt (Cesar & Microsoft, 2019). Damit ist auch das Trainieren von Deep-Learning-Modellen möglich.

MLPack

MLPack ist eine ML-Bibliothek, die in Form von erweiterbaren C++ Klassen klassische ML-Algorithmen bereitstellt (Curtin et al., 2018). Dabei lassen sich die Algorithmen auch in Python und Julia sowie als eigenständige Kommandozeilenprogramme verwenden. Die Softwarebibliothek hat Diversität in den Algorithmen als auch deren schnelle Ausführungsgeschwindigkeit als Ziel. Es stehen bekannte Algorithmen wie Gaussian-Mixture-Modelle, Hidden-Markov-Modelle, K-Means-Clustering, Lineare Regression, Naive-Bayes-Klassifizierer Baum-basierende Nachbarsuche uvm. zur Verfügung.

QlikSense

QlikSense ist eine von QlikTech entwickelte Plattform zur Datenanalyse. Das Tool reiht sich damit in die Kategorie der BI-Plattformen unter Microsoft-Power-BI und Pentaho ein. Mittels Drag-and-Drop lassen sich Analyseprozesse gestalten, um große Datenmengen schnell und zielgerichtet auszuwerten. Mittels KI-Unterstützung trifft die Plattform Vorschläge und generiert automatische Prozesse zur besseren Analysemöglichkeit. Bestehende Analyse lassen sich interaktiv in Form von Dashboards darstellen und teilen. Als eine Besonderheit kann die Integration in Kommunikations-Diensten

wie Slack oder Skype gesehen werden. Über die dialogorientierte Analyse lassen sich somit schnelle interaktive Abfragen in QlikSense gestalten. Durch offenen Schnittstellen lässt sich QlikSense erweitern und in andere Projekte integrieren (QlikTech International AB, 2020). QlikSense kann On-Premises oder als Cloud-Lösung bereitgestellt werden.

ONNX

ONNX ist ein quelloffenes Format zum Austausch von ML- und Deep-Learning-Modellen. Die grundlegende Idee ist es, ein einheitliches Format für ML-Modelle zum Austausch unter unterschiedlichen Tools zu stellen. In diesem Vorhaben wird ONNX von Microsoft, Amazon, Facebook und weiteren Open-Source-Projekten unterstützt. Damit verstehen große Cloud-Provider das Format und ermöglichen somit das Trainieren und Evaluieren von Modellen auf Cloud-Instanzen. Die unterstützten Tools reichen dabei von TensorFlow und Keras über CNTK und MXNet bis hin zu Pytorch und Scikit Learn (Onnx, 2019b). Ende 2019 hat die Linux Foundation das Projekt übernommen (Boyd, 2019), wodurch das Format weiter an Bedeutung gewonnen hat. Mit ONNX.js existiert zudem eine JavaScript-Bibliothek, welche das Ausführen von Modellen clientseitig im Browser sowie mittels NodeJS auf seiten des Servers ermöglicht. Nennenswerte Partner sind neben Microsoft und Amazon auch NVIDIA, Intel, Alibaba und IBM (Onnx, 2019a).

Orange Data Mining

Bei Orange handelt es sich um eine Desktop-Anwendung für ML und Datenvisualisierung. Das besondere an der Datenvisualisierung ist die Möglichkeit interaktive Komponenten integrieren zu können (Orange, 2020b). Damit können statistische Verteilungen, Entscheidungsbäume und Ergebnisse einer Clustering besser visualisiert werden. Auch die Visualisierung von multidimensionalen Daten ist möglich. Die Visualisierung lässt sich mit visueller Programmierung - also dem Platzieren und Verbinden von Bausteinen - interaktiv gestalten. Auch zu Lehrzwecken eignet sich die Software indem komplexe Data-Science-Aspekte visualisiert werden können. Beispiele hierfür sind die Visualisierung von linearer Regression oder dem populären k-Means-Cluster-Algorithmus (Orange, 2020b). Neben der Möglichkeit Orange als eigene Anwendung zu nutzen, existiert auch ein Modul für die Integration in eigene Python-Projekte (Orange, 2020a).

Pentaho

Pentaho ist ein BI-Plattform ähnlich zu Microsoft-Power-BI. Dabei lassen sich Daten aus unterschiedlichen Quellen integrieren, analysieren und visualisieren. Mithilfe von

generierbaren Berichten lassen sich die Daten sowie Ergebnisse der Analysen aufbereiten. Unterstützte Datenquellen sind relationale Datenbanken, Apache Spark und Hadoop, Analytische Datenbanken, Business Anwendungen wie SAP, Google Analytics sowie auf Dateiebene XML, JSON, Excel und vieles mehr (Hitachi, 2019). Darüber hinaus verfügt Pentaho über einen Benutzer, um gezielt Berechtigungen für das Anschauen oder Modifizieren von Auswertungen zu vergeben. Dank eines containerisierten Ansatzes lässt sich Pentaho beliebig skalieren und erweitern.

PyCaret

PyCaret ist eine quelloffene „low-code“ ML-Bibliothek. Ziel des Tools ist es, die Zeit für das Schreiben von Quellcode zu minimieren, in dem eine abstrakte Schnittstelle für diverse ML-Bibliotheken wie scikit-learn, MS LightGBM, XGBoost, uvm. geschaffen wurde. Dabei wird der Ansatz von der Entwicklung bis zur tatsächlichen Bereitstellung verfolgt, weshalb sich PyCaret auch als Ende-zu-Ende Bibliothek bezeichnet. Dieses Ende-zu-Ende-Erlebnis wird in Form einer portablen Pipeline ermöglicht, die darüber hinaus auch skalierbar ist. Damit eignet sich PyCaret für Data Scientists, die wenig Programmiererfahrung haben, für die Entwicklung von schnellen Lösungen sowie zu Lehrzwecken im Bereich ML. Die Bibliothek unterteilt sich in Module des überwachten und unüberwachten Lernens und bietet eine Vielzahl an Beispielen (PyCaret, 2020).

PyTorch

PyTorch ist eine von Facebook initial entwickelte Programmierbibliothek, welche als Framework den Anforderungen eines wissenschaftlichen Informationsverarbeitens gerecht wird und eine große Sammlung von ML-Algorithmen bereitstellt. Diese können aufgrund der GPU-Unterstützung schnell prozessiert werden. Pakete, die von der Community von PyTorch bereitgestellt werden, fokussieren die Bereiche des ML, Computer Vision, Signalverarbeitung, parallele Verarbeitung, Bild-, Video, und Audioverarbeitung sowie das in Netzwerken organisierte Verarbeiten (Desale, 2016). Ursprünglich basiert PyTorch auf der in LUA geschriebenen Bibliothek namens Torch, adressiert aber insbesondere das komfortable Arbeiten mittels Python.

PyXLL

PyXLL ist ein kostenpflichtiges Excel-Add-In, das es ermöglicht, Python innerhalb von Excel zu verwenden (PyXLL, 2020). Anstelle eines VBA-Codes können so Datenauswertungen mittels Python implementiert werden. Daher kann PyXLL als eine Ablaufumgebung für Python betrachtet werden. Aus diesem Grund, kann auch jegliche

Python-Bibliothek verwendet werden, um beispielsweise Daten zu analysieren oder in mithilfe von ML-Modellen zu verarbeiten.

R Platform

R ist eine quelloffene Entwicklungsumgebung für statistische Auswertungen. Diese erfreut sich einer großen Bekanntheit und ist im Bereich der wissenschaftlichen Auswertungen ein Standard. Dabei wird eine Vielzahl an statistischen Ansätzen unterstützt. Beispiele lassen sich in klassischen statistischen Tests, linearer und nicht-linearer Modellierung, Zeitreihenanalyse, Klassifikation und Clustering finden (The R Foundation, 2020). Über eine eigene Programmiersprache R, die sich an der Programmiersprache S orientiert, lassen sich die Auswertungen implementieren (Teetor, 2011). Mithilfe von Schnittstellen können die Implementierungen in andere Softwareprojekte integriert, oder mittels eines graphischen UI wie durch Rattle Data Mining (Togaware, 2019) zugänglich gemacht werden. Weiterhin ist es möglich Grafiken zu spezifizieren, welche im Analyselauf automatisch erstellt werden und somit sowohl die Präsentation der Daten als auch den vorherigen Analyseprozess vereinfachen (Hilfiger, 2015). Über zusätzliche Pakete lässt sich die Umgebung erweitern, sodass auch spezifische Fragestellungen bearbeitet werden können. Beispiele belaufen sich auf das Paket namens Caret (Kuhn, 2019) oder Erweiterungen zur Zeitreihenanalyse (\$Cryer and Chan, 2008). Es unterstützt Datenvorverarbeitung, Feature-Selektion sowie das Training und das Verbessern des ML-Modells und somit alle Bereiche eines ML-Prozesses.

RapidMiner

RapidMiner ist eine Data-Mining-Plattform, die die visuelle Erstellung von KI-Workflows ermöglicht. Dabei stellt RapidMiner Vorlagen für z.B. Predictive Maintenance, Fraud Detection (Rapidminer, 2020). bereit. Weiterhin kann mit RapidMiner Auto Model automatisiert in wenigen Handgriffen ein fertiges ML-Modell erstellt werden. Dabei werden sowohl überwachte- als auch unüberwachte Ansätze unterstützt. Neben dem Training können die resultierenden Modelle auch direkt in RapidMiner validiert werden. Jeder Workflow lässt sich zudem visuell darstellen. Die resultierende Pipeline geht dabei von Daten-Vorverarbeitung über Training bis hin zur Evaluation des Modells. RapidMiner kann mithilfe von R oder Python sowie deren Bibliotheken erweitert werden. Darüber hinaus können die erstellten Workflows mittels der zugehörigen Python-Bibliothek in bestehende Python-Projekte integriert werden (Rapidminer, 2020).

Rattle Data Mining

Rattle ist eine quelloffene und kostenfreie Desktop-GUI-Anwendung für Data-Mining-Analysen. Dabei arbeitet die Anwendung auf Basis der Programmiersprache R und ermöglicht neben der Analyse und der Transformation von Daten auch die Erstellung von überwachten und unüberwachten ML-Modellen (Williams, 2011b). Die Operationen die Rattle vollführt basieren auf R-Paketen und werden in einem R-Skript konsolidiert (Williams, 2011a). Dies erlaubt das Migrieren des Skripts auf die R Plattform. Insbesondere eignet sich die Anwendung daher auch für Einsteiger im Bereich Data Science. Erstellte ML-Modelle können dann auf Codeebene weiter verbessert werden.

SAS Base

Statistical Analysis System (SAS) Base ist eine Programmiersprachensoftware, die eine web-basierte Programmierschnittstelle, einsatzbereite Programme für Datenmanipulation, Informationsspeicherung und -abruf, beschreibende Statistiken und Berichterstattung, ein zentrales Metadaten-Repository und eine Makrofunktion bietet, die Programmierzeit und Wartungsaufwand reduziert (G2, 2020). SAS Base bietet eine skalierbare, integrierte Softwareumgebung, die für Datenzugriff, Transformation und Berichterstellung konzipiert ist. Diese Technologie umfasst eine Programmiersprache, Programme zur Datenverarbeitung, Informationsspeicherung und -abfrage, deskriptive Statistiken, Berichterstellung und eine Makrofunktion. SAS Base ist eine Datenverarbeitungsmaschine, die eine Vielzahl von Datenquellen lesen und schreiben kann. SAS Base speichert keine Daten innerhalb des SAS-Systems. Der Benutzer hat die vollständige Kontrolle darüber, wo und wie Daten gespeichert werden (z.B. lokal oder auf einem Datenbankserver). Der Nutzer kann auch die Art der Daten wählen, die abgerufen, analysiert und gespeichert werden sollen. SAS Base ermöglicht den Zugriff auf SAS Cloud Analytics Services (CAS) und SAS Viya (eine Cloud-fähige, speicherinterne Analysemaschine) (U.S. Department of Veterans Affairs, 2020).

SAS Enterprise Miner

SAS Enterprise Miner ist eine Lösung zur Erstellung präziser prädiktiver und beschreibender Modelle für große Datenmengen aus verschiedenen Quellen im Unternehmen. SAS Enterprise Miner bietet viele Features und viel Funktionalität für die Geschäftsanalysen zur Modellierung ihrer Daten. Einige der Geschäftsanwendungen dienen dazu, Betrug zu erkennen, Risiken und Ressourcenanforderungen zu minimieren, Ausfallzeiten von Anlagen zu reduzieren, Kampagnen durchzuführen und die Kundenabwanderung zu verringern. Es handelt sich um eine Predictive Analytics-Software, die eine benutzerfreundliche GUI und Stapelverarbeitung bietet. Weitere Anwendungsmöglichkeiten sind: Datenvorbereitung, -zusammenfassung und -exploration, prädiktive

und deskriptive Modellierung, Open-Source-Integration mit R, Modellgeneration; Modellvergleiche, Berichterstellung und -verwaltung, automatisierte Bewertung, SAS Viya-Aktionen innerhalb eines Prozessablaufs aufzurufen, skalierbare Verarbeitung, Cloud-Bereitstellungsoption (Predictive Analytics Today, 2020).

SAP HANA

SAP HANA ist eine hochleistungsfähige In-Memory-Datenbank, die datengesteuerte Echtzeit-Entscheidungen und -Aktionen beschleunigt und alle Arbeitslasten mit umfassenden Analysen auf Daten aus mehreren Modellen, sowohl vor Ort als auch in der Cloud, unterstützt (SAP, 2020). Es handelt sich um ein spaltenorientiertes, relationales Datenbank-Managementsystem, das von SAP entwickelt und vermarktet wird. Die Hauptfunktion als Datenbankserver besteht darin, Daten zu speichern und abzurufen, wie sie von den Anwendungen angefordert werden. Darüber hinaus führt es erweiterte Analysen durch (prädiktive Analysen, räumliche Datenverarbeitung, Textanalyse, Textsuche, Streaming-Analyse, Graphdatenverarbeitung) und umfasst Extraktions-, Transformations- und Ladefunktionen (ETL) sowie einen Anwendungsserver.

SAP HANA ist im Kern eine Datenbank, kann jedoch auch als Entwicklungsplattform genutzt werden. Es ist eine Technologielandschaft und fungiert als Befähiger für geschäftliche und organisatorische Veränderungen. SAP HANA hat sich seit seiner ersten Einführung drastisch weiterentwickelt. Heute ist es der Eckpfeiler der Technologie- und Entwicklungsplattform der SAP für die Zukunft. SAP erwirbt neue Werkzeuge und entwickelt bestehende Werkzeuge weiter, wobei SAP HANA die zugrundeliegende Plattform darstellt. Es werden neue Werkzeuge und Funktionen aufgebaut, die für die Zusammenarbeit mit SAP HANA optimiert sind. Ähnlich wie der SAP Solution Manager wird auch SAP HANA immer mehr zu einem integrierten Bestandteil der SAP-Gesamtfunktionalität. Vor diesem Hintergrund positioniert SAP HANA weiterhin als Basistechnologie zur Unterstützung hochoptimierter Geschäftsprozesse und Analysen. Fluggesellschaften und Spediteure nutzen SAP HANA-fähige Lösungen zur Routenoptimierung und Rentabilität, Einzelhändler nutzen SAP HANA-fähige Lösungen zur Erkennung von Betrug und Preisabsprachen mit ihren Lieferanten, und das alles in Echtzeit (Berg & Penny, 2013).

SAP Predictive Analytics

Mit SAP Predictive Analytics können Sie produktiver arbeiten und genaue Ergebnisse erzielen, indem Sie den gesamten Prozess der prädiktiven Modellierung automatisieren. Die Datenvorbereitungsfunktionen ermöglichen es Ihnen, Tausende von abgeleiteten Variablen aus einer Vielzahl von Datenquellen schneller und ohne eine einzige Zeile Code zu schreiben, zu erstellen. Auf diese Weise stellt sie wiederverwendbare

prädiktive Datensätze bereit, die von Mitarbeitern auf allen Ebenen verwendet werden können. SAP Predictive Analytics bietet einen Self-Service mit einem geführten Workflow, um durch Automatisierung Modelle in wenigen Minuten zu erstellen (Opalwave, 2020).

SAP Analytics-Lösungen haben Unternehmen dabei geholfen, die Datenanalyse zu verbessern, damit sie besser informiert sind, bessere Entscheidungen treffen und schneller auf Chancen und Probleme reagieren können (TEI-Analysen, Total Economic Impact). SAP Predictive Analytics unterstützt Organisation bei der Reduzierung von Sicherheitsproblemen der Mitarbeiter. Die Analyse und Überprüfung früherer Aufgaben- und Servicedaten führt in Unternehmen zu einer erheblichen Standardisierung der Prozesse (Forrester Research, 2016).

SAS Viya

SAS Viya ist eine Plattform, die es Unternehmen ermöglicht, schnell unterschiedliche Arten von analytischen Herausforderungen zu bewältigen, problemlos Lösungen mit eingebetteten Analysediensten zusammenzustellen und zu teilen. Weiter lassen sich regelkonforme Anwendungen erstellen, die schnelle Antworten und Ergebnisse liefern (SAS, 2016).

scikit-learn

Scikit-learn ist eine quelloffene und freie Python-Bibliothek, die viele Implementierungen für gängige ML-Ansätze im Bereich der Klassifikation, der Regression oder auch dem Clustering bereitstellt. Dabei arbeitet die Bibliothek eng mit den Python-Bibliotheken Matplotlib und Numpy zusammen. Dies ermöglicht beispielsweise ein direktes Erstellen von Plot-Grafiken aus der Bibliothek heraus. Scikit-learn verfolgt zudem das Ziel, die Benutzbarkeit durch ein hohes generalisiertes Sprachlevel zu erhöhen (Pedregosa et al., 2011). In Verbindung mit einer ausgereiften Dokumentation lassen sich so Ansätze von scikit-learn einfach in eigene Lösungen integrieren. Dies macht die Bibliothek gerade für Einsteiger im ML-Bereich attraktiv.

Shogun

Shogun ist eine quelloffene, in C++ implementierte, Bibliothek, welche weitreichende ML-Algorithmen bereitstellt. Das Projekt wurde bereits 1990 gestartet. Gerade weil die Bibliothek in C++ umgesetzt ist, profitieren diese Algorithmen einer schnellen Ausführungsgeschwindigkeit. Durch eine definierte Schnittstelle kann Shogun unter anderem in den Sprachen Python, Octave, R, Java, Lua, C#, Ruby sowie in MATLAB verwendet werden. Shogun unterstützt unter anderem den Einsatz von Support-Vector-Maschinen und Hidden-Markov-Modellen (Sonnenburg et al., 2010).

Sisense

Sisense ist, ähnlich wie Microsoft Power BI, eine BI-Lösung und bezeichnet sich selbst als Ende-zu-Ende-BI-Lösung. Darunter versteht das gleichnamige Unternehmen das Zusammenwirken zwischen Data-Scientist, Entwicklern und Analysten (Sisense, 2020a). Durch den Einsatz der Cloud lassen sich bereitgestellte Applikationen skalieren. Die Plattform ermöglicht die Integration diverser Datenquellen. Schnell lassen sich Dashboards mit Drag-and-Drop zusammenbauen. Durch die Integration von Sisense Blox lassen sich interaktive Widgets in Dashboards einbauen (Sisense, 2020b). Mithilfe von Sisense Hunch können große Datenmengen automatisiert in Form eines Deep-Neuronalen-Netzes abgefragt werden (Sisense, 2020c). Damit verspricht das Unternehmen eine interaktive Daten-Exploration von Big-Data durch eine geringe Antwortzeit zu ermöglichen (Sisense, 2020c).

SPMF

SPMF ist eine auf Pattern Mining spezialisierte quelloffene Data-Mining-Bibliothek für Java. Es werden insgesamt 196 Data-Mining-Algorithmen in einer Implementation bereitgestellt. Darunter befinden sich Algorithmen für Assoziationsanalyse, Itemset-Mining, Time-Series-Mining sowie Cluster- und Klassifikations-Implementierungen. Neben der Möglichkeit SPMF als Bibliothek in eigene Java-Projekte zu integrieren, bietet SPMF auch ein einfaches User-Interface sowie eine Nutzung per Kommandozeile an (Fournier-Viger, 2020).

Sqream

Sqream ist ein Datenbank-Management-System, das insbesondere für die Verwaltung von Big Data-Daten geeignet ist. Dabei schafft Sqream eine hohe Abfrageperformance durch die Unterstützung von Nvidia CUDA (Sqream, 2020). Gerade bei der Analyse von Daten in Big Data Größenordnungen kann Sqream demnach hiervon profitieren. Sqream kann dabei auf eigener Hardware als auch in der Cloud betrieben werden. Unter den Cloud-Anbietern, die Sqream anbieten, befinden sich Amazon AWS, Microsoft Azure sowie die Alibaba Cloud.

Stata

Bei Stata handelt es sich um ein Tool zur statistischen Analyse von einfachen wie aber auch komplexen Datensätzen (Stata, 2020). Da für Stata eine eigene Syntax zur Verfügung gestellt wird, können sämtliche Befehle sowohl über Menü-Einträge aber auch über eine Kommandozeile aufgerufen werden, sodass komplexe Analyseabläufe auch in DO-Dateien zusammengefasst werden können. Im Unterschied zum Tool SPSS fokussiert Stata jedoch auf das Erzeugen publikationsreifer Grafiken.

Tableau

Die Desktop-Version von Tableau ist eine traditionelle BI-Plattform, die das Aufbereiten, Analysieren und Zusammenfassen von großen Datenmengen unterschiedlicher Herkunft ermöglicht. Unterstützte Datenquellen sind unter anderem relationale Datenbanken, Big-Data-Lösungen sowie Google Analytics und Salesforce (Tableau Software LLC, 2020a). Mithilfe von Drag-and-Drop lassen sich auf einfache Weise interaktive Dashboards aufbauen (Tableau Software LLC, 2020a). Durch Tableau Server ist eine Bereitstellung der aufbereiteten Daten unternehmensweit möglich (Tableau Software LLC, 2020b). Ergänzend kann Tableau Mobile für den mobilen Zugriff genutzt werden (Tableau Software LLC, 2020c). Tableau Prep ermöglicht Vorverarbeitung und Aufbereitung von großen Datenmengen (Tableau Software LLC, 2020d). Alle Tableau-Bausteine lassen sich miteinander kombinieren und schaffen damit ein Ökosystem zum Verarbeiten, Analysieren und Visualisieren von Daten.

TensorFlow

TensorFlow ist eine von Google bereit gestellte Open-Source-Software-Plattform, welche als eine auf C++-basierten Programmierbibliothek die Datenfluss-orientierte Erstellung von Neuronalen Netzwerken ermöglicht. Als ein sehr weit verbreitetes Tool erfreut es sich insbesondere durch große Cloudanbieter wie Google, Azure und AWS einer großen Verbreitung. Insbesondere die Verfügbarkeit von grafischen Interfaces erleichtert das Hantieren mit komplexen Netzwerken und deren technische Realisierung: Entsprechend des Namens von TensorFlow operiert die Bibliothek auf Basis sogenannter Tensoren, welche eine relevante Dateneinheit repräsentiert. Weiter können Experimentläufe mittels des Browser-basierten TensorBords erstellt und gesteuert werden. Keras stellt hier eine sinnvolle Erweiterung dar und insbesondere das Arbeiten mit tiefen Netzwerken wird somit erleichtert (Desale, 2016; Koshy, 2017).

Teradata

Teradata ist in erster Linie ein relationales Datenbank-Management-System, das auf die Verarbeitung großer Datenmengen im Big Data-Bereich spezialisiert ist. Es lassen sich groß-skalierte Data-Warehouse-Anwendungen realisieren, indem die Daten parallel verarbeitet werden können. Mit *Teradata Vantage* wird das erprobte DBMS um eine analytische Komponente erweitert, sodass es als vollwertige Analytics-Plattform betrachtet werden kann (Teradata, 2020a). Hierbei kann auf *Teradata NewSQL* zurückgegriffen werden, welches die Erweiterungen des traditionellen SQLs für die Erstellung analytischer Auswertungen unterstützt. Die in Vantage eingegliederte ML-Engine ermöglicht es, analytische Funktionen auf dem gesamten Datensatz durchführen zu kön-

nen. Die Daten können hierbei in strukturierter, unstrukturierter oder multistrukturierter Weise vorliegen: Es können beispielsweise Text-, numerische, tabellarische, Dateitypen, CLOB- und BLOB-Datentypen in einem Ansatz analysiert werden (Teradata, 2020b).

TFlearn

TFlearn ist eine Deep-Learning-Programmiersbibliothek, die eine graphische Benutzeroberfläche bietet und das Erstellen von Experimenten erleichtert sowie beschleunigen soll, jedoch auf die technische Fundierung der Programmiersbibliothek von TensorFlow zurückgreift (TFlearn, 2020). Da sie als Aufsatz für TensorFlow dient und TFlearn vorrangig als High-Level-API zu interpretieren ist, ist TFlearn in den Übersichtsabbildungen nicht als eigenständiges Tool repräsentiert, sondern bei TensorFlow zuzuordnen (siehe Abbildung 5-1, linke Seite).

Theano

Theano ist eine Python-Programmiersbibliothek, welche mittels multidimensionalen Arrays die schnelle numerische Berechnung unterstützt und sowohl mittels der CPU, als auch der GPU realisiert werden kann. Mathematische Ausdrücke können somit in eine effiziente C++-Beschreibung oder für CUDA-freundliche Umgebungen von Theano kodiert und schnell berechnet werden (Brownlee, 2016a; Desale, 2016).

Tibco Data Science / Spotfire

Die Tibco-Produkte Data Science und Spotfire werden inzwischen von einer Reihe weiterer Produkte ergänzt, sodass sie insgesamt das Feld des Arbeitens mit ML-Techniken bedienen können (Tibco, 2020b). Während Tibco Data Science das kollaborative Erstellen von ML-Modellen im gesamten Unternehmen unterstützt, und bspw. das gemeinsame Monitoren, Verwalten und sicherheitsbewusste Erstellen von Datenpipelines und Modellen ermöglicht (Tibco, 2020a), ermöglicht Tibco Spotfire als umfassende Analytics-Lösung das Durchführen von simplen und KI-basierten Suchanfragen sowie das Visualisieren von Analyseergebnissen (Tibco, 2020c). Weitere Tibco-Produkte bereichern diese Analyseaktivitäten mit Cloud-basierten Lösungen, komfortablen Schnittstellen wie SaaS oder erweiterten Visualisierungsmöglichkeiten.

Tribuo

Tribuo ist eine von Oracle entwickelte ML-Bibliothek für Java. Ein erster öffentlicher Release der Bibliothek fand im August 2020 auf Github statt (Pocock, 2020). Damit ist Tribuo gegenüber DL4J noch jung auf dem Markt. Gerade weil aber Oracle hinter dem Projekt steht, sollte hier in Zukunft noch viel Weiterentwicklung zu sehen sein. Neben

klassische Implementierung maschineller Lernalgorithmen im Bereich der Klassifikation, der Regression und des Clustering, stellt die Bibliothek auch Schnittstellen zu anderen ML-Bibliotheken bereit. So können in Python trainierte ML-Modelle beispielsweise über die ONNX-Schnittstelle oder die TensorFlow-Schnittstelle angesteuert werden (Tribuo, 2020).

Turi Platform

Die Turi Platform ist eine von Apple als Open-Source bereitgestellte Plattform, welches es Anwendern ermöglicht, einfach und schnell ML-Techniken in App-Entwicklungen zu integrieren (Woodie, 2017 #358}. Hierzu zählen Techniken zur Generierung von Empfehlungen, Objekterkennungen, Bildklassifikation, Bildähnlichkeitserkennung sowie Aktivitätsklassifikation. Hierbei stellt es insbesondere den Anspruch, flexibel mit diversen Datentypen umgehen zu können und den Fluss von Daten mittels einer geeigneten Visualisierung analysieren zu können. Im Fokus steht jedoch das effiziente Einspielen von Trainingsmodellen auf Apple's Kerngeräten im MacOS, iOS, watchOS sowie tvOS (Apple, 2020b).

Vowpal Wabbit

Vowal Wabbit stellt ein ML-System dar, welches initial von Yahoo! Research und später von Microsoft Research unterstützt wurde und an sich selbst den Anspruch einer schnellen Durchführung von Lernansätzen stellt (Microsoft Corp., 2014). Es fokussiert das Erproben neuer ML-Ansätze, sodass Techniken des online, hashing, allreduce, reductions, learning2search, active sowie interactive learning hier bereitgestellt werden. Damit stellt es insbesondere ein Vehikel für eine fortgeschrittene Forschung dar und ist in zahlreichen Veröffentlichungen zur Anwendung gekommen (Microsoft Corp., 2014).

Weka Machine Learning Workbench

Bei Weka handelt es sich um eine Sammlung von ML-Algorithmen für Data Mining (Holmes, Donkin & Witten, 1994), sodass z.B. folgende Bereiche bedient werden können: Datenvorverarbeitung, Klassifikation, Regression, Clustering, Assoziations-basiertes Lernen, Attributauswahl und Filtern (Eibe et al., 2018). Diese können entweder über das Command Line Interface (CLI), eine Java-API oder diverse graphische Interfaces bedient werden (Brownlee, 2016b). So können mit dem Weka Explorer Daten für diverse Algorithmen transformiert und unterschiedliche Ansätze testweise erprobt werden. Der Weka Experimenter unterstützt das Erstellen von Experimenten auf Basis gewählter Algorithmen und Datensätze sowie der Analyse von Experimentläufen. Weka

Knowledge Flow unterstützt das graphische Erstellen von Transformationsschritten, Algorithmenauswahl und Experimentdurchführungen und dessen Analysen.

Da die Workbench von Weka diverse GUI-Interfaces, wie z.B. den Weka Explorer oder die Weka Experimentierumgebung, als Interface bereitstellt, stellt die Weka Machine Learning Workbench im Zusammenspiel der einzelnen Tools den mächtigsten Funktionsumfang bereit.

WSO2 Stream Processor

Als Open-Source-Tool unter der „CC BY 4.0“-Lizenz unterstützt der WSO2 Stream Processor das Empfangen und Analysieren von Datenströmen in Echtzeit. Beispiele werden hier unter der Apache-2-Lizenz bereitgestellt. Auf Basis von SQL-basierten Abfragen können so zum Beispiel Sensordaten, Geräteinformationen oder Applikationsdaten auf die Plattform übertragen und Analyseergebnisse als Alarmer, Handlungsaufforderungen, Dashboardvisualisierungen oder API-Interfaces in den jeweiligen Anwendungskontext übertragen werden (WSO2 Inc., 2020b). Im Vergleich zu anderen Plattformen differenziert sich der WSO2 Stream Prozessor dadurch, dass 100.000 Events je Sekunde mit lediglich zwei Servern realisieren werden können (WSO2 Inc., 2020a).

5.4 Heuristische Tools im Detail

Cross-domain hyper-heuristic Frameworks

Bei dieser Ansammlung an Frameworks handelt es sich um in Java implementierte Bibliotheken, welche es Anwendungsentwicklern vereinfachen sollen, Lösungsansätze in einer Hyperebene zu formulieren. Somit wird eine große Bandbreite an Tools bedient, welche von einer lokalen Suche abstrahieren und die Hypersuche leicht zugänglich machen (Ryser-Welch & Miller, 2014a). HyFlex kann als Beispiel-Framework genannt werden (Lehrbaum & Musliu, 2012), welches sich neben weiteren im Benchmark bewähren musste (Ryser-Welch & Miller, 2014b).

HeuristicLab Framework

Es handelt sich um eine Softwareumgebung, welche eine grafische Umgebung bereitstellt, um auch Nutzern ohne Programmierkenntnissen einen Zugang zu heuristischen und evolutionären Algorithmen zu ermöglichen (HeuristicLab, 2018). Aufgrund einer Plug-In-basierten Architektur können Softwarenutzer individuell Erweiterungen ergänzen, ohne den gesamten Quellcode kennen zu müssen. Zu den Stärken gehören insbesondere die Verfügbarkeit zahlreicher genetischer Algorithmen (GP, Tabu-Suche, Simulated Annealing, Hill Climbig), die Verfügbarkeit einer GUI, die Erweiterbarkeit

durch Plug-Ins, die Unterstützung im Experimentdesign und die Realisierung von parallelen Verarbeitungen. Da HeuristicLab in Microsoft C# geschrieben ist, erfolgt die beste Einbindung in Windows-Betriebssystemen sowie Microsofts .NET-Framework (Elyasaf & Sipper, 2014).

hMod

Das Tool hMod ist eine Java-basierte Bibliothek, welche den Anwendungsentwickler leitet, um Hyperebenen mittels zwei separater XML-Dateien zu definieren. Hierbei repräsentiert die erste Datei den heuristischen Auswahlprozess und die zweite Datei den Akzeptanzlauf (Ryser-Welch & Miller, 2014a). Hilfsfunktionen fokussieren auf die Bereitstellung von Modulen, einer Konsole, über die Befehle abgesetzt werden können, sowie eine Skriptsprache (Urrea, Cubillos, Cabrera-Paniagua & Mellado, 2019).

Hyperion Framework

Das Hyperion-Framework stellt eine Sammlung von Tools bereit, mittels welcher Prototypen einer Lösung schnell und effizient erstellt werden sollen. Hierbei liegt das Hauptaugenmerk auf der Identifizierung von Teilkomponenten, die an der guten Performance eines Algorithmus mitwirken (Swan, Özcan & Kendall, 2011). Insbesondere die rekursive Instanziierung von Hyperheuristiken einer Hyperheuristik werden durch das Hyperion-Framework ermöglicht (Swan et al., 2011).

ParHyFlex

ParHyFlex basiert auf den sequentiellen Framework namens HyFlex und fokussiert auf die parallele Bearbeitung von Hyperheuristiken, ohne dass der Anwendungsentwickler diese kennen muss (Van Onsem & Demoen, 2013). Grundebenen beinhalten Bibliotheken von bekannten, kombinierbaren Problemomänen sowie deren Benchmark. Hier angesiedelte Heuristiken stellen eine Sammlung von Operatoren zur Verfügung, sodass Änderungen in der Lösung erstellt werden können und Perturbationen aufgrund von Ähnlichkeiten in der Nachbarschaft aufgefunden werden können (Delft University of Technology 2013).

SATzilla

SATzilla ist ein konzeptioneller Ansatz mittels dem ein Portfolio an Algorithmen auf ein Problem angewendet wird. Im Fokus stehen Erfüllbarkeitsprobleme der Aussagenlogik (Boolean satisfiability problems), in denen gefragt wird, inwiefern eine aussagenlogische Formel erfüllbar ist. Ein offline-Lernprozess unterstützt zunächst die Entwicklung eines Portfolios. Im Anschluss wird jeder Portfolioeintrag, bzw. ausgewählter Algorithmus auf die Problem Instanz angewendet. Im direkten Vergleich kann somit der schnellste Algorithmus identifiziert werden (Xu, Hutter, Hoos & Leyton-Brown, 2008).

Snappy

Bei Snappy handelt es sich um einen konzeptionellen Ansatz mittels dem ein Portfolio an Algorithmen auf ein Problem angewendet werden kann. Vergleichbar mit SATzilla stehen Erfüllbarkeitsprobleme der Aussagenlogik (Boolean satisfiability problems) im Fokus. Snappy hebt sich hier allerdings hervor, da ein Offline-Lernen oder Wissen über komplexe ML-Tools nicht notwendig ist (Samulowitz, Reddy, Sabharwal & Sellmann, 2013).

Basierend auf einfachen nachbarschaftsbasierten Portfolioalgorithmen in Python, kann die eigene Performance durch online-Lernen verbessert werden, sodass Snappy State-of-the-Art-Benchmarks problemlos lösen kann (Snappy by Google, 2017).

5.5 Genetische Tools im Detail

DREAM

Die Java-basierte Softwareumgebung namens DREAM (Distributed Resource Evolutionary Algorithm Machine) umfasst eine Sammlung an Bibliotheken, User-Interfaces sowie einer Netzwerk-Engine, mittels welcher verteilte Peer-to-Peer-Umgebungen aufgebaut werden können, um heuristische Optimierungen realisieren zu können (DR-EA-M, 2013). Ein ursprünglicher Entwicklungsfokus liegt auf der Bereitstellung von Software- und Technologieinfrastrukturen, um eine effiziente Nutzung von CPU-Zeit zu realisieren und gleichzeitig dem Anspruch zu entsprechen, universell, offen und skalierbar zu sein (Paechter et al., 2000).

EASEA-CLOUD

Bei der EASEA-CLOUD handelt es sich um eine Open Source Softwareplattform, welche unter der „GNU Affero v3 General Public“-Lizenz zur Verfügung gestellt wird (SONIC (Stochastic Optimisation and Nature Inspired Computing) group of the BFO team, 2017). Ihr Name EASEA leitet sich aus dem Englischen „EAsy Specification of Evolutionary Algorithms“ ab und steht auch als nicht-Cloud-basierte Variante EASEA zur Verfügung.

Da die Plattform primär Wissenschaftler adressiert, welche rudimentäre Informatikkompetenzen aufweisen, können evolutionäre Algorithmen mit Hilfe der Plattform leicht parallelisiert und auf Multi-Kern-Architekturen oder auch in Computer-Ökosystemen mit oder ohne GPU verteilt berechnet werden. Die Weiterentwicklung der Plattform kombiniert evolutionäre Algorithmen mit KNN und wird als PANACEA „Parallel Artificial Neural Architecture Coupling with Evolutionary Algorithms“ bezeichnet.

Evolving Objects (EO)

Das Framework „Evolving Objects (EO)“ kann als evolutionäre Programmierbibliothek betrachtet werden, welche das Erstellen von stochastischen Optimierungsalgorithmen ermöglicht (Keijzer, Merelo, Romero & Schoenauer, 2001). Als komponentenbasiertes Rahmenwerk können einzelne Bauteile ähnlich zu Legobausteinen zusammengeslossen werden und einen komplexeren Algorithmus bilden, welcher an eine Fitnessfunktion gekoppelt wird (Keijzer et al., 2001). Im iterativen Vorgehen werden Zufallsänderungen mit dieser Funktion evaluiert (GeNeura Team, 1997a), sodass ein Optimum für kontinuierliche sowie kombinierbare Lösungsrepräsentationen gefunden werden können.

GAGS

Bei GAGS (genetic algorithm C++ class library) handelt es sich um eine wenig verbreitete C++-basierte Programmbibliothek, welche Klassen zur Realisierung genetischer Algorithmen bereitstellt (GeNeura Team, 1997b). Diese repräsentieren hierbei Chromosomen, genetische Operatoren, Sichten auf Chromosomen sowie Populationen als Chromosomenliste (Merelo & Romero, 1999) und objektbasiert genetische Problemlösungen realisieren (Merelo & Prieto, 1996).

Galib

Bei Galib handelt es sich um eine wenig verbreitete Programmierbibliothek, welche in C++ verfasste genetische Algorithmen hält und diese für Unix-, Mac- sowie Windows-Plattformen bereitstellt (Wall, 1996). Das Tool unterstützt hierbei den dreistufigen Prozess zur Definition genetischer Ansätze, indem zunächst eine Repräsentation geschaffen werden kann, dann genetische Operatoren sowie eine Zielfunktion definiert werden können. Auf Basis dieser Definitionen sollen Probleme anhand der geschaffenen Strukturen effizient gelöst werden können (Wall, 1996). Das Bilden von Hierarchien von Klassen erleichtert hierbei das Wiederverwenden und Einsetzen definierter genetischer Ansätze.

HeuristicLab

Bei HeuristicLab handelt es sich um eine Softwareumgebung für heuristische und evolutionäre Algorithmen, welche es aufgrund seiner graphischen Oberfläche insbesondere Nutzern ohne fortgeschrittene Programmierfähigkeiten einen Einstieg in heuristische Verfahren ermöglichen möchte (Heuristic and Evolutionary Algorithms Laboratory, 2002). Da Algorithmen hier in operatorbasierten Graphen repräsentiert werden, können durch das Ändern oder Reorganisieren von Graphen die zugrundeliegenden Algorithmen angepasst werden. Obwohl HeuristicLab eine weite Bandbreite

an Lösungsansätzen für genetische Probleme sowie ML-Algorithmen hält, können diese durch Eigenimplementierungen als Plug-In erweitert werden.

JCLEC

JCLEC (Java Class Library for Evolutionary Computation) ist ein Softwaresystem für die evolutionäre Lösung von Problemen [Jcle2018]. Es ist Java-basiert und ermöglicht das Anwenden von evolutionären, genetischen Algorithmen sowie die Implementierung von genetischen und evolutionären Ansätzen (Ventura, Romero, Zafra, Delgado & Hervás, 2008). Hierbei stehen drei Hauptmodule zur Definition auf abstrakter Ebene, der Implementierung sowie ein Experimentenwizard zur Verfügung (Cano, Luna, Zafra & Ventura, 2015; Ventura et al., 2008). Als Open-Source-Projekt ist die Software unter der GNU General Public License (GPL) frei verfügbar.

SHARK ML Library

Die SHARK bezeichnet sich als eine schnelle, modulare und detailreiche Open-Source-ML-Programmiersbibliothek, welche C++-basiert ist und das Design und die Optimierung von adaptiven Systemen durch evolutionäre und Gradientenbasierten Algorithmen erleichtert [Shar2018]. Somit finden sich bei SHARK Ansätze zum Training von Feed-forward Netzen, von spezifischen ML-Ansätzen, wie Principal Component Analysis, Nearest Neighbor Classification, K-Means Clustering, Support Vektor Maschinen, Boltzmannmaschinen, aber auch einfacheren Ansätzen der linearen Regression oder zum Datenhandling. Die Bibliothek wird für Windows, MacOS X, und Linux unter der GNU Lesser General Public License bereitgestellt (Igel, Heidrich-Meisner & Glasmachers, 2008).

6 Resümee

Der Markt an verfügbaren ML-Lösungen ist für Novizen, aber auch für erfahrene Anwender oder Entwickler teilweise schwer zu überschauen. Es existiert eine Vielzahl von ML-Tools mit jeweils unterschiedlichem Funktionsumfang für diverse Anwendungsbereiche. Erschwerend kommt hinzu, dass die KI- und ML-Toolwelt hoch volatil ist – einige Lösungen werden von großen Firmen aufgekauft, unter anderem Namen weiterentwickelt und stetig erscheinen neue Lösungen. Selbst beim Verfassen dieses Buches mussten in der Finalisierungsphase immer wieder Tools aufgenommen werden, die aus Sicht der Autoren im Markt eine wichtige Rolle spielen werden. Als Beispiel sei an dieser Stelle die von Oracle am 22. August 2020 veröffentlichte ML-Bibliothek Tribuo für die Programmiersprache Java angebracht.

Das vorliegende Buch verfolgt das Ziel, einen Beitrag zur Systematisierung und Kontextualisierung verfügbarer ML-Tools zu leisten und somit dieser Herausforderung Abhilfe zu schaffen. Zu diesem Zweck wurden zunächst notwendige Grundlagen von ML und KI dargelegt. Darauf aufbauend wurden unterschiedliche Anwendungsbereiche vorgestellt. Beispielhaft sei an dieser Stelle auf die Einsatzmöglichkeiten zur Vorhersage oder Diagnose im Energiesektor, den Möglichkeiten im Bereich der Flugindustrie, aber auch in den Bereichen Gesundheit, Mobilität oder Finanzen und IT verwiesen. Anschließend wurde eine Auswahlmethodik für ML-Algorithmen vorgestellt. Im Fokus standen dabei u.a. eine unternehmensorientierte ML-Tool-Auswahl, die projektorientierte ML-Systementwicklung sowie ein Vorgehensmodell zum Einsatz des ausgewählten ML-Tools. Abschließend wurde eine Marktübersicht relevanter und aktueller Tools vorgestellt. Dabei wird gerade auch aufgrund des dynamischen Umfeldes in diesem Themenbereich kein Anspruch auf Vollständigkeit postuliert. Neben aktuellen ML-Frameworks und Bibliotheken wurden in den Klassen der genetischen und der heuristischen Tools ebenfalls eine Auswahl an Lösungen präsentiert, welchen auf Basis der SLR und aus Sicht der Autoren eine hohe Bedeutung zugemessen wird.

Durch den Aufbau des Buches von den Grundlagen, bis hin zu den einsortierten Tools wird sowohl dem Einsteiger als auch dem Fortgeschrittenen eine Hilfestellung zur konkreten Auswahl eines ML-Tools gewährt. Mithilfe der Anwendungsbeispiele werden Einsatzmöglichkeiten dargelegt, welche sich auf eigene und neuartige Herausforderungen übertragen lassen. Die zahlreichen Quellen erlauben ein genaueres Analysieren der Möglichkeiten und Lösungsansätze.

Unabhängig vom ausgewählten ML-Tools ist es entscheidend, das eigentliche Problem zu verstehen, welches es mit der zu entwickelten Anwendung zu lösen gilt. Obwohl

sich einige wenige Vertreter als mächtige Tools herausgestellt haben, schwimmen prototypisch entwickelte Tools als Proof-of-Concept auf den Markt. Diese stellen neue Ansätze und vielversprechende Verfahren – zum Großteil aus der Forschung kommend – einer breiten Masse bereit, verfügen jedoch meist über einen hochspezialisierten Charakter. Eine Auswahl der passenden ML-Tools zur Realisierung eines KI- und ML-Projekts wird somit weiterhin eine Herausforderung bleiben. Dieses Buch kann zum Beginn des Jahres 2021 einen Beitrag zur Übersicht und Orientierung leisten. Gleichzeitig erlauben die definierten Bausteine der Marktübersicht das schnelle Eingliedern noch kommender ML-Tools.

Literaturverzeichnis

- .Net Foundation. (2020). *machinelearning*. Verfügbar unter: <https://github.com/dotnet/machinelearning>.
- Accord.NET contributors. (2017). *Sample gallery*. Verfügbar unter: <http://accord-framework.net/samples.html>.
- Accord.NET contributors. (2020). *Current Status*. Verfügbar unter: <https://github.com/accord-net/framework#current-status>.
- Additive. (2020). *Salford Predictive Modeler: Machine Learning and Predictive Analysis*. Verfügbar unter: <https://www.additive-net.de/en/software/produkte/minitab/salford-systems>.
- Aggarwal, C. C. & Han, J. (2014). *Frequent Pattern Mining*: Springer Publishing Company, Incorporated.
- Agrawal, R., Imieliński, T. & Swami, A. (1993). *Mining association rules between sets of items in large databases*. Paper presented at the Proceedings of the 1993 ACM SIGMOD international conference on Management of data.
- Aha, D. W. (1997a). Artificial intelligence review. *Editorial: Lazy Learning*, 11(1-5), 7-10.
- Aha, D. W. (1997b). *Lazy learning*, Lazy learning: Kluwer Academic Publishers, Norwell, MA.
- Airlines, T. (2020). *Check-in-Optionen am Flughafen | Check-in | Turkish Airlines* ®. Verfügbar unter: <https://www.turkishairlines.com/de-de/any-questions/check-in/airport-check-in-options/>.
- Aletras, N., Tsarapatsanis, D., Preoțiuc-Pietro, D. & Lampos, V. (2016). Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective. *PeerJ Computer Science*, 2, e93.
- Alexandrov, A., Bergmann, R., Ewen, S., Freytag, J.-C., Hueske, F., Heise, A. et al. (2014). The stratosphere platform for big data analytics. *The VLDB Journal*, 23(6), 939-964.
- Alphabet. (2019, 2019). *Google to Acquire Looker*. Verfügbar unter: <https://abc.xyz/investor/news/releases/2019/0606/>.
- Altair Engineering Inc. (2020a). *Altair Knowledge Studio: Fortschrittliches maschinelles Lernen und künstliche Intelligenz*. Verfügbar unter: <https://www.altair.de/knowledge-studio/>.
- Altair Engineering Inc. (2020b). *Data Analytics*. Verfügbar unter: <https://www.altair.com/data-analytics-applications/>.
- Alteryx. (2020). *Warum Alteryx für Data Analysts*. Verfügbar unter: <https://www.alteryx.com/de/why-alteryx/data-analysts>.
- Amazon Web Services. (2020a). *HDInsight: Einfacher und kostengünstiger Dienst auf Unternehmensniveau für Open Source-Analysen*. Verfügbar unter: <https://azure.microsoft.com/de-de/services/hdinsight/>.
- Amazon Web Services. (2020b). *KNIME Server Small for AWS*. Verfügbar unter: <https://aws.amazon.com/marketplace/pp/KNIMEcom-KNIME-Server-Small-for-AWS/B07JH2TGJV>.

- Amazon Web Services. (2020c). *Machine Learning in AWS: Wir legen Machine Learning in die Hände eines jeden Entwicklers*. Verfügbar unter: <https://aws.amazon.com/de/machine-learning/>.
- Amjady, N., Keynia, F. & Zareipour, H. (2010). Short-term load forecast of microgrids by a new bilevel prediction strategy. *IEEE Transactions on smart grid*, 1(3), 286-294.
- Anaconda. (2017). *Conda*. Verfügbar unter: <https://docs.conda.io/en/latest/>.
- Anaconda. (2020). *Enterprise Edition: The end-to-end data science platform*. Verfügbar unter: <https://www.anaconda.com/enterprise/>.
- Apache Software Foundation. (2018a). *Apache Spark MLlib*. Verfügbar unter: <http://spark.apache.org/mllib/>.
- Apache Software Foundation. (2018b). *Apache Spark: Lightning-fast unified analytics engine*. Verfügbar unter: <https://spark.apache.org/>.
- Apache Software Foundation. (2018c). *Ozone*. Verfügbar unter: <https://hadoop.apache.org/ozone/>.
- Apache Software Foundation. (2019a). *Apache Kafka - Clients*. Verfügbar unter: <https://cwiki.apache.org/confluence/display/KAFKA/Clients>.
- Apache Software Foundation. (2019b). *HDFS Architecture*. Verfügbar unter: <https://hadoop.apache.org/docs/r2.10.0/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html>.
- Apache Software Foundation. (2019c). *MapReduce Tutorial*. Verfügbar unter: <https://hadoop.apache.org/docs/r2.10.0/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html>.
- Apache Software Foundation. (2020a). *Data Streaming Fault Tolerance*. Verfügbar unter: https://ci.apache.org/projects/flink/flink-docs-release-1.10/internals/stream_checkpointing.html.
- Apache Software Foundation. (2020b). *INTRODUCTION: Everything you need to know about Kafka in 10 minutes*. Verfügbar unter: <https://kafka.apache.org/intro>.
- Apache Software Foundation. (2020c). *Mahout: For Creating Scalable Performant Machine Learning Applications*. Verfügbar unter: <https://mahout.apache.org/>.
- Apache Software Foundation. (2020d). *Singa: Software Stack*. Verfügbar unter: <https://singa.apache.org/docs/software-stack/>.
- Apple. (2019). *Memoji auf dem iPhone oder iPad Pro verwenden*. Verfügbar unter: <https://support.apple.com/de-de/HT208986>.
- Apple. (2020a). *Apple CarPlay: Der perfekte Beifahrer*. Verfügbar unter: <http://www.apple.com/de/ios/carplay/>.
- Apple. (2020b). *Turicreate*. Verfügbar unter: <https://github.com/apple/turicreate>.
- Araghi, B. N., Pedersen, K. S., Christensen, L. T., Krishnan, R. & Lahrmann, H. (2015). Accuracy of travel time estimation using Bluetooth technology: Case study Limfjord tunnel Aalborg. *International Journal of Intelligent Transportation Systems Research*, 13(3), 166-191.
- Atkeson, C., Moore, A. & Schaal, S. (1997). Locally weighted learning. *AI Review* 11: 11-73. *Kluwer*.

- Audi AG. (2020). *Head-up-Display*. Verfügbar unter: <https://www.audi-technology-portal.de/de/elektrik-elektronik/bedienung/head-up-display>.
- Auer, P. (1997). *On learning from multi-instance examples: Empirical evaluation of a theoretical approach*. Paper presented at the Proceedings of the fourteenth international conference on machine learning.
- BaggageAI. (2020). *BaggageAI*. Verfügbar unter: <http://www.baggageai.com/>.
- Barnhart, C., Cohn, A., Johnson, E., Klabjan, D., Nemhauser, G. & Vance, P. (2003). *Airline Crew Scheduling*. In (S. 517-560).
- Bayesia. (2020). *BayesiaLab WebSimulator*. Verfügbar unter: <https://library.bayesia.com/articles/#!/bayesialab-knowledge-hub/bayesialab-websimulator-12714041>.
- Benker, H. (2016). *MATHEMATICA kompakt*: Springer.
- Berg, B. & Penny, S. (2013). *SAP HANA: An Introduction*: Galileo Press.
- Berg, S., Kutra, D., Kroeger, T., Straehle, C. N., Kausler, B. X., Haubold, C. et al. (2019). *Ilastik: interactive machine learning for (bio) image analysis*. *Nature Methods*, 1-7.
- Bergenheim, C., Shladover, S., Coelingh, E., Englund, C. & Tsugawa, S. (2012). *Overview of platooning systems*. Paper presented at the Proceedings of the 19th ITS World Congress, Oct 22-26, Vienna, Austria (2012).
- Berkeley Artificial Intelligence Research. (2020). *Caffe*. Verfügbar unter: <https://caffe.berkeleyvision.org/>.
- Berlinger, E. (2019). *Gewaltige Woche im Analyse & BI Umfeld*. Verfügbar unter: <https://www.massiveart.com/blog/gewaltige-woche-im-analyse-bi-umfeld>.
- Bierwirth, C. (2014). *Enzyklopädie der Wirtschaftsinformatik (Online Lexikon) - Heuristik*. Verfügbar unter: <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/technologien-methoden/Operations-Research/Heuristik/index.html>.
- Bifet, A., Gavaldà, R., Holmes, G. & Pfahringer, B. (2018). *Machine learning for data streams: with practical examples in MOA*: MIT Press.
- Bifet, A., Holmes, G., Kirkby, R. & Pfahringer, B. (2010). *MOA: Massive Online Analysis*. *Journal of Machine Learning Research*, 11(5).
- BigML. (2020). *Bindings & Libraries*. Verfügbar unter: <https://bigml.com/tools/bindings>.
- Birattari, M., Bontempi, G. & Bersini, H. (1999). *Lazy learning meets the recursive least squares algorithm*. Paper presented at the Advances in neural information processing systems.
- BMW China. (2015). *INNOVATIONS. AHEAD OF THEIR TIME: Innovative functionality in the all-new BMW 7 Series*. Verfügbar unter: http://www.un1que.cc/cn/en/newvehicles/7series/sedan/2015/showroom/innovative_functionality.html.
- Bower, J. M. & Beeman, D. (2012). *The book of GENESIS: exploring realistic neural models with the GEneral NEural Simulation System*: Springer Science & Business Media.
- Boyd, E. (2019). *ONNX joins Linux Foundation*. Verfügbar unter: <https://cloudblogs.microsoft.com/opensource/2019/11/14/onnx-joins-linux-foundation/>.

- Brien, J. (2019). Radar-Gestensteuerung – FCC erlaubt Googles Project Soli. *t3n*. Verfügbar unter: <https://t3n.de/news/project-soli-fcc-google-radar-1135357/#:~:text=Mit%20Project%20Soli%20k%C3%B6nnen%20Eingaben,Microsoft%20setzt%20Google%20auf%20Radar.>
- Brink, H., Richards, J. & Fetherolf, M. (2016). *Real-World Machine Learning*: Manning Publications Co.
- Brownlee, J. (2016a). Introduction to the Python Deep Learning Library Theano.
- Brownlee, J. (2016b). *A Tour of the Weka Machine Learning Workbench*. Verfügbar unter: <https://machinelearningmastery.com/tour-weka-machine-learning-workbench/>.
- Bundesnetzagentur. (2020). *Unsere Energiewende: sicher, sauber, bezahlbar*. Verfügbar unter: <https://www.bmwi.de/Redaktion/DE/Dossier/energiewende.html>.
- Buxmann, P. & Schmidt, H. (2019). Grundlagen der künstlichen Intelligenz und des maschinellen Lernens. In *Künstliche Intelligenz* (S. 3-19): Springer.
- Cano, A., Luna, J. M., Zafra, A. & Ventura, S. (2015). A classification module for genetic programming algorithms in JCLEC. *The Journal of Machine Learning Research*, 16(1), 491-494.
- Cesar & Microsoft. (2019). *Announcing ML.NET 1.4 general availability (Machine Learning for .NET)*. Verfügbar unter: <https://devblogs.microsoft.com/dotnet/announcing-ml-net-1-4-global-availability-machine-learning-for-net/>.
- Chakrabarty, N. (2019). *A Data Mining Approach to Flight Arrival Delay Prediction for American Airlines*. Paper presented at the 2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON).
- Chowdhury, G. G. (2003). Natural language processing. *Annual Review of Information Science and Technology*, 37(1), 51-89.
- Cloudera Inc. (2020a). *Cloudera Data Science Workbench*. Verfügbar unter: <https://de.cloudera.com/products/data-science-and-engineering/data-science-workbench.html>.
- Cloudera Inc. (2020b). *Cloudera Data Science Workbench Overview*. Verfügbar unter: https://docs.cloudera.com/documentation/data-science-workbench/1-8-x/topics/cdsw_overview.html.
- Cloudflight. (2020). *Optical Baggage Tracking & Identification*. Verfügbar unter: <https://www.catalysts.cc/projects/baggage-tracking/>.
- Conejo, A. J., Contreras, J., Espínola, R. & Plazas, M. A. (2005). Forecasting electricity prices for a day-ahead pool-based electric energy market. *International journal of forecasting*, 21(3), 435-462.
- Conrad, A. (2020). *Looker vs Tableau: Which BI Tool is the Winner For Your Business?* Verfügbar unter: <https://www.selecthub.com/business-intelligence/looker-vs-tableau-bi-tools/#:~:text=%E2%80%9CPowered%20by%20Looker%E2%80%9D%20is%20the%20software%E2%80%99s%20embeddable%20analytics,some%20features%20of%20SQL%2C%20but%20with%20improved%20functionality.>

- Continental Automotive GmbH. (2014). *Head-up-Displays*. Verfügbar unter: <https://www.continental-automotive.com/de-DE/Passenger-Cars/Information-Management/Head-Up-Displays>.
- Contributor, G. (2019, 2019-05-14T18:00:43+00:00). *What can Artificial Intelligence do for the Aviation industry*. Verfügbar unter: <https://hub.packtpub.com/what-can-artificial-intelligence-do-for-the-aviation-industry/>.
- Cornell, D. (2008a). *Static Analysis, Dynamic Analysis and How to Use Them Together*. Denim Group. Verfügbar unter: https://denimgroup.com/media/pdfs/DenimGroup_StaticAnalysisTechniquesForTestingApplicationSecurity_OWASPSanAntonio_20080131.pdf.
- Cornell, D. (2008b). *Static Analysis, Dynamic Analysis and How to Use Them Together*. Verfügbar unter: https://denimgroup.com/media/pdfs/DenimGroup_StaticAnalysisTechniquesForTestingApplicationSecurity_OWASPSanAntonio_20080131.pdf.
- Cruz, J. A. & Wishart, D. S. (2006). Applications of machine learning in cancer prediction and prognosis. *Cancer informatics*, 2, 59-78.
- Curtin, R. R., Edel, M., Lozhnikov, M., Mentekidis, Y., Ghaisas, S. & Zhang, S. (2018). mlpack 3: a fast, flexible machine learning library. *Journal of Open Source Software*, 3(26), 726.
- Dadras, S., Gerdes, R. M. & Sharma, R. (2015). *Vehicular platooning in an adversarial environment*. Paper presented at the Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security.
- Databricks. (2020a). *Die Databricks Unified Data Analytics-Plattform*. Verfügbar unter: <https://databricks.com/de/product/unified-data-analytics-platform>.
- Databricks. (2020b). *Eine vereinheitlichte Plattform für Daten und KI*. Verfügbar unter: <https://databricks.com/de/>.
- Dataiku. (2020). *Your Path to Enterprise AI*. Verfügbar unter: <https://pages.dataiku.com/hubfs/Brochure-September-2018.pdf>.
- Datameer. (2020). *Product Overview: Unified Platform for the Entire Big Data Lifecycle*. Verfügbar unter: <https://www.datameer.com/product/>.
- DataRobot. (2020). *Machine Learning Operations (MLOps)*. Verfügbar unter: <https://www.datarobot.com/platform/mlops/>.
- Dean, J. & Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1), 107-113.
- Delft University of Technology (2013). *BNAIC 2013 – Proceedings of the 25th Benelux Conference on Artificial Intelligence*. Verfügbar unter: <http://resolver.tudelft.nl/uuid:980e5c74-a030-483e-b241-5eb7b1f9a10d>.
- Denecke, K., Lutz Hochreutener, S., Pöpel, A. & May, R. (2018). *Talking to ana: A mobile self-anamnesis application with conversational user interface*. Paper presented at the Proceedings of the 2018 International Conference on Digital Health.
- Desale, D. (2016). *Top 15 Frameworks for Machine Learning Experts*. In: *KDnuggets*. Verfügbar unter: <https://www.kdnuggets.com/2016/04/top-15-frameworks-machine-learning-experts.html>.
- DL4J. (2020). *Deep Learning for Java*. Verfügbar unter: <https://deeplearning4j.org/>.

- Dörner, R., Broll, W., Grimm, P. & Jung, B. (2014). *Virtual und Augmented (VR/AR) - Grundlagen und Methoden der Virtuellen und Augmentierten Realität*. Berlin Heidelberg: Springer-Verlag
- DR-EA-M. (2013). *DR-EA-M*. Verfügbar unter: <https://sourceforge.net/projects/dr-ea-m/>.
- DuPaul, N. (2017). *Static Testing vs. Dynamic Testing*. Verfügbar unter: <https://www.veracode.com/blog/2013/12/static-testing-vs-dynamic-testing>.
- EATON. (2017). *Self-Healing Grid*. Verfügbar unter: http://www.cooperindustries.com/content/public/en/power_systems/solutions/self-healing.html.
- Eaton, J. (2020). *GNU Octave: Scientific Programming Language*. Verfügbar unter: <https://www.gnu.org/software/octave/>.
- Eibe, F., Hall, M., Holmes, G., Kirkby, R., Pfahringer, B. & Witten, I. H. (2018). *Chapter 1. WEKA: A Machine Learning Workbench for Data Mining. Department of Computer Science*. Verfügbar unter: https://www.cs.waikato.ac.nz/ml/publications/2005/weka_dmh.pdf.
- Elki-project. (2020). *ELKI: Environment for Developing KDD-Applications Supported by Index-Structures*. Verfügbar unter: <https://elki-project.github.io/>.
- Elyasaf, A. & Sipper, M. (2014). Software review: the HeuristicLab framework. *Genetic Programming and Evolvable Machines*, 15(2), 215-218.
- europa.eu. (2020). *Air passenger rights*. Verfügbar unter: https://europa.eu/youreurope/citizens/travel/passenger-rights/air/index_en.htm#ex-circumstances-delayed-1.
- European Organisation for the Safety of Air Navigation (EUROCONTROL). (2018). *Predicting flight routes with a Deep Neural Network in the operational Air Traffic Flow and Capacity Management system*. Verfügbar unter: <https://www.eurocontrol.int/sites/default/files/2019-06/traffic-prediction-improvements-tpi-factsheet.pdf>.
- Everitt, B. S., Landau, S., Leese, M. and Stahl, D. (2011). An Introduction to Classification and Clustering. In *Cluster Analysis* (Bd. 5, S. 1-13): W. A. Shewhart, S. S. Wilks, B. S. Everitt, S. Landau, M. Leese and D. Stahl.
- Evsan, I. (2016). *Predictive Maintenance: Ein Kernstück der Industrie 4.0*. Verfügbar unter: <https://bigdatablog.de/2016/07/21/predictive-maintenance-ein-kernstueck-der-industrie-4-0/>.
- Fang, X., Misra, S., Xue, G. & Yang, D. (2011). Smart grid—The new and improved power grid: A survey. *IEEE communications surveys & tutorials*, 14(4), 944-980.
- Faruqui, A. & Sergici, S. (2010). Household response to dynamic pricing of electricity: a survey of 15 experiments. *Journal of regulatory Economics*, 38(2), 193-225.
- fastai. (2020). *swifitai*. Verfügbar unter: <https://github.com/fastai/swifitai>.
- Fischer, P. (2013). *Algorithmisches Lernen*: Springer-Verlag.
- Flath, C., Nicolay, D., Conte, T., van Dinther, C. & Filipova-Neumann, L. (2012). Cluster analysis of smart metering data. *Business & Information Systems Engineering*, 4(1), 31-39.

- Ford, V., Siraj, A. & Eberle, W. (2014). *Smart grid energy fraud detection using artificial neural networks*. Paper presented at the 2014 IEEE Symposium on Computational Intelligence Applications in Smart Grid (CIASG).
- Forrester Research. (2016). *The Total Economic Impact Of SAP Analytics: Key Findings*. Verfügbar unter: https://sap.parel.co/wp-content/uploads/sac-resources/The_Total_Economic_Impact_Of_SAP_Analytics.pdf.
- Fournier-Viger, P. (2020). *SPMF: Introduction*. Verfügbar unter: <http://www.philippe-fournier-viger.com/spmf/>.
- Frontline Systems. (2020a). *Analytic solver data mining add-in for excel (formerly XLminer)*. Verfügbar unter: <https://www.solver.com/xlminer-data-mining>.
- Frontline Systems. (2020b). *Frontline solvers excel product overview*. Verfügbar unter: <https://www.solver.com/products-overview>.
- G2. (2020). *SAS Base Reviews & Product Details*. Verfügbar unter: <https://www.g2.com/products/sas-base/reviews>.
- Gardner-Bonneau, D. & Blanchard, H. E. (2007). *Human factors and voice interactive systems*: Springer Science & Business Media.
- GeNeura Team. (1997a). *Evolving Objects (EO): an Evolutionary Computation Framework*. Verfügbar unter: <http://eodev.sourceforge.net/>.
- GeNeura Team. (1997b). *GAGS genetic algorithm C++ class library*. Verfügbar unter: <http://kal-el.ugr.es/GAGS/>.
- Gieselmann, H. (2018). *Valve bekämpft Cheater in Counter-Strike mit Deep Learning*. Verfügbar unter: <https://www.heise.de/newsticker/meldung/Valve-bekaempft-Cheater-in-Counter-Strike-mit-Deep-Learning-4003190.html>.
- Gigerenzer, G. & Todd, P. M. (1999). *Simple heuristics that make us smart*: Oxford University Press, USA.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*: Addison-Wesley Longman Publishing Co., Inc.
- Goldhill, O. (2018). *Machines know when someone's about to attempt suicide. How should we use that information?* Verfügbar unter: <https://qz.com/1367197/machines-know-when-someones-about-to-attempt-suicide-how-should-we-use-that-information/>.
- Google. (2020). *Was ist Colaboratory?* Verfügbar unter: <https://colab.research.google.com/notebooks/intro.ipynb#scrollTo=5fCEDCUqrC0>.
- Google Cloud. (2020a). *Cloud AutoML: Mit minimalem Aufwand und minimalen einschlägigen Fachkenntnissen hochwertige benutzerdefinierte Modelle für maschinelles Lernen trainieren*.
- Google Cloud. (2020b). *Einführung in BigQuery ML*. Verfügbar unter: <https://cloud.google.com/bigquery-ml/docs/bigqueryml-intro?hl=de>.
- Grum, M. (2020). *CoNM*. Verfügbar unter: <https://github.com/MarcusGrum/CoNM>.
- Gupta, S., Gordon, J. & Allison, K. (2018). *Standardizing on Keras: Guidance on High-level APIs in TensorFlow 2.0*. Verfügbar unter: <https://medium.com/tensorflow/standardizing-on-keras-guidance-on-high-level-apis-in-tensorflow-2-0-bad2b04c819a>.

- Gurobi. (2020a). *Gurobi Cloud: Fully managed Gurobi Solver on Microsoft Azure® and Amazon Web Services®*. Verfügbar unter: <https://www.gurobi.com/de/products/gurobi-instant-cloud/>.
- Gurobi. (2020b). *Gurobi Optimizer*. Verfügbar unter: <https://www.gurobi.com/de/products/gurobi-optimizer/>.
- H2O.ai. (2020). H2O: The #1 open-source machine learning platform for the enterprise.
- Hecking, C. (2016). *So fliegen Sie billig*. Verfügbar unter: <https://www.zeit.de/zustimmung?url=https%3A%2F%2Fwww.zeit.de%2F2016%2F31%2Fflug-preis-gebuehr-lufthansa-billigflieger%2Fseite-4>.
- Heide, A. & Henning, K. (2006). The "cognitive car": A roadmap for research issues in the automotive sector. *Annual reviews in control*, 30(2), 197-203.
- Hesse, S. & Schnell, G. (2014). *Sensoren für die Prozess-und Fabrikautomation*: Springer.
- Heuristic and Evolutionary Algorithms Laboratory. (2002). *HeuristicLab*. Verfügbar unter: <https://dev.heuristiclab.com/trac.fcgi/>.
- HeuristicLab. (2018). HeuristicLab – A Paradigm-Independent and Extensible Environment for Heuristic Optimization.
- Hilfiger, J. J. (2015). *Graphing Data with R: An Introduction*: " O'Reilly Media, Inc."
- Hitachi. (2019). *Pentaho Data Integration: The Power To Access, Prepare and Blend Multiple Data Sources Faster*. Verfügbar unter: <https://www.hitachivantara.com/en-us/pdf/datasheet/pentaho-data-integration-datasheet.pdf>.
- Hoef, C. (2018). *Gaming with Windows ML*. Verfügbar unter: <https://devblogs.microsoft.com/directx/gaming-with-windows-ml>.
- Holmes, G., Donkin, A. & Witten, I. H. (1994). *Weka: A machine learning workbench*. Paper presented at the Proceedings of ANZIS'94-Australian New Zealand Intelligent Information Systems Conference.
- Holtforth. (2018). *Machine Learning im E-commerce - Grundlagen, Anwendung und Zugang*.
- Howard, J. & Gugger, S. (2020). Fastai: A layered API for deep learning. *Information*, 11(2), 108.
- Hummel, J., Strehmel, N., Selbig, J., Walther, D. & Kopka, J. (2010). Decision tree supported substructure prediction of metabolites from GC-MS profiles. *Metabolomics*, 6(2), 322-333.
- IATA. (2020). *IATA Industry Fraud Prevention*. Verfügbar unter: <https://www.iata.org/en/programs/airline-distribution/industry-fraud-prevention-initiative/>.
- IBM. (2020a). *IBM Cognos Analytics: Alle nötigen Antworten an zentraler Stelle*. Verfügbar unter: <https://www.ibm.com/de-de/products/cognos-analytics>.
- IBM. (2020b). *IBM Data Science Experience (DSX) platform (now known as Watson Studio)*. Verfügbar unter: <https://developer.ibm.com/recipes/tutorials/ibm-data-science-experience-dsx-platform/>.
- IBM. (2020c). *IBM SPSS Modeler*. Verfügbar unter: <https://www.ibm.com/de-de/products/spss-modeler>.

- IBM. (2020d). *IBM SPSS Statistics*. Verfügbar unter: <https://www.ibm.com/de-de/products/spss-statistics>.
- IEC-62559. (2013). 62559: Use Case Methodology–Part 2: Definition of the templates for use cases, actor list and requirements list: IEC TC8 Standard.
- Igel, C., Heidrich-Meisner, V. & Glasmachers, T. (2008). Shark. *Journal of machine learning research*, 9(Jun), 993-996.
- Instream Oy. (2017). *Master the energy market with our agile tools*. Verfügbar unter: <https://www.gasum.com/en/gas--energy/energy-market-services/software-solutions/>.
- Izadian, A., Girrens, N. & Khayyer, P. (2013). Renewable energy policies: A brief review of the latest US and EU policies. *IEEE Industrial Electronics Magazine*, 7(3), 21-34.
- Jelica, A. (2016). *Microsoft & Mercedes-Benz: Partnerschaft bringt Office ins Auto*. Verfügbar unter: <https://windowsarea.de/2016/09/microsoft-mercedes-benz-partnerschaft-bringt-office-ins-auto/>.
- Jiang, R., Lu, R., Wang, Y., Luo, J., Shen, C. & Shen, X. (2014). Energy-theft detection issues for advanced metering infrastructure in smart grid. *Tsinghua Science and Technology*, 19(2), 105-120.
- Johanning, V. & Mildner, R. (2015). *Car IT kompakt*. Wiesbaden: Springer.
- Joos, T. (2020). *Workflows für KI und ML mit Kubernetes bereitstellen: Kubeflow 1.x: Machine Learning mit Kubernetes*. Verfügbar unter: <https://www.datacenter-insider.de/kubeflow-1x-machine-learning-mit-kubernetes-a-911750/>.
- Kamphuis, C., Barsom, E., Schijven, M. & Christoph, N. (2014). Augmented reality in medical education? *Perspectives on medical education*, 3(4), 300-311.
- Kavathekar, P. & Chen, Y. (2011). *Vehicle platooning: A brief survey and categorization*. Paper presented at the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference.
- Keijzer, M., Merelo, J. J., Romero, G. & Schoenauer, M. (2001). *Evolving objects: A general purpose evolutionary computation library*. Paper presented at the International Conference on Artificial Evolution (Evolution Artificielle).
- Kennedy, J. (2010). Particle Swarm Optimization. In C. Sammut & G. I. Webb (Hrsg.), *Encyclopedia of Machine Learning* (S. 760-766). Boston, MA: Springer US.
- Knime. (2020). *KNIME Analytics Platform: Creating data science*. Verfügbar unter: <https://www.knime.com/knime-analytics-platform>.
- Knime AG. (2020). *KNIME Analytics Platform: Creating data science*. Verfügbar unter: <https://www.knime.com/knime-analytics-platform>.
- Koshy, J. (2017). *Top 10 Machine Learning Frameworks*. Verfügbar unter: <https://www.promptcloud.com/blog/top-10-machine-learning-frameworks>.
- Kramer, O. (2006). *Evolutionäre Algorithmen kurz gefasst*. Verfügbar unter: https://www.researchgate.net/profile/Oliver_Kramer2/publication/251750785_Evolution_äre_Algorithmen_kurz_gefasst1/links/543d135a0cf2c432f74247ea.pdf
- Krassmann, S. & Egbert, S. (2018). Predictive Policing: Eine ethnographische Studie neuer Technologien zur Vorhersage von Straftaten und ihre Folgen für die

polizeiliche Praxis. Verfügbar unter: <https://www.wiso.uni-hamburg.de/fachbereich-sowi/ueber-den-fachbereich/fachgebiete/fachgebiet-kriminologische-sozialforschung/predictive-policing/egbert-krasman-2019-predictive-policing-projektabschlussbericht.pdf>.

Krempl, S. (2018). Predictive Policing: Die Polizei arbeitet verstärkt wie ein Geheimdienst. Verfügbar unter: <https://www.heise.de/newsticker/meldung/Predictive-Policing-Die-Polizei-arbeitet-verstaerkt-wie-ein-Geheimdienst-4008214.html>.

Kuhn, M. (2019). *The caret Package*. Verfügbar unter: <http://topepo.github.io/caret/>.

L'Oréal. (2017). *Kérastase and Withings Unveil World's First Smart Hairbrush at CES 2017*. Verfügbar unter: <https://www.loreal.com/media/press-releases/2017/jan/kerastase-and-withings-unveil-worlds-first-smart-hairbrush-at-ces-2017>.

Larrañaga, P., Atienza, D., Diaz-Rozo, J., Ogbechie, A., Puerto-Santana, C. E. & Bielza, C. (2018). *Industrial applications of machine learning*: CRC Press.

Lehrbaum, A. & Musliu, N. (2012). *A new hyperheuristic algorithm for cross-domain search problems*. Paper presented at the International Conference on Learning and Intelligent Optimization.

Li, N., Chen, L. & Low, S. H. (2011). *Optimal demand response based on utility maximization in power networks*. Paper presented at the 2011 IEEE power and energy society general meeting.

Li, T.-R., Xu, Y., Ruan, D. & Pan, W.-m. (2005). Sequential Pattern Mining*. In D. Ruan, G. Chen, E. E. Kerre & G. Wets (Hrsg.), *Intelligent Data Mining: Techniques and Applications* (S. 103-122). Berlin, Heidelberg: Springer Berlin Heidelberg.

Looker Data Sciences Inc. (2020). *Data science analytics & business intelligence*. Verfügbar unter: <https://looker.com/solutions/data-science>.

Lu, J. (2018). Machine learning modeling for time series problem: Predicting flight ticket prices. *arXiv:1705.07205 [cs]*.

Lucks, K. (2020). *Der Wettlauf um die Digitalisierung: Potenziale und Hürden in Industrie, Gesellschaft und Verwaltung*: Schäffer-Poeschel.

Lufthansa. (2019). *Künstliche Intelligenz (KI): Wegbereiter für mehr Effizienz in der digitalen Luftfahrt*. Verfügbar unter: <https://www.lhsystems.de/article/kuenstliche-intelligenz-ki-wegbereiter-fuer-mehr-effizienz-der-digitalen-luftfahrt>.

LuggageHero. (2019). *Lost Luggage Report 2019*. Verfügbar unter: <https://luggagehero.com/de/lost-luggage-report-2019/>.

Lundberg, S. M., Nair, B., Vavilala, M. S., Horibe, M., Eisses, M. J., Adams, T. et al. (2018). Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature biomedical engineering*, 2(10), 749-760.

Marr, B. (2016). *What Everyone Should Know About Cognitive Computing*. Verfügbar unter: <https://www.forbes.com/sites/bernardmarr/2016/03/23/what-everyone-should-know-about-cognitive-computing/#2c8b32315088>

Masong, F. (2018). Leaving the Nest: Heron donated to Apache Software Foundation.

Matthews, S. (2016). *What is cognitive IoT?* Verfügbar unter: <https://www.ibmbigdatahub.com/blog/what-cognitive-iot>.

- Mayer, H. M. (2016). *Getting strategic about sub-station spare parts* Verfügbar unter: <https://www.engerati.com/energy-retail/getting-strategic-about-substation-spare-parts/>.
- Mayo, M. (2016). *Top 10 Deep Learning Projects on Github*. Verfügbar unter: <https://www.kdnuggets.com/2016/01/top-10-deep-learning-github.html>.
- Merelo, J. J. & Prieto, A. (1996). *GAGS, a flexible object-oriented library for evolutionary computation*. Paper presented at the Proceedings of the First International Workshop on Machine Learning, Forecasting and Optimization.
- Merelo, J. J. & Romero, G. (1999). Doing GAs with GAGS. In *Practical handbook of genetic algorithms* (Bd. 3). Boca Raton, FL: CRC Press.
- Metaflow. (2020a). *A framework for real-life data science*. Verfügbar unter: <https://metaflow.org/>.
- Metaflow. (2020b). *Metaflow on AWS*.
- Metaflow. (2020c). *What is Metaflow*. Verfügbar unter: <https://docs.metaflow.org/introduction/what-is-metaflow>.
- Meyer, G. & Beiker, S. (2015). *Road vehicle automation*: Springer.
- Meyer, G. & Valldorf, J. (2011). *Advanced microsystems for automotive applications 2010: smart systems for green cars and safe mobility*: Springer Science & Business Media.
- Meyer, M. (2015). *Machine learning to detect online grooming*. Verfügbar unter: <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A846981&dswid=-1605>.
- Microsoft. (2015). *Windows Embedded Automotive 7*. Verfügbar unter: <https://www.microsoft.com/windowseembedded/en-us/windows-embedded-automotive-7.aspx>.
- Microsoft. (2017a). *AWS and Microsoft announce Gluon, making deep learning accessible to all developers*. Verfügbar unter: <https://news.microsoft.com/2017/10/12/aws-and-microsoft-announce-gluon-making-deep-learning-accessible-to-all-developers/>.
- Microsoft. (2017b). *The Microsoft Cognitive Toolkit*. Verfügbar unter: <https://docs.microsoft.com/en-us/cognitive-toolkit/>.
- Microsoft. (2019a). *Hallo ML.NET-Welt*. Verfügbar unter: <https://docs.microsoft.com/de-de/dotnet/machine-learning/how-does-ml-dotnet-work#hello-mlnet-world>.
- Microsoft. (2019b). *What is Machine Learning Server?* Verfügbar unter: <https://docs.microsoft.com/en-us/machine-learning-server/what-is-machine-learning-server>.
- Microsoft. (2020a). *Azure Machine Learning: Machine-Learning-Dienst für Unternehmen zur schnelleren Erstellung und Bereitstellung von Modellen*. Verfügbar unter: <https://azure.microsoft.com/de-de/services/machine-learning/>.
- Microsoft. (2020b). *Microsoft Machine Learning Server*. Verfügbar unter: [https://download.microsoft.com/download/3/A/B/3AB5E5CD-8A6C-4596-849D-25285DCD8C21/Machine Learning Server Datasheet EN US.pdf](https://download.microsoft.com/download/3/A/B/3AB5E5CD-8A6C-4596-849D-25285DCD8C21/Machine%20Learning%20Server%20Datasheet%20EN%20US.pdf).

- Microsoft. (2020c). *XLMiner Analysis ToolPak*. Verfügbar unter: <https://appssource.microsoft.com/de-de/product/office/WA104379190?tab=Overview>.
- Microsoft Corp. (2014). *Vowpal Wabbit*. Verfügbar unter: https://github.com/VowpalWabbit/vowpal_wabbit/wiki
- MicroStrategy. (2020). *Wichtiger denn je: Schnelle und fundierte Entscheidungen*. Verfügbar unter: <https://www3.microstrategy.com/de>.
- Miessler, D. (2018). *The Future of Insurance is Real-time Auto-Adjusted Premiums*. Verfügbar unter: <https://danielmiessler.com/blog/the-future-of-insurance-is-real-time-auto-adjusted-premiums/>.
- Milsoft. (2017). *Outage Management Systems | OMS Software for Utilities*. Verfügbar unter: <https://www.milsoft.com/engineering-operations/outage-management/>.
- Minitab. (2020). *Salford Predictive Modeler: Software für maschinelles Lernen und prädiktive Analysen*. Verfügbar unter: <https://www.minitab.com/de-de/products/spm/>.
- MOA. (2020a). *Machine Learning for Data Streams*. Verfügbar unter: <https://moa.cms.waikato.ac.nz/>.
- MOA. (2020b). *MOA Blog*. Verfügbar unter: <https://moa.cms.waikato.ac.nz/blog/>.
- Moher, D., Liberati, A., Tetzlaff, J., Altman, D. G. & Group, P. (2009). Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement. *PLoS med*, 6(7), e1000097.
- Moraschi, D. (2013). *Business Intelligence with MicroStrategy Cookbook*: Packt Publishing Ltd.
- Mousa, A. (2019). *Artificial Intelligent in Aviation /AI in Aviation/Intelligent Aviation. Plenary Speech in EuroSciCon*. Verfügbar unter: <https://de.slideshare.net/assemam/artificial-intelligent-in-aviation-ai-in-aviationintelligent-aviation>.
- Mukhopadhyay, S. C. & Leung, H. (2010). *Advances in wireless sensors and sensor networks* (Bd. 64): Springer.
- Müller, B. (2018). VORAUSSCHAUENDE WARTUNG IN DER INDUSTRIE 4.0: Der größte Störfaktor ist der Mensch. *Wirtschaftswoche*. Verfügbar unter: <https://www.wiwo.de/unternehmen/mittelstand/hannovermesse/vorausschau-ende-wartung-in-der-industrie-4-0-der-groesste-stoerfaktor-ist-der-mensch/21166908.html>.
- Müller, K. J. (2011). Verordnete Sicherheit—das Schutzprofil für das Smart Metering Gateway. *Datenschutz und Datensicherheit-DuD*, 35(8), 547.
- Murphy, R. F. (2011). An active role for machine learning in drug development. *Nature chemical biology*, 7(6), 327-330.
- Nadem, M. (2015). *Why SonarQube: An Introduction to Static Code Analysis*. Verfügbar unter: <https://dzone.com/articles/why-sonarqube-1>.
- Neubig, G., Dyer, C., Goldberg, Y., Matthews, A., Ammar, W., Anastasopoulos, A. et al. (2017). Dynet: The dynamic neural network toolkit. *arXiv preprint arXiv:1701.03980*.
- Neumann, M. (2015). *MÖGLICHKEITEN UND GRENZEN IN DER DYNAMISCHEN CODEANALYSE VON C++ SOFTWARE*. Verfügbar unter:

- <http://docplayer.org/18045534-Moeglichkeiten-und-grenzen-in-der-dynamischen-codeanalyse-von-c-software-von-matthias-neumann.html>.
- Nilsson, N. J. (2009). *The quest for artificial intelligence*: Cambridge University Press.
- Octave. (2020). *A Brief Introduction to Octave*. Verfügbar unter: <https://octave.org/doc/v5.2.0/Introduction.html#Introduction>.
- Omar, S., Ngadi, A. & Jebur, H. H. (2013). Machine learning techniques for anomaly detection: an overview. *International Journal of Computer Applications*, 79(2).
- Onnx. (2019a). About.
- Onnx. (2019b). Supported tools: Build model.
- Opalwave. (2020). *Predictive Analytics*. Verfügbar unter: <https://opalwave.com/sap-analytics/predictive-analytics/>.
- Oracle. (2017). *Utilities Meter Data Management | Oracle*. Verfügbar unter: <https://www.oracle.com/industries/utilities/products/meter-data-management/>.
- Orange. (2020a). *Orange3*. Verfügbar unter: {Orange, 2020 #321}.
- Orange. (2020b). *Teachers and Students Love It*. Verfügbar unter: https://orange.biolab.si/home/teachers_and_students_love_it/.
- Otte, R. (2019). *Künstliche Intelligenz für Dummies*: John Wiley & Sons.
- Paechter, B., Back, T., Schoenauer, M., Sebag, M., Eiben, A., Merelo, J. J. et al. (2000). *A distributed resource evolutionary algorithm machine (DREAM)*. Paper presented at the Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No. 00TH8512).
- Palem, G. (2013). Condition-based maintenance using sensor arrays and telematics. *arXiv preprint arXiv:1309.1921*.
- Palladino, T. (2018). *Here Are All the Things in You Can Do in Augmented Reality with the Samsung Galaxy S9 & S9+*. Verfügbar unter: <https://mobile-ar.reality.news/news/here-are-all-things-you-can-do-augmented-reality-with-samsung-galaxy-s9-s9-0183131/>.
- Papadakis, M. (2017). Predicting Airfare Prices. Verfügbar unter: <http://cs229.stanford.edu/proj2012/Papadakis-PredictingAirfarePrices.pdf>.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O. et al. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, 12, 2825-2830.
- Piatetsky, G. (2019). *Python leads the 11 top Data Science, Machine Learning platforms: Trends and Analysis*. Verfügbar unter: <https://www.kdnuggets.com/2019/05/poll-top-data-science-machine-learning-platforms.html>.
- Pickford, A. T. & Blythe, P. T. (2006). *Road user charging and electronic toll collection*.
- Pitt, M., Brown, A. & Smith, A. (2002). Waste Management at Airports. *Facilities*, 20, 198-207.
- Pocock, A. (2020). *Tribuo*. Verfügbar unter: <https://github.com/oracle/tribuo/releases/tag/v4.0.0>.
- Poli, R., Kennedy, J. & Blackwell, T. (2007). Particle swarm optimization. *Swarm Intelligence*, 1(1), 33-57.

- Predictive Analytics Today. (2020). *SAS Enterprise Miner*. Verfügbar unter: <https://www.predictiveanalyticstoday.com/sas-enterprise-miner/>.
- Prince, P. (2020). *Delta Gives Green Light to RFID Baggage Tracking* | *RFID JOURNAL*. Verfügbar unter: <https://www.rfidjournal.com/delta-gives-green-light-to-rfid-baggage-tracking>.
- Pudjianto, D., Ramsay, C. & Strbac, G. (2007). Virtual power plant and system integration of distributed energy resources. *IET Renewable power generation*, 1(1), 10-16.
- Püttmann, A. (2017, 2017-05-02T16:48:08+00:00). *Overbooking: Warum überbuchen Airlines ihre Flüge?* Verfügbar unter: <https://pilotstories.de/overbooking-ueberbuchungen-bei-airlines/>.
- PwC. (2017). Artificial Intelligence and Robotics – 2017: Leveraging artificial intelligence and robotics for sustainable growth. Verfügbar unter: <https://www.pwc.in/assets/pdfs/publications/2017/artificial-intelligence-and-robotics-2017.pdf>.
- PyCaret. (2020). *Modules*. Verfügbar unter: <https://pycaret.org/modules/>.
- PyXLL. (2020). *Write Excel Add-Ins in Python*. Verfügbar unter: <https://www.pyxll.com/>.
- Qian, L. P., Zhang, Y. J. A., Huang, J. & Wu, Y. (2013). Demand response management via real-time electricity price control in smart grids. *IEEE Journal on Selected areas in Communications*, 31(7), 1268-1280.
- QlikTech International AB. (2020). *Moderne Datenanalysen für den unternehmensweiten Einsatz*. Verfügbar unter: <https://www.qlik.com/de-de/products/qlik-sense#modern-analytics>.
- Rajankar, S. & Sakharkar, N. (2019). A Survey on Flight Pricing Prediction using Machine Learning. *International Journal of Engineering Research & Technology*, 8(6).
- Ramacher, U. & von der Malsburg, C. (2009). *Zur Konstruktion künstlicher Gehirne*: Springer Science & Business Media.
- Ramasamy, K. (2015). *Flying faster with Twitter Heron*. Verfügbar unter: https://blog.twitter.com/engineering/en_us/a/2015/flying-faster-with-twitter-heron.html.
- Ramsundar, B., Eastman, P., Walters, P. & Pande, V. (2019). *Deep learning for the life sciences: applying deep learning to genomics, microscopy, drug discovery, and more*: O'Reilly Media, Inc.
- Rapidminer. (2020). RapidMiner Studio: Comprehensive data science platform with visual workflow design and full automation.
- Ren, Q. (2017). When to Book: Predicting Flight Pricing. Verfügbar unter: <http://cs229.stanford.edu/proj2017/final-reports/5244211.pdf>.
- Rohleder, B. (2018). *Die Zukunft des Reisens ist digital*. Verfügbar unter: <https://www.bitkom.org/sites/default/files/2019-06/bitkom-pressekonferenz-die-zukunft-des-reisens-ist-digital-070318-praesentation.pdf>.
- Rojas, R. (1996). The backpropagation algorithm. In *Neural networks* (S. 149-182): Springer.

- Rosterize. (2020). *Rosterize – Оперативное оптимизационное планирование*. Verfügbar unter: <https://rosterize.cloud/>.
- Rouse, M. (2010). *Source code analysis*. Verfügbar unter: <https://searchsoftwarequality.techtarget.com/definition/source-code-analysis>.
- Roweis, S. T. & Saul, L. K. (2000). Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500), 2323-2326.
- Rulequest. (2017). *Is See5/C5.0 Better Than C4.5?* Verfügbar unter: <https://www.rulequest.com/see5-comparison.html>.
- Ryser-Welch, P. & Miller, J. (2014a). *A Review of Hyper-Heuristic Frameworks*.
- Ryser-Welch, P. & Miller, J. (2014b). *A Review of Hyper-Heuristic Frameworks*. Paper presented at the AISB 2014 - 50th Annual Convention of the AISB.
- Sæle, H. & Grande, O. S. (2011). Demand response from household customers: experiences from a pilot study in Norway. *IEEE Transactions on Smart Grid*, 2(1), 102-109.
- Salesforce. (2016). *Lead-Management*. Verfügbar unter: <https://www.salesforce.com/de/products/sales-cloud/features/marketing-lead-tracking-solutions/>.
- Salhi, S. (2017). Population-Based Heuristics. In *Heuristic Search: The Emerging Science of Problem Solving* (S. 77-128). Cham: Springer International Publishing.
- Samsung. (2019). *Augmented Reality - AR-Emoji*. Verfügbar unter: <https://www.samsung.com/de/smartphones/galaxy-s9/augmented-reality/>.
- Samulowitz, H., Reddy, C., Sabharwal, A. & Sellmann, M. (2013). Snappy: A Simple Algorithm Portfolio. In M. Järvisalo & A. Van Gelder (Hrsg.), *Theory and Applications of Satisfiability Testing – SAT 2013* (Bd. 7962). Berlin, Heidelberg: Springer.
- SAP. (2020). *Harness the power of an in-memory database with SAP HANA*. Verfügbar unter: <https://www.sap.com/products/hana.html>.
- SAS. (2016). *SAS Viya: Eine innovative, offene Plattform für jede analytische Herausforderung*. Verfügbar unter: https://www.sas.com/content/dam/SAS/bp_de/doc/solutionoverview/sas-viya-ger.pdf#:~:text=SAS%20Viya%20ist%20eine%20einzigartige%20Plattform,%20Sie%20erm%C3%B6glicht,schnelle%20Antworten%20und%20immer%20die%20besten%20Ergebnisse%20liefern.
- SAS. (2020a). *Announcing: JMP® Live and JMP® 15*. Verfügbar unter: https://www.jmp.com/en_us/software/new-release/preview-jmp-live-jmp15.html.
- SAS. (2020b). *JMP®: Software zur statistischen Datenanalyse für Mac und Windows*. Verfügbar unter: https://www.jmp.com/de_de/software/data-analysis-software.html.
- Scannera. (2020). *SCANNERA – Smart Real Time Video Analysis Software*. Verfügbar unter: <https://scannera.ai/>.
- Schultz, M. & Reitmann, S. (2019). Machine learning approach to predict aircraft boarding. *Transportation Research Part C: Emerging Technologies*, 98, 391-408.

- Sengupta, S. (2019, 2019-03-26T07:04:41.871Z). *Flight Price prediction*. Verfügbar unter: <https://medium.com/code-to-express/flight-price-prediction-7c83616a13bb>.
- Shalev-Shwartz, S. & Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*: Cambridge University Press.
- Sharma, V., Kumar, A., Lakshmi Panat, D. & Karajkhede, G. (2015). Malaria outbreak prediction model using machine learning. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, 4(12).
- Shelhamer, E. (2014). *Deep Learning for Computer Vision with Caffe and cuDNN*. Verfügbar unter: <https://developer.nvidia.com/blog/deep-learning-computer-vision-caffe-cudnn/>.
- Siebenpfeiffer, W. (2015). *Fahrerassistenzsysteme und Effiziente Antriebe*: Springer.
- Siemens. (2017). *SIMATIC WinCC V7*. Verfügbar unter: <https://new.siemens.com/global/de/produkte/automatisierung/industrie-software/automatisierungs-software/scada/simatic-wincc-v7.html#default.aspx>.
- Sisense. (2020a). *Agile BI-Lösung für moderne Analytic Apps*. Verfügbar unter: <https://www.sisense.com/de/why-sisense/>.
- Sisense. (2020b). *Blox*. Verfügbar unter: <https://www.sisense.com/marketplace/blox-widgets/>.
- Sisense. (2020c). *Sisense of Rules the Rewrites Analytics*. Verfügbar unter: <https://cdn.sisense.com/wp-content/uploads/Sisense-Hunch-Brochure.pdf>.
- Smart Grid Coordination Group. (2012). *Smart Grid Reference Architecture*. Verfügbar unter: https://ec.europa.eu/energy/sites/ener/files/documents/xpert_group1_reference_architecture.pdf.
- Snappy by Google. (2017). *Snappy – A fast compressor/decompressor*. Verfügbar unter: <https://google.github.io/snappy/>.
- Software Systems, I. (2017). *Meter Management*. Verfügbar unter: <http://software-systems.net/ubillity/meter-management/>.
- Sommer, C., Straehle, C., Köthe, U. & Hamprecht, F. A. (2011, 30 March-2 April 2011). *Ilastik: Interactive learning and segmentation toolkit*. Paper presented at the 2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro.
- SONIC (Stochastic Optimisation and Nature Inspired Computing) group of the BFO team. (2017). *EASEA platform*. Verfügbar unter: http://easea.unistra.fr/index.php/EASEA_platform.
- Sonnenburg, S., Rätsch, G., Henschel, S., Widmer, C., Behr, J., Zien, A. et al. (2010). The SHOGUN machine learning toolbox. *The Journal of Machine Learning Research*, 11, 1799-1802.
- Souza, C., Kirillov, A., Catalano, M. D. & Accord.NET contributors. (2014). *The Accord.NET Framework*. Verfügbar unter: <http://accord-framework.net>.
- Sqream. (2020). *Recommended Hardware Configuration for Sqream DB*. Verfügbar unter: <https://sqream.com/product/hardware/>.

- Stanton, N. A., Plant, K. L., Roberts, A. P., Harvey, C. & Thomas, T. G. (2016). Extending helicopter operations to meet future integrated transportation needs. *Applied ergonomics*, 53, 364-373.
- Stata. (2020). *Your data tell a story*. Verfügbar unter: <https://www.stata.com/>.
- Stein, O. (2001). Ein Schnellkurs in Matlab. Verfügbar unter: <http://www.mathematik.uni-kassel.de/~birken/matlab.pdf>.
- Stromnetze. (2017). Automatisierte Netzausbauplanung im Verteilnetz.
- Sullivan, B. (2016). *Mapping global fishing activity with machine learning*. Verfügbar unter: <https://blog.google/products/maps/mapping-global-fishing-activity-machine-learning/>.
- Sultanow, E. (2017). Digitalization from an architectural perspective. Verfügbar unter: <https://www.capgemini.com/de-de/wp-content/uploads/sites/5/2018/06/PoV-Digitalization-from-an-architectural-perspective.pdf>.
- Sultanow, E., von Tucher, C., Biedulski, A., Lesny, J. & Hanser, M. (2017). Eine Referenzarchitektur für die Digitalisierung des Energie-Sektors Teil 2: Modellierung der fachlichen Domänen. *NEWSolutions*, 2.
- Swan, J., Özcan, E. & Kendall, G. (2011). *Hyperion—a recursive hyper-heuristic framework*. Paper presented at the International Conference on Learning and Intelligent Optimization.
- Syte. (2020). *The Leader in Visual AI for Retail*. Verfügbar unter: https://www.syte.ai/?utm_source=Google&utm_medium=Search&utm_campaign=Visual%20Search&utm_term=Retail%20AI&utm_content=automated-search-via-image&gclid=CjwKCAiAqgTuBRBAEiwA7B66hRxNPGnyMNIBOsNdIKK1QDTARc8ezmKaHcWclo6DWLa8n5SB318NWRoCTqkQAvD_BwE.
- Tableau Software LLC. (2020a). *Fragen mit Tableau Desktop gedankenschnell beantworten*. Verfügbar unter: <https://www.tableau.com/de-de/products/desktop>.
- Tableau Software LLC. (2020b). *Kontrollierte Selfservice-Analytics im großen Stil mit Tableau Server*. Verfügbar unter: <https://www.tableau.com/de-de/products/server>.
- Tableau Software LLC. (2020c). *Machen Sie Ihre Daten mit Tableau Mobile überall zugänglich*. Verfügbar unter: <https://www.tableau.com/de-de/products/mobile>.
- Tableau Software LLC. (2020d). *Mit Tableau Prep können Sie Ihre Daten für die Analyse kombinieren, formatieren und aufbereiten*. Verfügbar unter: <https://www.tableau.com/de-de/products/prep>.
- Teetor, P. (2011). *R cookbook: Proven recipes for data analysis, statistics, and graphics*: " O'Reilly Media, Inc."
- Teradata. (2020a). *Teradata documentation*. Verfügbar unter: https://docs.teradata.com/reader/WJbR7YRqjgLiL9fZwxIBOG/OLijNkYzfC0UCck~KyP_Rw.
- Teradata. (2020b). *Teradata Machine Learning Engine*. Verfügbar unter: https://docs.teradata.com/reader/N06ooHwOGcycpbHnjyA_rw/Gzvt12ErfMJ36qoR0IkLfw.

- TFlearn. (2020). TFlearn: Deep learning library featuring a higher-level API for TensorFlow.
- The R Foundation. (2020). *What is R?* Verfügbar unter: <https://www.r-project.org/about.html>.
- Tibco. (2020a). *Machen Sie KI zur Realität: Operationalisierung von Data Science*. Verfügbar unter: <https://www.tibco.com/de/products/data-science>.
- Tibco. (2020b). *Tibco products*. Verfügbar unter: <https://www.tibco.com/de/products>.
- Tibco. (2020c). *TIBCO Spotfire: Augment Intelligence für Ihr Unternehmen*. Verfügbar unter: <https://www.tibco.com/de/products/tibco-spotfire>.
- Togaware. (2019). *Rattle: A Graphical User Interface for Data Mining using R*. Verfügbar unter: <https://rattle.togaware.com/>.
- Tönnis, M. (2010). *Augmented reality: Einblicke in die erweiterte Realität*: Springer-Verlag.
- Tribuo. (2020). *Interoperability*. Verfügbar unter: <https://tribuo.org/learn/4.0/docs/features.html#interoperability>.
- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236), 433.
- Tziridis, K., Kalampokas, T., Papakostas, G. A. & Diamantaras, K. I. (2017, 8/2017). *Airfare prices prediction using machine learning techniques*. Paper presented at the 2017 25th European Signal Processing Conference (EUSIPCO).
- U.S. Department of Veterans Affairs. (2020). *VA Technical Reference Model v 20.8: SAS Base*. Verfügbar unter: <https://www.oit.va.gov/Services/TRM/ToolPage.aspx?tid=6195>.
- Urra, E., Cubillos, C., Cabrera-Paniagua, D. & Mellado, R. (2019). hMod: A software framework for assembling highly detailed heuristics algorithms. *Software: Practice and Experience*, 49(6), 971-994.
- Van Onsem, W. & Demoen, B. (2013). *Parhyflex: A framework for parallel hyper-heuristics*. Paper presented at the Proceedings of the 25th Benelux Conference on Artificial Intelligence.
- VDO. (2020). *TIS-WEB® -TACHOGRAPHENAUSWERTUNG FÜR UNTERNEHMER*. Verfügbar unter: <https://www.fleet.vdo.de/produkt-finder/flottemanagement/tis-web/>.
- Ventura, S., Romero, C., Zafra, A., Delgado, J. A. & Hervás, C. (2008). JCLEC: a Java framework for evolutionary computation. *Soft computing*, 12(4), 381-392.
- Volvo. (2020). *Volvo holt „Skype for Business“ an Bord*. Verfügbar unter: <https://www.volvocars-partner.ch/blog/2017/01/30/volvo-holt-skype-for-business-an-bord/>.
- Walker, J. (2019). *AI in Food Processing – Use Cases and Applications That Matter*. Verfügbar unter: <https://emerj.com/ai-sector-overviews/ai-in-food-processing/>.
- Wall, M. (1996). GAlib: A C++ library of genetic algorithm components. *Mechanical Engineering Department, Massachusetts Institute of Technology*, 87, 54.
- Williams, G. (2011a). *Data mining with Rattle and R: The art of excavating data for knowledge discovery*: Springer Science & Business Media.
- Williams, G. (2011b). *Rattle: A Graphical User Interface for Data Mining using R*. Verfügbar unter: <https://rattle.togaware.com/>.

- Winter, J. (2018). Künstliche Intelligenz und datenbasierte Geschäftsmodellinnovationen–Warum Unternehmen jetzt handeln sollten. In *Service Business Development* (S. 61-79): Springer.
- WS02 Inc. (2020a). *Introducing Stream Processor*. Verfügbar unter: <https://docs.wso2.com/display/SP440/Introducing+Stream+Processor>.
- WS02 Inc. (2020b). *Quick Start Guide*. Verfügbar unter: <https://docs.wso2.com/display/SP400/Quick+Start+Guide>.
- Wu, Q., Ding, G., Xu, Y., Feng, S., Du, Z., Wang, J. et al. (2014). Cognitive internet of things: a new paradigm beyond connection. *IEEE Internet of Things Journal*, 1(2), 129-143.
- Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H. et al. (2008). Top 10 algorithms in data mining. *Knowledge and information systems*, 14(1), 1-37.
- Xu, L., Hutter, F., Hoos, H. H. & Leyton-Brown, K. (2008). SATzilla: portfolio-based algorithm selection for SAT. *Journal of artificial intelligence research*, 32, 565-606.
- Yolmo, Y. R. (2020). Airport Authority Of India To Introduce AI-Powered Baggage Screening, *Analytics India Magazine*.