

Knut Hinkelmann
Thomas Hoppe
Bernhard G. Humm *Hrsg.*

Hybride KI mit Machine Learning und Knowledge Graphs

Innovative Lösungen aus der Praxis

OPEN ACCESS

 Springer Vieweg

Hybride KI mit Machine Learning und Knowledge Graphs

Knut Hinkelmann · Thomas Hoppe ·
Bernhard G. Humm
(Hrsg.)


Hybride KI mit Machine Learning und Knowledge Graphs

Innovative Lösungen aus der Praxis

Hrsg.

Knut Hinkelmann 
Fachhochschule Nordwestschweiz FHNW
Olten, Schweiz

Thomas Hoppe 
Fraunhofer-Institut FOKUS
Berlin, Deutschland

Bernhard G. Humm 
Hochschule Darmstadt
Darmstadt, Hessen, Deutschland



ISBN 978-3-658-44780-9 ISBN 978-3-658-44781-6 (eBook)
<https://doi.org/10.1007/978-3-658-44781-6>

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

Hochschule DarmstadtJohan Wolfgang Goethe-UniversitätTechnische Hochschule MittelhessenFraunhofer-Institut FOKUS

© Der/die Herausgeber bzw. der/die Autor(en) 2025. Dieses Buch ist eine Open-Access-Publikation.

Open Access Dieses Buch wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor*in(nen) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Buch enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des/der betreffenden Rechteinhaber*in einzuholen.

Die Wiedergabe von allgemein beschreibenden Bezeichnungen, Marken, Unternehmensnamen etc. in diesem Werk bedeutet nicht, dass diese frei durch jede Person benutzt werden dürfen. Die Berechtigung zur Benutzung unterliegt, auch ohne gesonderten Hinweis hierzu, den Regeln des Markenrechts. Die Rechte des/der jeweiligen Zeicheninhaber*in sind zu beachten.

Der Verlag, die Autor*innen und die Herausgeber*innen gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag noch die Autor*innen oder die Herausgeber*innen übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Planung/Lektorat: David Imgrund

Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft Springer Fachmedien Wiesbaden GmbH und ist ein Teil von Springer Nature.

Die Anschrift der Gesellschaft ist: Abraham-Lincoln-Str. 46, 65189 Wiesbaden, Germany

Wenn Sie dieses Produkt entsorgen, geben Sie das Papier bitte zum Recycling.

Geleitwort

Die Geschichte der künstlichen Intelligenz in den letzten fast 70 Jahren lässt sich vielleicht am besten als Pendeln zwischen zwei Denkschulen beschreiben: den datengetriebenen Ansätzen, die lernende Systeme entwickeln, und den wissensbasierten Ansätzen, die schlussfolgernde Systeme aufbauen. Manchmal hatte einer dieser beiden Ansätze die Oberhand und sowohl Forscher als auch die Industrie investierten massiv in ihn, bis die Grenzen dieses Ansatzes deutlich wurden und sie alle ihre Hoffnung auf den anderen Ansatz setzten, bis das Gegenteil eintrat – und so weiter. Seit den späten 1950er-Jahren schwingt dieses Pendel etwa in jedem Jahrzehnt zwischen den datengetriebenen und den wissensbasierten KI-Methoden hin und her. Das letzte Jahrzehnt wurde zweifellos von den datengetriebenen maschinellen Lernsystemen dominiert, mit dem jüngsten Aufkommen der „generativen KI“ in Form von Chatbots wie ChatGPT und Bilderzeugungsprogrammen wie DALL-E.

Alternativ wurde die Geschichte der KI manchmal als Achterbahnfahrt beschrieben, mit sehr steilen Anstiegen auf einen Höhepunkt der Begeisterung, gefolgt von einer sehr beängstigenden Fahrt hinunter in ein Tal der Desillusionierung. Dieses Buch erscheint sicherlich zu einer Zeit, in der die Achterbahnfahrt der KI einen Höhepunkt erreicht hat. Die Erfolge des maschinellen Lernens sind unübersehbar, sowohl in der Gesellschaft als auch in der KI-Disziplin selbst. Gleichzeitig werden aber auch die Grenzen rein datengetriebener Ansätze immer deutlicher. Ihre Qualität ist schwer vorhersehbar, ihre Ergebnisse sind schwer zu erklären und ihr Training ist kostspielig, sowohl was die Daten als auch die benötigte Energie angeht.

Daher erscheint dieses Buch genau zum richtigen Zeitpunkt: Es argumentiert überzeugend, dass der Weg in die Zukunft der KI-Forschung und -Anwendungen nicht an einem der beiden Enden des schwingenden Pendels zu finden ist, sondern eher in der Mitte, in Systemen, die Lernen und logisches Denken kombinieren, den sogenannten hybriden KI-Systemen. Die Entwicklungen bei wissensbasierten Systemen haben es ermöglicht, dass diese nun mit lernenden Systemen kombiniert werden können: Wissensbasierte Systeme sind nicht mehr auf kleine Wissensbasen in handverlesenen Domänen

beschränkt, sondern verwenden heute routinemäßig sehr große Wissensgraphen, die viele Millionen von Entitäten und Beziehungen zwischen ihnen umfassen und durch ein Vokabular von Hunderttausenden von Begriffen beschrieben werden.

Obwohl das Feld der hybriden KI (oder „neurosymbolischen Systeme“, wie sie manchmal genannt werden) bis in die 1990er-Jahre zurückreicht (wenn nicht sogar noch früher), erlebt es derzeit eine Explosion des Interesses. Jede Woche erscheinen neue Vorschläge für Algorithmen und Architekturen und das Feld befindet sich in einem Zustand des starken Wandels. Daher ist es wichtig, dass der vorliegende Band eine Reihe von Möglichkeiten bietet, dieses junge Gebiet klar zu organisieren. Die Kapitel des Buches sind in vier Gruppen gegliedert: Systeme, bei denen das Lernen dem Schlussfolgern hilft, Systeme, bei denen das Schlussfolgern dem Lernen hilft, Systeme, bei denen Lernen und Schlussfolgern eng gekoppelt sind, und Systeme, bei denen sie nur lose gekoppelt sind. Die Kapitel des Buches decken ein weites Feld ab, mit unterschiedlichen Kombinationen von KI-Technologien, KI-Anwendungen, gesellschaftlichen Bereichen und Datentypen.

Viele Aspekte der Kombination von maschinellem Lernen und Wissensverarbeitung werden diskutiert: die Notwendigkeit der Erklärbarkeit, die Notwendigkeit, mit weniger Daten zu trainieren, die Notwendigkeit, KI-Systeme energieeffizienter zu machen, die Notwendigkeit, ein System schnell an einen neuen Bereich anzupassen.

Wie gesagt, die hybriden/neurosymbolischen Methoden stehen noch am Anfang. Daher ist es sehr ermutigend zu sehen, dass das Buch eine große Anzahl von Anwendungen hybrider KI versammelt: medizinische Forschung, Dokumentenklassifikation, Unterstützung von Verwaltungsaufgaben des medizinischen Personals, Knowledge Engineering, semantische Suche, Entscheidungshilfe für medizinisches Personal, juristische Entscheidungshilfe und sogar die Vorhersage des Ausgangs von Fußballspielen.

Ich freue mich sehr (und fühle mich geehrt), dass die Autoren die Methodik und Notation für hybride KI-Systeme gewählt haben, die wir zuvor vorgeschlagen haben, um die verschiedenen in diesem Band beschriebenen Systeme und Anwendungen zu systematisieren und zu verdeutlichen. Zusammen mit der gut strukturierten Gliederung des Buches wird dies sicherlich dazu beitragen, diesen neuen Bereich der KI zu verdeutlichen.

Mit seiner sehr klaren Struktur und seiner Fülle an Material steht dieses Buch im Mittelpunkt der aktuellen Debatten in der KI. Vielleicht gelingt es uns dieses Mal, das Pendel in der Mitte zu stoppen!

Vorwort der Herausgeber

Intelligente Anwendungen umsetzen, ist das Ziel der Teilnehmenden der Workshops zu „Applied Machine Intelligence“, die seit 2015 im Leibniz-Zentrum für Informatik Schloss Dagstuhl stattfinden, ursprünglich unter dem Namen „Corporate Semantic Web“. Aus diesen sind bereits drei Bücher^{1 2 3} und sechs Journal-Artikel^{4 5 6 7 8 9} hervorgegangen.

¹ Ege B, Humm B, Reibold A (eds) (2015) Corporate Semantic Web – Wie Anwendungen in Unternehmen Nutzen stiften. Springer, Heidelberg. ISBN 978-3-642-54885.

² Hoppe T, Humm B, Reibold A (eds) (2018) Semantic applications – methodology, technology, corporate use. Springer, Berlin. ISBN 978-3-662-55432.

³ Hoppe T (2020) Semantische Suche – Grundlagen und Methoden semantischer Suche von Textdokumenten. Springer Vieweg, Wiesbaden. ISBN 978-3-658-30426.

⁴ Humm BG, Archer P, Bense H, Bernier C, Goetz G, Hoppe T, Schumann F, Siegel M, Wenning R, Zender A: New directions for applied knowledge-based AI and machine learning. Selected results of the 2022 Dagstuhl Workshop on Applied Machine Intelligence. Informatik Spektrum 46:65–78. <https://doi.org/10.1007/s00287-022-01513-9>.

⁵ Humm BG, Bense H, Fuchs M, Gernhardt B, Hemmje M, Hoppe T, Kaupp L, Lothary S, Schäfer KU, Thull B, Vogel T, Wenning R (2021) Machine intelligence today: applications, methodology, and technology. Informatik Spektrum 44:104–114. <https://doi.org/10.1007/s00287-021-01343-1>.

⁶ Bense H, Gernhardt B, Haase P, Hoppe T, Hemmje M, Humm B, Paschke A, Schade U, Schäfermeier R, Schmidt M, Siegel M, Vogel T, Wenning R (2016) Emerging trends in corporate semantic web – selected results of the 2016 Dagstuhl workshop on corporate semantic web. Informatik Spektrum 39(6):474–480.

⁷ Hoppe T, Humm B, Schade U, Heuss T, Hemmje M, Vogel T, Gernhardt B (2015) Corporate semantic web – applications, technology, methodology. Informatik Spektrum 39(1):57–63. <https://doi.org/10.1007/s00287-015-0939-0>.

⁸ Busse J, Humm B, Lübbert C, Moelter F, Reibold A, Rewald M, Schlüter V, Seiler B, Tegtmeier E, Zeh T (2015) Actually, what does “Ontology” mean? A term coined by philosophy in the light of different scientific disciplines. J Comput Inform Technol 23(1):29–41. <https://doi.org/10.2498/cit.1002508>.

⁹ Humm BG, Bense H, Classen M, Geißler S, Hoppe T, Juwig O, Paschke A, Schäfermeier R, Siegel M, Weichardt F, Wenning R (2019) Current trends in applied machine intelligence. Informatik Spektrum 42(1):28–37. <https://doi.org/10.1007/s00287-018-01127-0>.

Die Idee zu diesem Buch entstand auf dem Workshop „Applied Machine Intelligence 2022“. Mit der Veröffentlichung von ChatGPT und damit durchgeführten Experimenten wurde offensichtlich, dass ein alleiniges Trainieren von Large Language Models (LLM) ohne zusätzliches Wissen über die Welt nicht ausreichend sein wird, um wirklich intelligente Anwendungen umzusetzen. Aus dem Austausch von Praxiserfahrungen und intensiven Diskussionen wurde uns immer klarer, dass es der Kombination von maschinellen Lernverfahren und wissensbasierter KI (künstliche Intelligenz) bedarf, um Defizite der jeweiligen einzelnen Verfahren zu überwinden. Und unsere Erfahrung zeigte, dass hybride KI-Anwendungen bereits in der Praxis angekommen waren. Wir wollten unsere Erfahrungen mit der Entwicklung solcher Anwendungen teilen und so wurde die Idee für das Buchprojekt geboren.

Relativ zügig haben wir ausreichend Interessenten gewonnen, um Buchkapitel beizusteuern. Aber einfach nur einen Sammelband erstellen wollten wir nicht. So trafen wir uns im März 2023 zu einem zweitägigen Autorenworkshop, um unsere Inhalte abzugleichen und eine gemeinsame Sprache zu finden, um über hybride KI zu sprechen. Ausdruck davon sind beispielsweise die Boxology-Diagramme, die sich in jedem Buchkapitel finden.

Wir hoffen mit der Zusammenstellung und Beschreibung der unterschiedlichen Anwendungsgebiete den Lesenden, seien es Lernende, Lehrende, Entwickelnde, Beratende oder Entscheidende,¹⁰ einen Einblick in die Möglichkeiten und einen Anreiz für die Übertragung und Umsetzung hybrider KI-Verfahren in ihrem Arbeitsbereich zu geben.

Wenn die Lesenden mit der Lektüre des Buchs neue Erkenntnisse gewinnen oder die Erfahrungen der Autoren für ihre eigenen Arbeiten erfolgreich nutzen können, dann haben wir unser Ziel des Wissenstransfers erreicht. Ohne die Arbeiten der Autoren und ihre Bereitschaft, einen Beitrag zu diesem Buch zu leisten, wäre dieses Buch nicht zustande gekommen. Daher möchten wir uns für ihre tatkräftige Unterstützung bei der Erstellung, Korrektur und Vorbereitung der Inhalte bedanken. Ebenso möchten wir Frank van Harmelen für die Bereitschaft danken für die Formulierung eines Geleitworts für dieses Buch.

Unser Dank geht auch an den Springer-Verlag, seine Mitarbeitenden und insbesondere an David Imgrund für die langjährige Zusammenarbeit bei der Publikation mehrerer Bücher. Die gemeinsame Zusammenarbeit war immer sehr angenehm, effizient und effektiv.

Olten, Berlin, Darmstadt
Oktober 2023

Knut Hinkelmann
Thomas Hoppe
Bernhard G. Humm

¹⁰ Eine Anmerkung zu den gendergerechten Schreibweisen: Sprachgebrauch lässt sich nicht vorschreiben. Die gelebte, natürliche Sprache unterliegt ihrer eigenen Evolution; daher überlassen wir es den Schreibenden, in welcher Form sie zu Gendergerechtigkeit beitragen wollen.

Inhaltsverzeichnis

1	Einführung in hybride künstliche Intelligenz	1
	Knut Hinkelmann, Thomas Hoppe und Bernhard G. Humm	
Teil I Machine Learning für wissensbasierte KI		
2	Unterstützung klinischer Studien mit hybrider KI	25
	Silvio Peikert, Naouel Karam, Ralph Schäfermeier, Le Duyen Sandra Vu, Johannes Jung, Şefika Efeoğlu, Jamal Al Qundus und Adrian Paschke	
3	Retrospektives Klassifizieren und Annotieren von Geschäftsdokumenten im Tagesgeschäft der Denkmalbehörden	47
	Christian Fillies, Andreas Giesenhausen und Frauke Weichhardt	
4	Text-getriebener Aufbau von domänenspezifischen Wissensgraphen mit neuronalen Netzen	67
	Felix Hamann	
5	Das Projekt OdeNet: Aufbau eines semantischen Netzes für die deutsche Sprache	97
	Melanie Siegel	
Teil II Wissensbasierte KI für Machine Learning		
6	Integration von terminologischen Wissen in eine Word-Embedding-basierende Semantische Suche	125
	Thomas Hoppe	
7	Vorhersage von landwirtschaftlichen Erträgen und Wachstum	153
	Alexander Münzberg und Miro Miranda Lorenz	
8	Ontologie-basiertes AutoML	169
	Alexander Zender und Bernhard G. Humm	

Teil III Inhärent hybride KI-Verfahren

- 9 Ermittlung elektiver Risikopatienten mittels Bayes'scher Netze** 197
 Thomas Hoppe, Isabel Mattig, Gina Barzen, Shirkouh Matini,
 Kerstin Köhler, Henryk Dreger und Friedrich Köhler
- 10 Transfer-Lernen für die Klassifikation medizinischer Texte** 223
 Daniel Bruneß, Matthias Bay, Christian Schulze, Michael Guckert
 und Mirjam Minor
- 11 Anomalie-Detektion in der verarbeitenden Industrie** 243
 Christian Götz und Bernhard G. Humm
- 12 Vorhersage von Sportergebnissen mittels probabilistischer
 Programmierung.** 271
 Christian Herta und Thomas Hoppe

Teil IV Kombination von Machine Learning und wissensbasierter KI

- 13 Hybrides wissensbasiertes Reasoning Für wissensintensive
 Prozesse am Beispiel von Notrufabfragen** 297
 Eric Rietzke, Carsten Maletzki, Lisa Grumbach und Ralph Bergmann
- 14 Optimierung der Entscheidungsfindung in autonomen Fahrsystemen
 mit neuro-symbolischem Wissen** 319
 Ya Wang, Kumar Manas und Adrian Paschke
- 15 Wissensgraphen und maschinelles Lernen im Spannungsfeld
 juristischer Sprache** 349
 Christian Dirschl und Christian Hartz
- 16 Kosteneffiziente Rekognition durch Out-of-the-Box KI und Semantik.** . . . 375
 Achim Reiz und Kurt Sandkuhl
- 17 Interaktive Entscheidungsfindung unter Einsatz von maschinellem
 Lernen und Regelbasierten Systemen.** 401
 Knut Hinkelmann, Mengyang Li, Maja Spahic-Bogdanovic
 und Hans Friedrich Witschel
- Stichwortverzeichnis** 419



Einführung in hybride künstliche Intelligenz

1

Knut Hinkelmann , Thomas Hoppe  und Bernhard G. Humm 

Kernaussagen

1. Machine Learning (ML) hat seit dem Beginn des „digitalen Zeitalters“ um die Jahrtausendwende beachtliche Erfolge erzielt.
2. Künstliche Intelligenz (KI) wird mittlerweile mit ML gleichgesetzt, auch wenn KI viel mehr umfasst.
3. Experten im Bereich des Deep Learning erkennen, dass ihre Modelle an Grenzen stoßen und neben den eigentlichen Daten auch zusätzliches Wissen über die Welt, logische Schlüsse und ethisches Handeln berücksichtigen müssen.
4. Hybride Ansätze, die entweder ML-Verfahren mit Hintergrundwissen ausstatten, semantische Wissensrepräsentations- und Inferenzverfahren um Lernkomponenten erweitern oder Verfahren aus beiden KI-Richtungen kombinieren, befinden sich bereits in der Anwendung resp. der anwendungsnahen Entwicklung.

K. Hinkelmann
Fachhochschule Nordwestschweiz FHNW, Olten, Schweiz
E-Mail: knut.hinkelmann@fhnw.ch

T. Hoppe (✉)
Fraunhofer-Institut Fokus, Berlin, Deutschland
E-Mail: thomas.hoppe@fokus.fraunhofer.de

B. G. Humm
Hochschule Darmstadt, Darmstadt, Deutschland
E-Mail: bernhard.humm@h-da.de

1.1 Einleitung¹

Künstliche Intelligenz hat als eigenes Forschungsgebiet ihren Ursprung in der Dartmouth-Konferenz von 1956. Dort haben sich Computer- und Kognitionswissenschaftler, Mathematiker, Psychologen und Ingenieure zum ersten Mal getroffen, um sich der Lösung der Frage zu widmen, ob Computer so programmiert werden können, dass sie intelligentes Verhalten simulieren. Bereits zum damaligen Zeitpunkt waren einige der Wissenschaftler an deduktiven Inferenzsystemen, andere an induktivem Machine Learning (ML) interessiert.¹

In der Folge der Konferenz wurden in den 1960er-Jahren einige erste Erfolge erzielt. Es wurden aber auch einige Rückschläge – die insbesondere durch die Komplexität der Fragestellung bedingt waren – verzeichnet. Bis zum Ende der 1990er-Jahre bildete die symbolische künstliche Intelligenz, die sich um Fragen der Wissensrepräsentation und logisch deduktiver Schlüsse drehte, den Schwerpunkt der Arbeiten, nicht zuletzt bedingt durch die geringe Menge an verfügbaren digitalen Daten.

Zu diesem Bereich der KI ist auch die Semantic-Web-Initiative von Tim-Berners Lee zu rechnen, der die World-Wide-Web-Protokolle primär für das Wissensmanagement entwickelte und das enorme Potenzial des sich entwickelnden WWWs und die darin verfügbaren Daten erkannte.

Die Jahrtausendwende stellte einen entscheidenden Wendepunkt in der Entwicklung von ML-Verfahren dar, da in den frühen 2000er-Jahren das „digitale Zeitalter“ begann, in dem die Menge der digitalen Daten die Menge der analog gespeicherten Daten überschritt (Hilbert und López 2011).² Gleichzeitig wurden durch das Internet die verteilte Speicherung und die Sammlung von Daten auf der Basis leistungsstarker Server-Hardware möglich.

Anfang der 2000er-Jahre gelangen Forschern im Bereich des Machine Learning mit neuronalen Netzen entscheidende Durchbrüche – insbesondere durch die Grundlagenforschung von Geoffrey Hinton, Yann LeCun und Yoshua Bengio zu Backpropagation, Convolutional Neural Networks, Recurrent Neural Networks und Natural Language Processing. Auch standen nun, insbesondere durch das WWW, große Mengen von – durch Benutzer gelabelten – Bilddaten, umfangreiche Textkorpora und Sprachdaten zur Verfügung. Bilderkennung, Spracherkennung und -verarbeitung standen einer größeren Öffentlichkeit zur Verfügung, sodass diese Form der KI stärker in den Fokus des öffentlichen Interesses rückte.

Parallel dazu entwickelten sich semantische Anwendungen in einem ganz anderen Bereich weiter, als ursprünglich von Tim-Berners Lee mit der Semantic-Web-Initiative

¹Eine weitergehende Darstellung der Entwicklung der KI findet sich u. a. in „A brief history of artificial intelligence: on the past, present, and future of artificial intelligence“, https://www.researchgate.net/publication/334539401_A_Brief_History_of_Artificial_Intelligence_On_the_Past_Present_and_Future_of_Artificial_Intelligence.

²<https://www.martinhilbert.net/worldinfocapacity-html/>, letzter Zugriff 18.05.2023.

vorgedacht. Semantische Anwendungen auf Basis expliziter Wissensrepräsentation wurden insbesondere in Unternehmenskontexten aufgegriffen. Einerseits, um Anwendungen im Unternehmenskontext mit umfangreichem terminologischen Wissen in Form von Thesauri und Ontologien auszustatten (Ege et al. 2015; Hoppe et al. 2018), andererseits wurden umfangreiche Wissensgraphen³ durch die Übernahme von Freebase durch Google (Ihlenfeld 2010), die Entwicklung des Google Knowledge Graph und die Übertragung von Freebase an Wikidata der Wikimedia Foundation (Klein 2014) geschaffen.

Mittlerweile existiert eine große Anzahl fachspezifischer Ontologien in den unterschiedlichsten Gebieten,⁴ die nicht aus rein wissenschaftlichem Interesse geschaffen wurden, sondern um einen Bedarf an kuratiertem Wissen zu decken. Dass der Einsatz symbolischer Verfahren bisher auf keine große öffentliche Resonanz gestoßen ist, liegt vermutlich daran, dass sie in eher unspektakulären Fachanwendungen, quasi unter der Motorhaube, eingesetzt werden.

Durch die Entwicklung generativer neuronaler Netze (GNN) zur Erzeugung irrealer, künstlicher oder fotorealistischer Bilder (Dall-E, Midjourney, Stable Diffusion) und von Transformer-Architekturen zur Generierung sprachlich kohärenter längerer Texte (Galactica, ChatGPT, AutoGPT) auf der Basis großer Sprachmodelle (wie Bert, RoBERTa, GPT, GPT-3, GPT-X, GPT-4, LaMDA, PaLM, BARD) erweckte die ML-Seite der KI in der breiten Öffentlichkeit nicht nur enormes Interesse, sondern auch heftige Diskussionen hinsichtlich Urheberrechtsfragen, Plagiaten, Betrugs in Lehre und Wissenschaft, politischer Einflussnahme auf Menschen, Verlusts von Arbeitsplätzen, ethisch/moralischer Entscheidungs- und Handlungsfragen bis hin zur Furcht über die vermeintlich absehbare Übernahme der Weltherrschaft durch eine dem Menschen weit überlegene KI.

Aus wissenschaftlicher Perspektive betrachtet zeigte sich jedoch schnell, dass, so bemerkenswert die Ergebnisse dieser Systeme auch sind, sie immer noch unter Kinderkrankheiten leiden. Generierte Bilder sind in den Details stellenweise inkohärent, was z. B. die Anzahl von Fingern oder die Anordnung von Gliedmaßen betrifft. Generierte Texte sind zwar sehr verständlich geschrieben und klingen oberflächlich gelesen plausibel. Wenn es aber um aktuelle Informationen und Fakten, logische Schlüsse, korrekte mathematische Berechnungen oder ein Verständnis menschlichen Handelns oder der Welt geht, wird bei genauerem Lesen der Texte klar, dass es diesen Systemen an Wissen fehlt, insbesondere über die Funktionsweise der Welt. Auch zeigen sich erste Hinweise, dass die Qualität sprachgenerierender Systeme im Lauf ihrer Nutzung zu degenerieren scheint. Derzeit wird noch spekuliert, ob dieser Effekt aus dem weiteren Lernen aus Nutzereingaben oder aus deren weiterem Training mit computergenerierten Texten entsteht.

Multimodale neuronale Netze, die sowohl über Manipulatoren (wie Roboterarme) und Sensoren (zur Bild- und Tonerkennung) verfügen, könnten zwar das Problem des „symbol groundings“, der Verknüpfung von Konzepten mit der realen Welt, durch

³Die genau genommen eigentlich als Faktengraphen bezeichnet werden müssten.

⁴<https://zenodo.org/record/7907025#.ZGciWs7P1PY3>.

aktives Experimentieren, Interaktion und Beobachtung mit der realen Welt zum Teil lösen. Dennoch bedarf es zusätzlicher Mechanismen, um diese entstehenden Systeme sowohl um logische als auch mathematische Schlüsse zu erweitern (Wolfram 2023). Wie aber die Imaginationsfähigkeit und Kreativität zur Invention neuer abstrakter Konzepte und zum Verständnis und Nachvollzug menschlichen Handelns und menschlicher Gedankengänge umgesetzt werden können, ist noch offen.

Interessanterweise weisen selbst KI-Forscher und Entwickler generativer KI-Systeme auf deren derzeitige Mängel und Einschränkungen hin und geben Hinweise, wie diese überwunden werden können.⁵

- Rodney Brooks argumentiert, dass die größten Erfolge der KI meist auf einfachen Entscheidungen basieren, deren Fehlentscheidungen keine großen Konsequenzen nach sich ziehen oder in deren Prozess irgendwo ein Mensch als Human in the Loop involviert ist (Brooks 2021).
- Yann LeCunn vertritt die Ansicht, dass künstliche neuronale Netze (NN) durch Skalierung doch „keine Intelligenz“ produzieren und Weltwissen in Form eines Weltmodells benötigt wird (Lu 2018).
- Dall-E kann nicht korrekt zählen und hat Probleme mit Konzepten, insbesondere mit Negation (Schreiner 2022).
- Nach dem Aufruf von ChatGPT wird explizit darauf hingewiesen, dass teilweise falsche, schädliche oder voreingenommene Informationen generiert werden und dass das System über beschränkte Informationen verfügt, die nur bis 2021 reichen.
- Dass den „Large Language Models“ ein gewisser Grad an Voreingenommenheit (Bias) inhärent ist, ergibt sich bereits aus der Übersicht über die Korpora, die lediglich aus frei verfügbaren Internetquellen stammen, mit denen sie trainiert werden (Schaul et al. 2023).
- Geoffrey Hinton sieht eine große Gefahr darin, dass diese Modelle eine neue Form von „Intelligenz“ verkörpern, die sich von der des Menschen gravierend unterscheidet. Während die Intelligenz von Menschen individuell ist und Wissen nur partiell und bilateral zwischen Menschen geteilt werden kann, verwenden diese digitalen Systeme in vielen unterschiedlichen Instanzen das gleiche Modell und gleiche Gewichtungen. Alle diese Kopien können separat lernen, teilen aber neu erworbenes Wissen umgehend, so dass sie mehr als jeder einzelne Mensch wissen können (BBC 2023).
- Dough Lenat und Gary Marcus stellen 16 Wünsche an zukünftige, vertrauenswürdige KI auf und diskutieren einen alternativen Ansatz, der theoretisch viele der derzeitigen Beschränkungen von LLMs adressieren könnte: „eine KI, die basiert auf kuratierten

⁵Links auf weitere aktuelle Hinweise darauf, dass andere NN-Forscher auf die Notwendigkeit von hybriden Methoden hinweisen, finden sich in https://www.linkedin.com/posts/petedietert-91750a87_ai-ai-activity-7001732471054643202-tEPI?utm_source=share&utm_medium=member_ios.

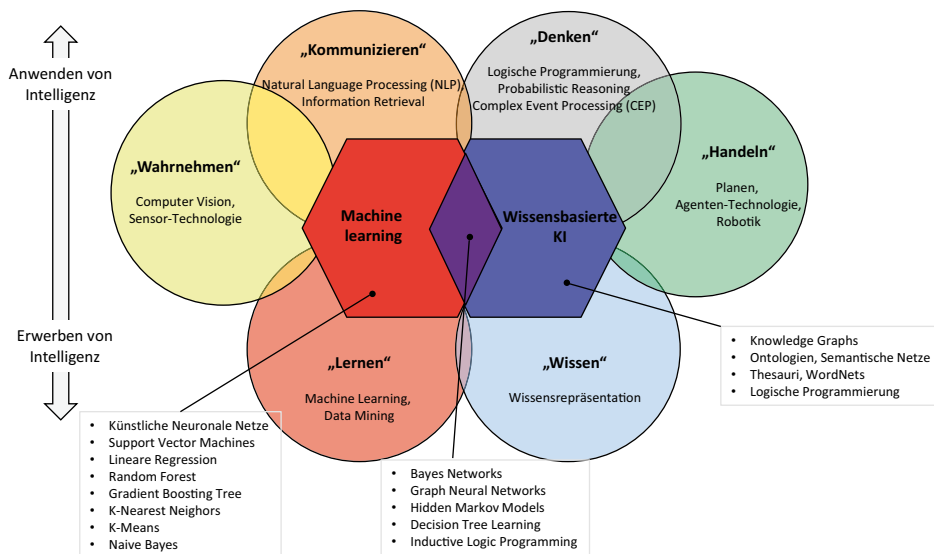


Abb. 1.1 Landkarte von KI (nach Humm et al. 2022, basierend auf einem KI-Spektrum-Poster⁶)

Teilen expliziten Wissens und Faustregeln, es einer Inferenzmaschine ermöglicht, automatisch die logischen Konsequenzen all dieses Wissens abzuleiten“ (Lenat und Marcus 2023).

- Steven Wolfram (2023) schlägt vor, generative KI-Systeme mit einem Subsystem zu kombinieren, das auf kuratiertem Wissen basiert und in der Lage ist, korrekte logische Schlüsse und mathematische Ableitungen und Berechnungen durchzuführen. Indem subsymbolische, numerische Berechnungen mit symbolischem Weltwissen kombiniert werden, können einige der Defizite von Verfahren ausgeglichen werden, die allein auf Machine Learning basieren. Er weist damit in die Richtung von Anwendungen hybrider künstlicher Intelligenz.

Hybride KI umfasst aber mehr als nur Sprach- und Textgenerierungsmodelle. Im folgenden Abschnitt beschreiben wir, was wir darunter verstehen.

1.2 Hybride KI

Bevor wir unser Verständnis von hybrider KI erläutern, möchten wir zunächst ein gemeinsames Begriffsverständnis anhand einer Landkarte der KI bilden (Abb. 1.1).

⁶<https://www.ai-spektrum.de/fachinhalte/fachposter/landkarte-der-kuenstlichen-intelligenz>, zuletzt zugegriffen am 01.09.2023.

In der Landkarte werden die traditionellen KI-Teilgebiete wie Computer Vision, Natural Language Processing (NLP), Robotik etc. in Kreisen dargestellt, strukturiert nach den Eigenschaften menschlicher Intelligenz, welche sie nachahmen: Computer Vision ahmt Wahrnehmung nach, NLP Kommunizieren, Robotik Handeln usw. Im Zentrum der Abbildung werden in zwei Sechsecken die beiden wesentlichen Familien von KI-Verfahren dargestellt: a) Machine-Learning-Verfahren wie künstliche neuronale Netze, Support Vector Machines etc.; b) wissensbasierte KI-Verfahren wie Wissensgraphen, Ontologien, logische Programmierung etc. Bewusst sind die Kreise untereinander sowie die Sechsecke mit den Kreisen überlappend dargestellt, da die meisten KI-Verfahren auch in den meisten KI-Teilgebieten angewendet werden können. Die Schnittmenge der beiden Sechsecke bildet eine Raute. Sie stellt KI-Verfahren dar, welche sowohl Machine Learning als auch wissensbasierter KI zugeordnet werden können, also inhärent hybrid sind. Beispiele sind Bayes'sche Netzwerke, Graph Neural Network, Hidden Markov Models etc.

Bevor wir auf hybride Ansätze näher eingehen, möchten wir die Charakteristika der beiden wesentlichen Familien von KI-Verfahren näher betrachten.

ML-Verfahren benötigen Daten, meist viele und umfangreiche, um gut zu funktionieren. Beim überwachten Lernen müssen diese von menschlichen Experten annotiert werden (Labeling), z. B. bei der Vorhersage von Krankheiten wie Krebs aus medizinischen Bildern mit der Information, ob ein Bild einen Tumor darstellt oder nicht. Nach der Konfiguration von ML-Verfahren (Model Selection, Hyperparameter Tuning) generiert die Maschine ein ML-Modell (Trainingsphase), welches die Charakteristika der Daten in Form eines mathematischen Modells kondensiert, aber in der Regel für den menschlichen Experten nicht verständlich ist. Dieses ML-Modell kann in KI-Anwendungen eingebaut werden, z. B. um für neue medizinische Bilder eine Vorhersage über das Vorhandensein eines Tumors zu machen.

Wissensbasierte KI-Verfahren sind dazu konträr. Es werden keine Trainingsdaten benötigt. Stattdessen modellieren menschliche Experten Zusammenhänge einer Anwendungsdomäne wie Medizin mit einem Formalismus, wie z. B. einer Ontologiesprache (Knowledge Engineering). Dieses formalisierte Wissen kann von Menschen inspiziert und qualitätsgesichert werden – im Gegensatz zum ML-Modell. Aber genau wie das ML-Modell kann es in eine KI-Anwendung eingebaut werden, z. B. um Ärzte bei der Diagnose von Krankheiten zu unterstützen.

Betrachtet man die Vor- und Nachteile von ML und wissensbasierter KI, so sind diese komplementär (Abb. 1.2).

Nachteil wissensbasierter KI ist, dass das Knowledge Engineering ein zeitaufwendiger und teurer Prozess sein kann. Auch ist dies nur anwendbar in Bereichen, in denen Wissen explizit spezifiziert werden kann. Für Bildverarbeitung ist dies beispiels-

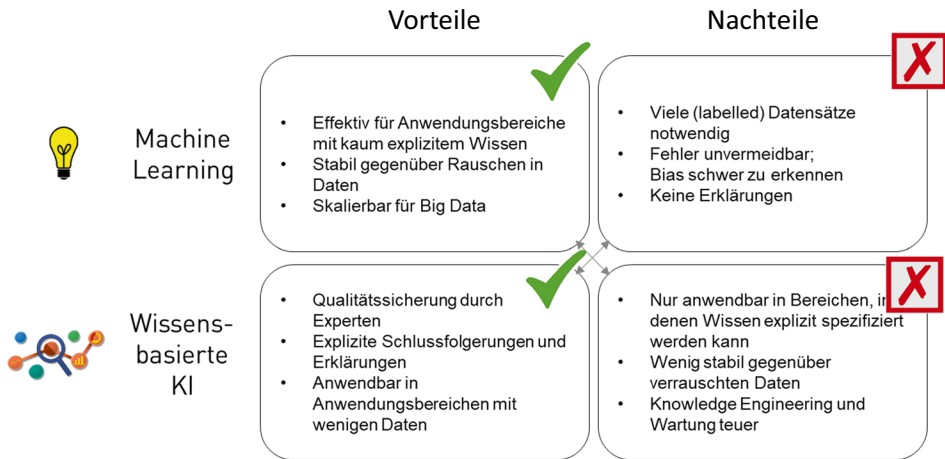


Abb. 1.2 Vor- und Nachteile von Machine Learning und wissensbasierter KI

weise gar nicht möglich. Auch sind wissensbasierte Verfahren wenig stabil gegenüber verrauschten Daten, wie dies bei Sensoren üblich ist.

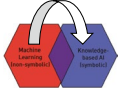
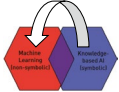
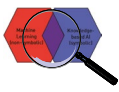

Hier genau liegen die Stärken von ML. ML-Modelle können ohne explizites Wissen trainiert werden und sie sind stabil gegenüber Rauschen und skalierbar für extrem große Datenmengen, wie große Sprachmodelle zeigen.

Allerdings ist eine der wesentlichen Schwächen von ML-Verfahren, dass sie immer fehlerbehaftet sind – man kann höchstens die Fehlerwahrscheinlichkeit mittels Metriken abschätzen. Die meisten ML-Verfahren sind per se nicht erklärbar und Verzerrungen (Bias) sind sehr schwer zu erkennen. Außerdem kann das Annotieren (Labeling) von großen Datenmengen kostspielig sein und für gute Resultate sind meist große Datenmengen vonnöten.

Dies wiederum sind genau die Stärken wissensbasierter KI: Sie sind per se erklärbar und können von Experten qualitätsgesichert werden. Darüber hinaus können sie in Domänen angewendet werden, in denen nur wenige Daten vorliegen.

Zusammengefasst kann man feststellen, dass sich die Stärken und Schwächen von ML und wissensbasierter KI komplementär verhalten. Die Kombination beider Ansätze bietet also die Chance, deren Vorteile zu kombinieren und Nachteile auszugleichen. Und genau das ist das Ziel von hybrider KI: die Kombination von ML und wissensbasierter KI.

Dabei unterscheiden wir vier Arten des Einsatzes von hybrider KI:

1.  **ML für wissensbasierte KI:** Hier geht es um den Einsatz von ML zur Verbesserung wissensbasierter KI, z. B. der Einsatz ML-basierter Textanalyse zum Aufbau von Knowledge Graphs.
2.  **Wissensbasierte KI für ML:** Hier geht es um den Einsatz von wissensbasierter KI für die Verbesserung von ML, z. B. die semantische Anreicherung von Trainingsdaten auf der Basis von Knowledge Graphs.
3.  **Inhärent hybride KI-Verfahren:** Hier geht es um den Einsatz von KI-Verfahren, die ML in Kombination von Daten und symbolischen Repräsentationen einsetzen, z. B. Bayes'sche Netze, Graph Neural Networks oder Conceptual Clustering.
4.  **Kombination von ML und wissensbasierter KI:** Hier geht es um den gleichberechtigten, kombinierten Einsatz beider Ansätze innerhalb einer Anwendung, z. B. beim autonomen Fahren wissensbasierte KI für die Formalisierung der Straßenverkehrsordnung und ML für die Verkehrserkennung.

Wir verwenden diese vier Arten des Einsatzes von hybrider KI zur Strukturierung des Buchs in vier Teile.

1.3 Anwendungen hybrider KI

In diesem Abschnitt behandeln wir Anwendungen hybrider KI, die in diesem Buch beschrieben werden, gruppiert nach den oben beschriebenen vier Arten.

1.3.1 ML für wissensbasierte KI

In Kap. 2 wird ein Anwendungsfall aus der Medizin beschrieben, speziell die Unterstützung klinischer Studien. Für Medizin und Pharmakologie existieren viele Ontologien, Terminologien und Datensätze, die zur Entwicklung genutzt werden können (wissensbasierte KI). Auf der anderen Seite unterstützen ML-Verfahren die Verarbeitung

großer Mengen medizinischer Daten. In dem Kapitel wird anschaulich gezeigt, wie hybride KI-Ansätze die Zuverlässigkeit modellgetriebener Verfahren mit Flexibilität von ML-Methoden kombinieren.

Können KI-Methoden Denkmalschützer bei ihren Analysen effektiv unterstützen? Dieser Frage gehen die Autoren von Kap. 3 nach. Sie verwenden NLP-Systeme (Natural Language Processing) zur Analyse von Bestandsdokumenten der Berliner Stadtbezirke. Durch die formale und fachliche Klassifikation von bestehenden Dokumenten mithilfe von ML in Kombination mit semantischen Technologien kann auf eine manuelle Klassifikation der Dokumente verzichtet werden. Eine wichtige Lehre, die die Autoren aus dem Projekt ziehen: Es lohnt sich, Prozesse, Dokumenttypen und Dokumentstrukturen daraufhin zu prüfen, welche Art von ML mit welchem Aufwand auf welche Weise am besten zum Ziel führt.

In Kap. 4 wird der interaktive Aufbau von Wissensgraphen mittels Deep Learning beschrieben. Die manuelle Erstellung von Wissensgraphen für spezifische Domänen wie die Versicherungsbranche ist ein kostenintensiver Prozess. Allerdings kann spezifisches Domänenwissen auch aus Textdokumenten extrahiert werden, die im operativen Geschäft anfallen. Im Kapitel wird dementsprechend ein Annotationsprozess vorgeschlagen, in dem ML-basierte Faktendetektion interaktiv erfolgt. Der Prozess sieht Werkzeuge vor, welche es dem Domänenexperten erlauben, Fakten, die automatisiert vorgeschlagen werden, zu prüfen und gegebenenfalls in einen Wissensgraphen zu übernehmen. Beschrieben wird dieses Verfahren anhand von zwei Anwendungsfällen aus der Praxis: Versicherungen und technische Services.

In Kap. 5 wird überzeugend dargestellt, wie mit Verfahren der Sprachtechnologie in begrenzter Zeit und mit begrenzten Ressourcen ein WordNet aufgesetzt werden kann. OdeNet ist ein Open-Source-WordNet für die deutsche Sprache, also ein semantisches Netz für die Sprachverarbeitung. WordNets manuell zu entwickeln und zu pflegen ist möglich, aber höchst zeit- und kostenintensiv. In dem Kapitel wird stattdessen vorgestellt, wie ML-basierte Verfahren eingesetzt werden können, um WordNet-Einträge vorzuschlagen. Das funktioniert erstaunlich gut, bei begrenztem Aufwand. Es zeigt sich auch, dass ein neu aufgebautes WordNet (noch) nicht perfekt sein muss, um in Sprachanwendungen nützlich zu sein.

1.3.2 Wissensbasierte KI für ML

Eine ähnliche Thematik wie das vorangegangene Kapitel verfolgt Kap. 6. Hier geht es um die semantische Suche in Jobportalen und Weiterbildungsdatenbanken, basierend auf der Modellierung von Hintergrundwissen. Allerdings ist die manuelle Modellierung aufwendig und rechtfertigt mitunter nicht den damit verbundenen Nutzen. Um den Aufwand zu reduzieren, kann das ML-basierte Verfahren der Word-Embeddings eingesetzt

werden. Word-Embeddings stellen eine aus Textdokumenten berechnete, numerische Vektorrepräsentation von Wortbedeutungen dar. Allerdings ist es nicht so einfach, ohne weitere Aufbereitung der Word-Embeddings den Dokumentensuchraum sinnvoll einzuschränken. Wie das gelingen kann, und insbesondere auch wie modellierter Hintergrund in Word-Embeddings einfließen kann, wird im Kapitel beschrieben.

Digitalisierung und KI haben längst auch in der Landwirtschaft Einzug gehalten – Stichwort Digital Farming. Kap. 7 erläutert, wie hybride KI-Modelle für das Modellieren von pflanzlichem Wachstum und Erträgen verwendet werden können. Durch Kombination von wissensbasierten und ML-basierten Technologien und ausreichend vorhandenem Expertenwissen können vorab trainierte Vorhersagemodelle auf unvollständige oder teilweise inkonsistente Daten angewendet werden. Das resultiert in deutlich besseren Vorhersagen als die einzelnen KI-Ansätze separat.

Effektive und effiziente ML-Modelle zu entwickeln, ist komplex und erfordert große Expertise. Im Kap. 8 wird eine ML-Plattform vorgestellt, die das Ziel hat, ML auch für Fachexperten einer Domäne, z. B. Mediziner, verfügbar zu machen. Aber was hat das mit hybrider KI zu tun? Interessanterweise setzen die Autoren wissensbasierte KI ein, um die Effektivität und Effizienz von ML zu erhöhen. Insbesondere wird ein Wissensgraph für die Konfiguration von ML eingesetzt sowie ein Regelsystem, um Strategien für Ressourceneffizienz umzusetzen: Green AI mittels wissensbasierter KI.

1.3.3 Inhärent hybride Verfahren

Was geschieht, wenn in Krisensituationen wie einer Pandemie reguläre medizinische Behandlungen zurückgestellt sowie notwendige intensivmedizinische Behandlungen an die verfügbaren klinischen Kapazitäten angepasst werden müssen? Kap. 9 beschreibt ein Verfahren, welches ein inhärent hybrides KI-Verfahren für dieses Problem einsetzt: Bayes'sche Netze. Dabei werden Abhängigkeiten zwischen medizinischen Parametern in Form eines Graphen modelliert. Durch Einsatz von Falldaten können Wahrscheinlichkeiten berechnet werden, die für die Entscheidungsunterstützung im Krankenhaus verwendet werden können.

Kap. 10 befasst sich mit der automatisierten Annotation von Patientenakten in Krankenhäusern mit medizinischen Codenummern zur Beschreibung von Krankheitsbildern. Die Codenummern werden u. a. für die Abrechnung mit den Krankenkassen verwendet. Dies ist eine wichtige Aufgabe, deren manuelle Erfassung jedoch personal- und kostenintensiv ist. Die Idee, hier ML-basierte Textklassifikationsverfahren einzusetzen, liegt nahe. Es zeigt sich jedoch, dass aufgrund unterschiedlicher Vokabulare ML-Modelle für jedes Krankenhaus neu trainiert werden müssten. In diesem Kapitel wird gezeigt, wie durch den Einsatz von Ontologien Transfer Learning ermöglicht werden kann.

Kap. 11 befasst sich mit Anomaliedetektion in der verarbeitenden Industrie – ein relevantes Thema, da das frühe Erkennen von Produktionsfehlern in der intelligenten Fabrik

enorme Kosten einsparen kann. State-of-the-Art-Verfahren für Anomaliedetektion setzen ML auf Strömen von Sensordaten an, ignorieren dabei aber die Topologie der Anlage: Welches Förderband verbindet welche Roboter und wie findet die Verarbeitung statt? In dem Kapitel wird ein Ansatz vorgestellt, bei dem mittels Graph Neural Networks sowohl die Topologie der Anlage – modelliert als Graph – als auch Ströme von Sensordaten verarbeitet werden. Dabei werden Anomalien nicht nur erkannt, sondern auch lokalisiert.

Glück oder Pech? In Kap. 12 geht es um die Vorhersage von Sportergebnissen wie Fußballspielen. Wie in Kap. 9 werden dafür Bayes'sche Verfahren in Form von probabilistischer Programmierung eingesetzt. Diese erlauben es, Hintergrundwissen in die Modellierung einfließen zu lassen, um die Parameter des Modells bereits mit relativ wenigen Daten zu ermitteln. Trainiert werden die Verfahren mit vergangenen Spielergebnissen. Die resultierenden Vorhersagen, z. B. für eine Bundesligasaison, sind erstaunlich gut.

1.3.4 Kombination von ML und wissensbasierter KI

Wenn ein Notruf an die Telefonnummer 112 abgesetzt wird, ist schnelle Hilfe essenziell. Der Disponent muss mit möglichst wenigen gezielten Fragen ausreichend Informationen zum Notfall erheben, um die passenden Einsätze zu koordinieren. In Kap. 13 wird ein hybrides KI-System vorgestellt, welches Disponenten dabei unterstützt. Dafür wird regelbasiertes Wissen (Gesetze, Richtlinien etc.) mit erfahrungsbasiertem Wissen (Fallbasis) kombiniert.

Für autonomes Fahren sind Sicherheit, Nutzerakzeptanz und rechtliche Haftung der Hersteller entscheidend. Dafür braucht es regelkonforme Entscheidungsfindung, Erklärbarkeit sowie hohe Leistung. In Kap. 14 werden Ontologien und Regelsysteme vorgestellt, die Verkehrs- und Fahrvorschriften abbilden. Diese wissensbasierten KI-Ansätze werden kombiniert mit ML-Verfahren zur Erfassung von Verkehrssituationen aus Videodaten. Daneben werden die Formalisierung von Verkehrs- und Fahrregeln und deren Anwendung aus der Planungs- und Vorhersageperspektive vorgestellt.

Die Anwendung von KI im juristischen Umfeld ist gespickt mit spezifischen Herausforderungen. Erklärbarkeit und Transparenz sind eine unabdingbare Voraussetzung für viele Anwendungsfälle im juristischen Umfeld. Der reine Einsatz von ML oder der reine Einsatz von wissensbasierter KI ist nur bedingt möglich; für viele Fragestellungen wird ein hybrider Ansatz benötigt. Darauf gehen die Autoren von Kap. 15 ein.

Das Training neuronaler Netze ist kostenintensiv und ein möglicher Hinderungsgrund für den Einsatz von ML. Zwar gibt es eine Vielzahl von vorgefertigten Lösungen; diese passen jedoch oft nicht auf einen neuen Anwendungsfall. Kap. 16 zeigt auf, wie die Verbindung von ML-Lösungen mit einer Semantik eine schnelle und kostengünstige Adaption auf eine neue Domäne erlaubt.

KI wird häufig zur Unterstützung von Entscheidungen eingesetzt wie die Gewährung von Beihilfen, die Vergabe von Krediten oder die Vergabe von Fördermitteln. In Kap. 17

geht es um die Entscheidung zur Zulassung von Studierenden zu einem Studiengang. Der alleinige Einsatz von ML stößt hier an seine Grenzen wegen geringer Datenmengen, unvollständiger Daten und sich ändernder Situationen. Außerdem erfordert die Einhaltung von internen Richtlinien (Compliance) nachvollziehbare Schlussfolgerungen. Daher werden zusätzlich regelbasierte Verfahren eingesetzt – ein typisches Beispiel für hybride KI.

1.4 Überblick über die Buchkapitel

Abb. 1.3 gibt eine Übersicht über die Einordnung der einzelnen Buchkapitel (weiße, nummerierte Kreise entsprechend den Kapitelnummern) in folgende Kategorien: Technologie (blau), Branche (ocker), Anwendung (grün) und Daten (orange). Die Einordnung jedes Kapitels wird zur Übersicht innerhalb der Kapitel mit der gleichen Farbkodierung wiederholt.

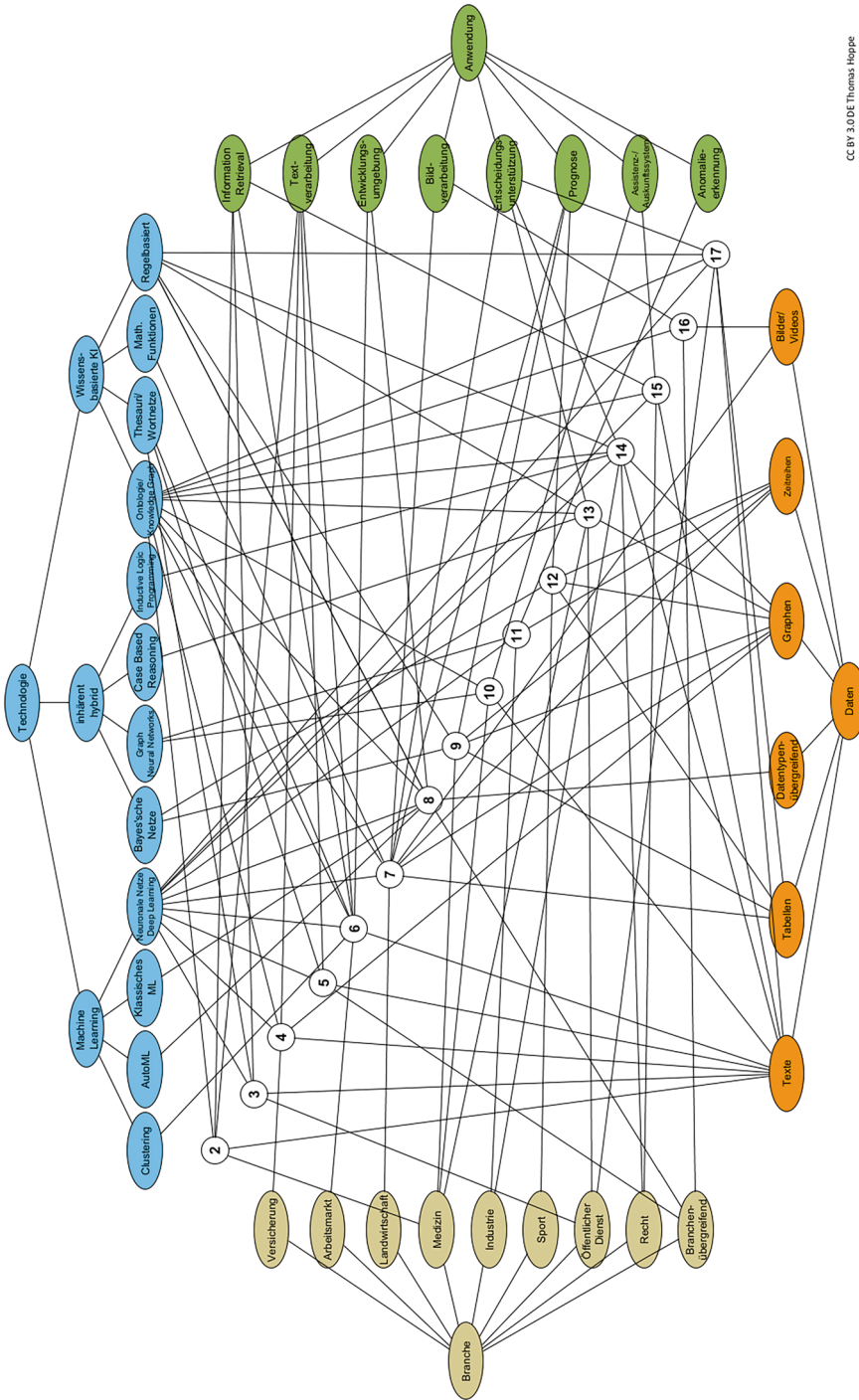
1.5 Boxology

Die Boxology (van Harmelen und ten Teije 2019) ist eine Notation zur graphischen Darstellung der vielfältigen Kombinationen von Techniken des Machine Learning mit symbolischen Techniken der Wissensverarbeitung. Die Boxology-Notation unterscheidet zwei Arten von Objekten:

- algorithmische Komponenten, dargestellt durch Ovale; *KR* (*Knowledge Representation*) basiert auf *deduktiven* Inferenzen und *ML* (*Machine Learning*) basiert auf *induktiven* Inferenzen;
- Ein-/Ausgabe algorithmischer Komponenten: *sym* steht für symbolische oder relationale Strukturen, die auf einem Datenmodell basieren („model-based“) und *data* steht für Daten ohne zugrunde liegendes Datenmodell, z. B. Vektoren, Bilder, Texte oder Zahlen („model-free“).

Die Verwendung dieser Boxology-Notation abstrahiert von konkreten Verfahren und Methoden und ermöglicht es somit, Anwendungen zu vergleichen, auch wenn sie unterschiedliche Methoden verwenden. In den einzelnen Kapiteln dieses Buchs verwenden wir die Boxology, um die vorgestellten Lösungen und Systeme zu systematisieren. So werden Gemeinsamkeiten und Unterschiede zwischen den Lösungen erkennbar.

Van Harmelen und ten Teije (2019) haben Muster für die in der Literatur vorkommenden Kombinationen dieser algorithmischen Komponenten und Kategorien ihrer Ein-/Ausgabe erstellt. Im Folgenden erläutern wir einige dieser Muster. Kom-



CC BY 3.0 DE Thomas Hoppe

Abb. 1.3 Kapitelübersicht

plexe Anwendungen – wie die meisten der in diesem Buch beschriebenen hybriden Anwendungen – können dabei auf der Kombination mehrerer Muster basieren.

1.5.1 Deduktive Inferenz

In der Boxology steht KR für die deduktive Inferenz, also die Herleitung von Erkenntnissen auf Basis von Beobachtungen oder Prämissen. Bei klassischen wissensbasierten Systemen besteht die Eingabe aus strukturierten Daten. Eine Inferenzkomponente leitet unter Nutzung von explizit repräsentiertem Wissen ein Ergebnis her (Abb. 1.4).

Die Grafiken in Abb. 1.5 zeigen zwei Beispielanwendungen für dieses Muster. Die Verwendung von Labels für die Knoten ermöglicht es, das Muster auf die Anwendung zu adaptieren, und vereinfacht so die Interpretation des Musters. Links ist ein Diagnosesystem für Krankheiten. Die Eingabe besteht aus einer strukturierten Eingabe von Symptomen und die Ausgabe ist die diagnostizierte Krankheit. Die zweite Anwendung zeigt die Konfiguration eines technischen Systems.

1.5.2 Lernen mit unstrukturierter Ein- und Ausgabe

Während (KR) für die deduktive Inferenz steht, repräsentiert (ML) eine induktive Inferenz. Sie entspricht der Trainingsphase beim maschinellen Lernen. Das Muster in Abb. 1.6 zeigt eine induktive Inferenz, bei der sowohl Ein- als auch Ausgabe modellfrei sind. Die Eingabe sind die Daten, aus denen gelernt wird, die Ausgabe ist das gelernte Modell, z. B. als neuronales Netz mit den erlernten Gewichten.

Beim Trainieren von Large Language Models besteht der Input aus einer großen Menge von Texten und die Ausgabe ist das Sprachmodell (Abb. 1.7).

Ein Klassifikator, z. B. zur Erkennung von Krankheiten oder Objekten, hat als Input eine große Menge von Bildern. Das Ergebnis des Lernens ist ein Klassifikator, z. B. in Form eines neuronalen Netzes (Abb. 1.8).

Zu beachten ist, dass das Ergebnis des Lernens nicht die Klassen sind, sondern ein Modell der Klassen. Dieses kann wiederum als Eingabe für eine deduktive Inferenz dienen, um ein konkretes Bild zu klassifizieren (Abb. 1.17 in Abschnitt „Kombination von induktiven und deduktiven Methoden“).

Abb. 1.4 Klassisches wissensbasiertes System





Abb. 1.5 Beispiele: **a** medizinische Diagnose, **b** Produktkonfiguration

Abb. 1.6 Klassisches Machine-Learning-System



Abb. 1.7 Beispiel: Trainieren eines Sprachmodells

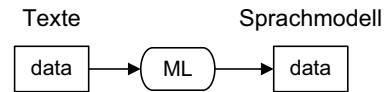
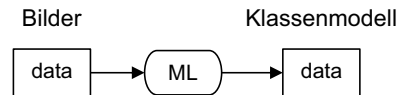


Abb. 1.8 Beispiel: Trainieren eines Modells zur Klassifikation von Krankheiten anhand von Bildern



1.5.3 Lernen mit symbolischer Ein- und Ausgabe

Anstatt auf modellfreie Daten kann ML auch auf symbolische Strukturen angewandt werden, was ebenfalls zu einer symbolischen Ausgabe führt (Abb. 1.9).

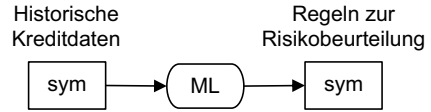
Beispiele hierfür sind induktive logische Programmierung (ILP; Muggleton und de Raedt 1994) und Markov Logic Networks (MLN; Richardson und Domingos 2006). Auch das Lernen von Entscheidungsbäumen aus strukturierten Daten (Quinlan 1986) lässt sich mit diesem Muster beschreiben. Dieses einfache Beispiel zeigt den Wert dieser abstrakten Muster: Auch wenn die Algorithmen und Darstellungen von ILP und MLN verschieden sind, zeigen die Architekturmuster, dass sie alle auf das gleiche Ziel ausgerichtet sind: induktives Schließen über symbolische Strukturen.

Anwendungen für dieses Muster findet man in vielen Bereichen, z. B. beim Lernen von Kriterien für die Vergabe von Krediten (Abb. 1.10).

Abb. 1.9 Symbolisches Machine Learning



Abb. 1.10 Beispiel: Lernen von Regeln über Kreditrisiken



1.5.4 Kombination von Daten und Symbolen

Es gibt Muster für den Wechsel zwischen symbolischen Strukturen und Daten. Ein Beispiel sind Anwendungen zur Graphenvervollständigung (Nickel et al. 2016; Wang et al. 2017). Abb. 1.11 zeigt einen ML-Algorithmus, einen Wissensgraphen in einen hochdimensionalen Vektorraum übersetzt. Induktive Schlussfolgerung wird verwendet, um zusätzliche Kanten vorherzusagen, die aufgrund von Mustern im Graphen als wahr angesehen werden, obwohl sie im ursprünglichen Graphen fehlen.

Eine Variante davon sind Anwendungen zum Lernen von symbolischen Mustern aus Daten (Abb. 1.12).

Dieses Muster erfasst das Lernen von Ontologien aus Texten (Abb. 1.13), für das es eine Reihe verschiedener Ansätze gibt: Ontologielernen durch induktive logische Programmierung (Konstantopoulos und Charalambidis 2010) oder Text Mining (Wong et al. 2012) sowie die Unterscheidung von Klassen und Instanzen (Padia et al. 2018).

1.5.5 Kombination von induktiven und deduktiven Methoden

Neben der Kombination von Daten und Symbolen gibt es auch Muster, die (zusätzlich) noch induktive und deduktive Inferenzen verbinden. Eine natürliche Erweiterung des Musters aus Abb. 1.13 ist die Verwendung der symbolischen Ausgabe als Eingabe für ein klassisches, schlussfolgerndes System, um eine verständliche Erklärung der ML-Ergebnisse zu erstellen (Abb. 1.14 und 1.15).

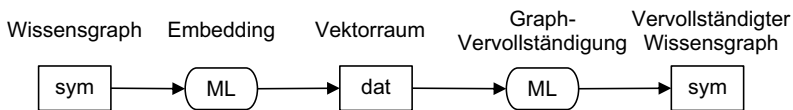
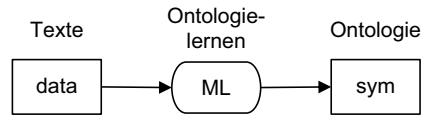


Abb. 1.11 Kombination von daten- und symbolbasiertem Machine Learning



Abb. 1.12 Lernen von symbolischen Mustern aus Daten

Abb. 1.13 Beispiel: Lernen von Ontologien



Für den Menschen verständliche Erklärungen basieren auf symbolischem Wissen. Das Muster aus Abb. 1.14 kann dahin gehend erweitert werden, dass es Anwendungen umfasst, die Hintergrundwissen verwenden, um die Erklärung herzuleiten. Tiddi et al. (2015) generieren Hintergrundwissen aus Linked Data, das selbst wiederum als symbolische Struktur repräsentiert ist, und leiten Erklärungen ab, die als Pfade von RDF-Properties repräsentiert sind.

Ein alternativer Ansatz für erklärbare Systeme wird in Chen et al. (2018) verfolgt, wo das Verhalten eines mit Transfer Learning trainierten neuronalen Netzes von einem Description Logic Reasoner untersucht wird, um das Verhalten des Lernalers zu erklären. Sie evaluieren ihr Verfahren mit Daten zur Vorhersage von Flugverspätungen (Abb. 1.16).

Das Beispiel in Abb. 1.17 zeigt die im Abschnitt „Lernen mit unstrukturierter Ein- und Ausgabe“ bereits erwähnte Anwendung eines aus modellfreien Daten gelernten Klassifikators zur Klassifikation eines Bildes.

Diese Muster zeigen verschiedene Möglichkeiten für die Kombination von induktiven und deduktiven Inferenzen mit Daten und Symbolen als Input und Output. Für eine vollständige Liste der Muster, die die in der Literatur bekannten Anwendungen beschreiben, siehe van Harmelen und ten Teije (2019).

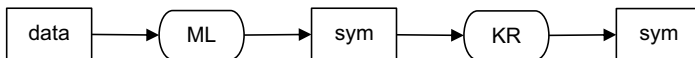


Abb. 1.14 Kombination von Machine Learning und wissensbasierter KI

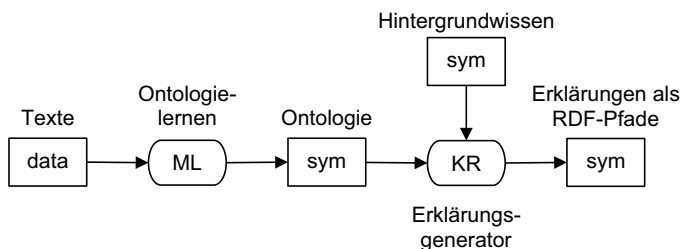


Abb. 1.15 Beispiel: Erklärungen von erlernten Ontologien

Abb. 1.16 Beispiel:
Erklärung von vorhergesagten
Flugverspätungen mittels
Inferenz

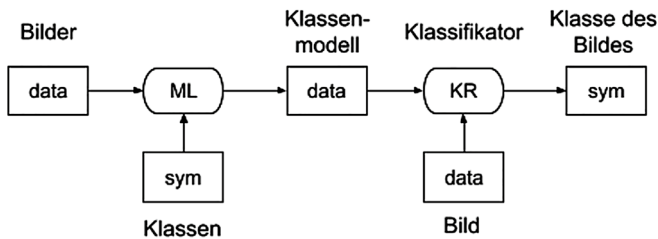
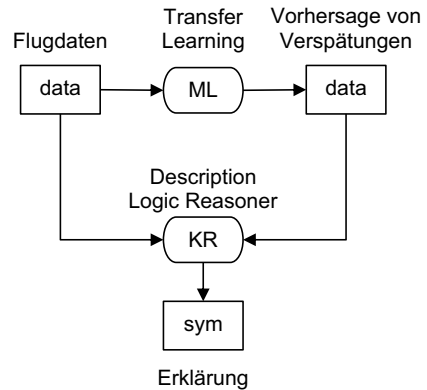


Abb. 1.17 Beispiel: Training und Anwendung eines Klassifikators

1.5.6 Human in the Loop

Witschel et al. (2021) haben die Boxology erweitert, um auch die Beteiligung von Menschen darstellen zu können. Das Muster in Abb. 1.18 beschreibt ein Szenario, bei dem maschinelles Lernen durch Feedback eines Menschen adaptiert wird. Das Muster kann als Spezialfall für inkrementelles Lernen angesehen werden. Der Mensch fungiert als „Lehrer“ oder „Trainer“ für das Lernsystem, indem er die Trainingsdaten anpasst oder beim Reinforcement Learning Rückmeldung gibt.

Durch Ersetzen der data-Box durch eine sym-Box beschreibt das Muster ein inkrementelles symbolisches Lernverfahren. Witschel et al. (2021) zeigen auch Beispiele für die Darstellung komplexer Anwendungen, bei denen Menschen mit induktiven und

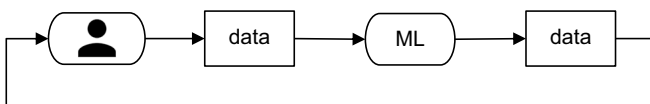


Abb. 1.18 Einbezug von Menschen in den Lernprozess

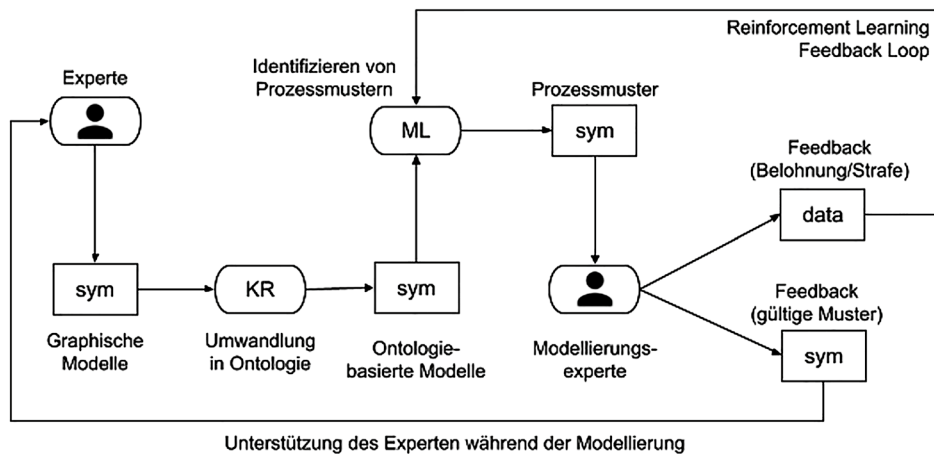


Abb. 1.19 Das Lernen von Prozessmustern (Witschel et al. 2021)

deduktiven Komponenten interagieren. Abb. 1.19 zeigt eine Boxology-Darstellung für einen Ansatz zum Lernen von Prozessmustern.

Wir halten die Boxology für eine geeignete Notation, um den hybriden Charakter einer KI-Anwendung herauszuarbeiten. Daher werden alle in diesem Buch beschriebenen KI-Anwendungen durch eine Boxology-Grafik charakterisiert.

1.6 Fazit

Seit den frühen Anfängen der KI in den 1950er-Jahren bestehen die beiden Familien von KI-Verfahren: wissensbasierte und ML-basierte Verfahren. Beide ergänzen sich ideal, haben aber in der Historie unterschiedlich große Aufmerksamkeit erfahren: in den Anfängen vorwiegend wissensbasierte Verfahren, seit den 2000er-Jahren vorwiegend ML-basierte Verfahren. Viele höchst anspruchsvolle Aufgaben lassen sich aber nicht mit einzelnen Verfahren, sondern nur mit hybrider KI, also der Kombination von wissensbasierten und ML-basierten Verfahren, zufriedenstellend lösen. Das ist technologisch anspruchsvoll. Wie es gelingen kann, zeigen 16 innovative Lösungen aus der Praxis für acht verschiedene Branchen. Sie werden in den folgenden Kapiteln beschrieben. In jedem Kapitel werden Kernaussagen und Handlungsempfehlungen gegeben, welche die Erfahrungen aus intensiver Projektarbeit kondensieren.

Überlegen Sie auch, hybride KI einzusetzen? Dann lassen Sie sich inspirieren!

Literatur

- (BBC, 2023) “AI ‘godfather’ Geoffrey Hinton warns of dangers as he quits Google”, BBC, Mai 2023, <https://www.bbc.com/news/world-us-canada-65452940>, letzter Zugriff 18.5.2023
- (Brooks, 2021), “An Inconvenient Truth About AI”, Rodney Brooks, IEEE Spektrum, September 2021, <https://spectrum.ieee.org/rodney-brooks-ai>, letzter Zugriff 18.5.2023
- (Chen et al. 2018) Knowledge-based transfer learning explanation. Jiaoyan Chen, Freddy Lécué, Jeff Z. Pan, Ian Horrocks, and Huajun Chen. In Michael Thielscher, Francesca Toni, and Frank Wolter, editors, Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference, KR 2018, Tempe, Arizona, 30 October - 2 November 2018., pages 349–358. AAAI Press, 2018.
- (Ege et al., 2015) „Corporate Semantic Web – Wie semantische Anwendungen in Unternehmen Nutzen stiften“, B. Ege, B. Humm, A. Reibold (Hrsg.), Springer Verlag, 2015.
- (Hilbert & López, 2011) “The World’s Technological Capacity to Store, Communicate, and Compute Information” Hilbert, M., López, P., *Science*, 332(6025), 60–65. <https://doi.org/10.1126/science.1200970>
- (Hoppe et al., 2018) „Semantic Applications – Methodology, Technology, Corporate Use“, T.Hoppe, B. Humm, A. Reibold, (Hrsg.), Springer Verlag, 2018.
- (Humm et al., 2022) Grundlagen und Anwendungen von KI. Bernhard G. Humm, Peter Buxmann, Jan C. Schmidt: In: Carl Friedrich Gethmann, Peter Buxmann, Julia Distelrath, Bernhard G. Humm, Stephan Lingner, Verena Nitsch, Jan C. Schmidt, Indra Spiecker genannt Döhmann: Künstliche Intelligenz in der Forschung – Neue Möglichkeiten und Herausforderungen für die Wissenschaft. Springer Verlag, Berlin, Germany, 2021. ISBN 978-3-662-63448-6, <https://doi.org/10.1007/978-3-662-63449-3>
- (Ihlenfeld, 2010) “Google kauft Metaweb”, Jens Ihlenfeld, 19.7.2010, in: golem.de, <https://www.golem.de/1007/76542.html>, letzter Zugriff 18.5.2023
- (Klein, 2014) “Faktendatenbank für Wikipedia: Google gibt Freebase an Wikidata”, Thorsten Klein, 17.12.2014, in: heise online, <https://www.heise.de/newsticker/meldung/Faktendatenbank-fuer-Wikipedia-Google-gibt-Freebase-an-Wikidata-2498806.html>, letzter Zugriff 18.5.2023
- (Konstantopoulos & Charalambidis, 2010). Formulating description logic learning as an inductive logic programming task. Stasinou Konstantopoulos, Angelos Charalambidis, IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 18–23 July, 2010.
- (Lenat & Marcus, 2023) “Getting from Generative AI to Trustworthy AI: What LLMs might learn from Cyc”, Doug Lenat, Gary Marcus, [arXiv:2308.04445](https://arxiv.org/abs/2308.04445), Juli, 2023, letzter Zugriff 2.9.2023
- (Lu, 2018) “Yann LeCun’s IJCAI Keynote: We Need a World Model“, Victor Lu, Juli 2018, syncedreview in medium.com <https://medium.com/syncedreview/yann-lecuns-ijcai-keynote-we-need-a-world-model-b5248f50e50f>, letzter Zugriff 18.5.2023
- (Muggleton & de Raedt, 1994) Inductive logic programming: Theory and methods. Stephen Muggleton, Luc de Raedt, *The Journal of Logic Programming*, 19, 629–679.
- (Nickel et al. 2016) A review of relational machine learning for knowledge graphs. Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich Proceedings of the IEEE, 104(1):11–33, 2016. <https://doi.org/10.1109/JPROC.2015.2483592>
- (Padia et al. 2018) Automating class/instance representational choices in knowledge bases. Ankur Padia, David Martin, and Peter F. Patel-Schneider. In Catherine Faron-Zucker, Chiara Ghidini, Amedeo Napoli, and Yannick Toussaint, editors, Knowledge Engineering and Knowledge Management – 21st International Conference, EKAW 2018, Nancy, France, November 12–16,

- 2018, Proceedings, volume 11313 of Lecture Notes in Computer Science, pages 273–288. Springer, 2018.
- (Quinlan, 1986). Induction of decision trees. J.R. Quinlan, *Machine Learning*, 1(1), 81–106.
- (Richardson & Domingos, 2006). Markov logic networks. Matthew Richardson, Pedro Domingos, *Machine learning*, 62, 107–136.
- (Schaul et al., 2023) “Inside the secret list of websites that make AI like ChatGPT sound smart”, Kevin Schaul, Szu Yu Chen, Nitasha Tiku, April 2023, in: The Washington Post, <https://www.washingtonpost.com/technology/interactive/2023/ai-chatbot-learning/>, letzter Zugriff 18.5.2023
- (Schreiner, 2022) “OpenAI: DALL-E 2 hat Probleme mit Konzepten – und kann nicht zählen”, Maximillian Schreiner, May 2022, in The Decoder <https://the-decoder.de/dall-e-2-wo-openais-ki-bildmaschine-noch-scheitert/>, letzter Zugriff 18.5.2023
- (Tiddi et al. 2015) Data patterns explained with linked data. Ilaria Tiddi, Mathieu d’Aquin, and Enrico Motta. In Albert Bifet, Michael May, Bianca Zadrozny, Ricard Gavaldà, Dino Pedreschi, Francesco Bonchi, Jaime S. Cardoso, and Myra Spiliopoulou, editors, Machine Learning and Knowledge Discovery in Databases – European Conference, ECML PKDD volume 9286 of Lecture Notes in Computer Science, pages 271–275. Springer.
- (van Harmelen & ten Teije, 2019) “A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems”, Frank van Harmelen, Annette ten Teije, *Journal of Web Engineering*, Vol. 18 1–3, 97–124. <https://doi.org/10.13052/jwe1540-9589.18133>
- (Wang et al. 2017) Knowledge graph embedding: A survey of approaches and applications. Q. Wang, Z. Mao, B. Wang, and L. Guo. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743.
- (Witschel et al., 2021) “Visualization of patterns for hybrid learning and reasoning with human involvement”, Hans Friedrich Witschel, Charuta Pande, Andreas Martin, Emanuele Laurenzi, Knut Hinkelmann, In: Dornberger, R. (eds) *New Trends in Business Information Systems and Technology. Studies in Systems, Decision and Control*, vol 294. Springer, Cham. https://doi.org/10.1007/978-3-030-48332-6_13
- (Wolfram, 2023) “WolframAlpha as the Way to Bring Computational Knowledge Superpowers to ChatGPT”, Steven Wolfram, January 2023, in : Steven Wolfram Writings, <https://writings.stephenwolfram.com/2023/01/wolframalpha-as-the-way-to-bring-computational-knowledge-superpowers-to-chatgpt/>, letzter Zugriff 18.5.2023
- (Wong et al. 2012) Ontology learning from text: A look back and into the future. Wilson Wong, Wei Liu, and Mohammed Bennamoun. *ACM Computing Surveys* 44(4):20:1–20:36, 2012.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.




Teil I

Machine Learning für wissensbasierte KI



Unterstützung klinischer Studien mit hybrider KI

2

Silvio Peikert, Naouel Karam, Ralph Schäfermeier, Le Duyen Sandra Vu, Johannes Jung, Şefika Efeoğlu, Jamal Al Qundus und Adrian Paschke 

Kernaussagen

1. Hybride KI-Ansätze vereinen die Zuverlässigkeit symbolischer KI und die Flexibilität von ML-Methoden.
2. Formale, wissensbasierte Ansätze erfüllen die in kritischen Anwendungsszenarien auftretenden Anforderungen bezüglich Präzision, Korrektheit, Zuverlässigkeit und Erklärbarkeit.
3. Maschinelle Lernverfahren unterstützen das Erfassen großer Datenmengen und das Auffinden relevanten Wissens sowie die Generierung von neuen Erkenntnissen.
4. Das Lösungsdesign für ein konkretes System erfordert eine Abwägung zwischen Modellierungs- und Trainingsaufwand.

S. Peikert · N. Karam · R. Schäfermeier · L. D. S. Vu · J. Jung
Fraunhofer-Institut Fokus, Berlin, Deutschland
E-Mail: silvio.peikert@fokus.fraunhofer.de

N. Karam
E-Mail: naouel.karam@fokus.fraunhofer.de

L. D. S. Vu
E-Mail: Le.Duyen.Sandra.Vu@fokus.fraunhofer.de

J. Jung
E-Mail: johannes.jung@fokus.fraunhofer.de

Ş. Efeoğlu · A. Paschke (✉)
Freie Universität Berlin, Berlin, Deutschland
E-Mail: adrian.paschke@fokus.fraunhofer.de

J. Al Qundus
Middle East University, Amman, Jordanien

5. Für Medizin und Pharmakologie existieren viele Ontologien, Terminologien und Datenbanken, die zur Entwicklung genutzt werden können.
6. Natürlichsprachliche Schnittstellen ermöglichen Fachexperten die Nutzung von Knowledge-Graphen auch ohne vertiefte technische Kenntnisse.

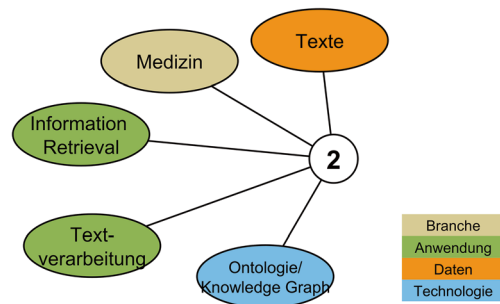
2.1 Einführung

Klinische Studien sind für Forscher das wichtigste Mittel, um die Wirksamkeit und Sicherheit chirurgischer oder pharmazeutischer Behandlungsmethoden zu evaluieren. Die als Prüfer verantwortlichen Ärzte tragen erhebliche Verantwortung und müssen die Sicherheit der Studienteilnehmer sowie die wissenschaftlich korrekte Erhebung und Analyse der Daten gewährleisten. Solche Studien sind in der Regel datenintensiv und müssen ggf. Daten aus einer Vielzahl von Quellen und Formaten verarbeiten und verknüpfen. Dadurch ergibt sich für dieses Anwendungsfeld ein hoher Bedarf mit anspruchsvollen Anforderungen an Verfahren zur Datenintegration, -analyse und -abfrage.

Nachfolgend wird ein Lösungsansatz vorgestellt, der Forscherinnen und Forscher bei der Durchführung klinischer Studien unterstützt, indem mittels hybrider KI das Auffinden und Analysieren relevanter Informationen erleichtert wird. Dabei werden heterogene Daten aus verschiedenen textuellen Quellen und Systemen in einen harmonisierten Knowledge Graph (KG) überführt, der als integrierte Sicht auf diese Informationen fungiert und so die Beantwortung einer Vielzahl von Fragestellungen ermöglicht. Wesentliche Aspekte der Datentransformation, Informationsextraktion, Wissensrepräsentation, Wissensabfrage und des Ontologiemanagements werden dabei betrachtet und es wird aufgezeigt, wie diese durch KI unterstützt werden können. Eine thematische Einordnung dieses Kapitels kann Abb. 2.1 entnommen werden.

Der Lösungsansatz wird anhand beispielhafter Einsatzszenarien vorgestellt. Klinische Studien haben Richtlinien zur Auswahl geeigneter Teilnehmer, die als Einschlusskriterien bezeichnet werden. Diese Einschlusskriterien werden auf Basis von Merkmalen wie Alter, Geschlecht, Art und Stadium einer Erkrankung, Vorbehandlungen und weiterer

Abb. 2.1 Thematische Einordnung des Kap. 2



<p>Inclusion criteria:</p> <ul style="list-style-type: none">• Known CAD¹ (angiography evidence of > 50% coronary artery stenosis or prior coronary revascularization) or PAD² brachial index (ABI³ or prior revascularization)• Currently on treatment with either aspirin (81 mg/qd), aspirin (81 mg/qd) plus clopidogrel (75 mg/qd) <p>Exclusion criteria:</p> <ul style="list-style-type: none">• Active pathological bleeding, history of clinically significant bleeding events, or deemed at increased risk of bleeding• Creatinine Clearance (CrCL) <20 mL/min• Any clinical indication to be on triple antithrombotic therapy (DAPT⁴ plus an oral anticoagulant)• Acute coronary event within the past 90 days• Prior hemorrhagic stroke or intracranial hemorrhage• Ischemic stroke or transient ischemic attack within the past 6 months• Chronic use of nonsteroidal anti-inflammatory drugs

Abb. 2.2 Beispiel für In- und Exklusionskriterien aus einem Studienprotokoll^{1,2,3,4,5}

medizinischer Umstände definiert und beinhalten sowohl Eignungskriterien – das sind erforderliche Merkmale zur Teilnahme – als auch Ausschlusskriterien – das sind Merkmale, die geeignete Teilnehmer nicht aufweisen. Abb. 2.2 zeigt ein Beispiel solcher Einschlusskriterien aus einem Studienprotokoll.

Zur Prüfung dieser Kriterien bezüglich einer teilnehmenden Person sind umfangreiches Fachwissen und Abgleiche mit verschiedenen Datenquellen nötig. Um beispielsweise das Ausschlusskriterium „nonsteroidal anti-inflammatory drugs“ zu prüfen, müssen zunächst alle dieser Gruppe angehörigen Medikamente sowie deren Inhaltsstoffe gelistet und anschließend mittels komplexer Abfragen mit Patientendatenbanken abgeglichen werden.

¹ Coronary Artery Disease (Koronare Herzkrankheit)

² Peripheral Arterial Disease (Periphere arterielle Verschlusskrankheit)

³ Ankle Brachial Index (Knöchel-Arm-Index)

⁴ Dual Antiplatelet Therapy (Duale antithrombozytäre Therapie)

⁵ <https://clinicaltrials.gov/ct2/show/NCT03718429>, aufgerufen am 05.06.2023.

2.2 Architektur

Abb. 2.3 zeigt die dreischichtige Architektur des Systems, bestehend aus einer zentralen Wissensrepräsentationsschicht, einer Datenintegrationsschicht und einer Wissenszugriffsschicht, als Boxology-Modell.⁶

Im Zentrum der Wissensrepräsentationsschicht steht ein KG, der alle zur Bearbeitung der Aufgabenstellungen nötigen Daten in einem einheitlichen Schema speichert. Darin werden gleichermaßen das nötige Fachwissen und die zu verarbeitenden Daten gespeichert. Das Fach- oder Domänenwissen wird aus geeigneten Wissensartefakten importiert. Das sind z. B. unabhängig entwickelte Taxonomien oder Ontologien, die domänenspezifisches Wissen – wie beispielsweise die Zusammenhänge zwischen Arzneimitteln, Inhaltsstoffen und Produktbezeichnungen – in maschinell interpretierbarer Form speichern. Für die Anwendung sind die Verknüpfung und Aufbereitung mehrerer solcher Wissensartefakte erforderlich. Dieser Vorgang des Knowledge Engineering wird in Abschn. 2.3 näher beschrieben. Zudem werden Daten aus bestehenden Dokumenten und IT-Systemen benötigt, die ebenfalls in den zentralen KG einfließen sollen. Daten aus IT-Systemen liegen bereits in strukturierter oder zumindest semistrukturierter Form vor und können mittels bekannter ETL-Prozesse (Extract, Transform, Load) in den KG überführt werden.

Die Datenintegrationsschicht bündelt Verfahren zur Integration von Informationen aus schwach strukturierten oder unstrukturierten Quellen, wie oben erwähnten Studienprotokollen oder Notizen. Lösungsmöglichkeiten für entsprechende Informationsextraktionsverfahren reichen hier von KI-unterstützten regelbasierten Methoden bis zu KI-Sprachverarbeitungsverfahren auf Basis von Large Language Models. Dies wird in Abschn. 2.4 näher beschrieben.

Zuletzt kommen in der Wissenszugriffsschicht regelbasierte Methoden und KI-Verfahren zum Einsatz. Während sich einfache Anwendungsfälle mittels statischer, vorab definierter Datenabfragen bearbeiten lassen, sind für komplexe Recherchen flexiblere Zugänge zum KG erforderlich. Abschn. 2.6 erläutert, wie KI-Verfahren Abfragen mittels natürlicher Sprache ermöglichen, diese auf die Struktur des KG abbilden und die gespeicherten Fakten bereitstellen.

2.3 Wissensrepräsentation

Intelligente Systeme haben oft mit komplexen Fragestellungen zu tun, deren Komplexität sich aus der zugrunde liegenden Problemdomäne ergibt. Dies trifft auch auf das Beispiel der Einnahme eines bestimmten Medikaments als Ausschlusskriterium für die Teilnahme an einer Studie zu. Die Komplexität ergibt sich aus der Tatsache, dass das

⁶Für eine Einführung in die Boxology siehe Kap. 1.

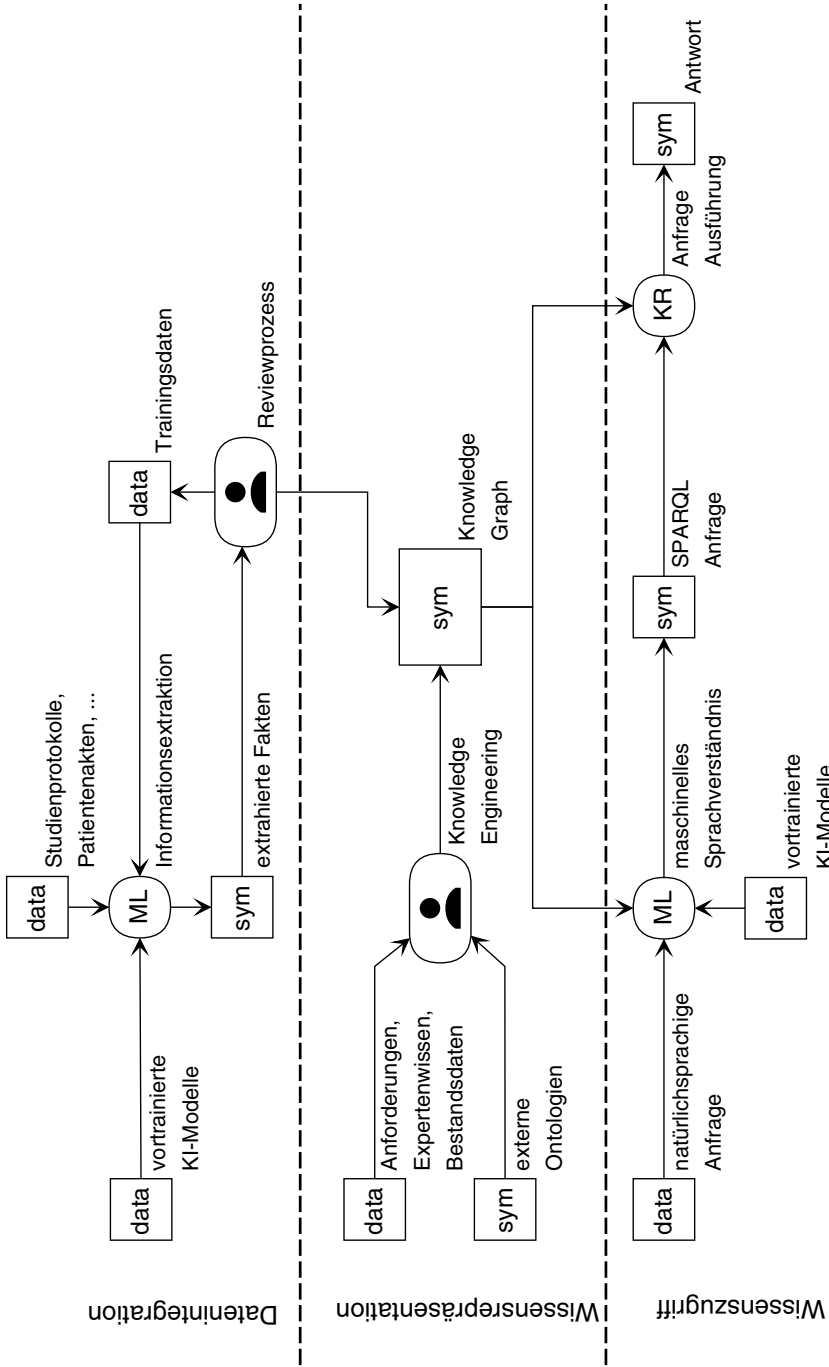


Abb. 2.3 Boxology-Modell des hybriden KI-Systems

im Studienprotokoll genannte Medikament lediglich ein generischer Bezeichner für eine ganze Gruppe von Medikamenten ist, die sich durch gemeinsame Eigenschaften (z. B. entzündungshemmend, nichtsteroidal) definieren lassen. Zudem ist meist gar nicht das Präparat selbst relevant, sondern ein bestimmter enthaltener Wirkstoff, sodass alle Generika, die diesen Wirkstoff enthalten, als Ausschlusskriterium infrage kommen.

Die Wissensrepräsentation stellt hier formale Methoden zur Verfügung, mit denen sich derartige komplexe Problemstellungen präzise formulieren lassen, sodass Abfragen auf entsprechenden Wissensbeständen durchgeführt werden können und dabei (beweisbar) korrekte Ergebnisse liefern, was in sicherheitskritischen Domänen, wie der Medizin, bei der Fehler im schlimmsten Fall Menschenleben kosten können, natürlich äußerst wichtig ist. Dies geschieht mithilfe der Semantik, die jeder Wissensrepräsentationsformalismus definiert und auf deren Grundlage eindeutige Abfragen und Schlussfolgerungen möglich werden.

Aber nicht nur bei der internen Wissensverarbeitung und -verwendung spielt die Semantik eine Rolle. In Bereichen wie der Medizin und Pharmakologie existieren umfangreiche, teils frei nutzbare Standardvokabulare in Form von Ontologien und Wissensbasen. Wie im Software Engineering hat sich auch im Bereich der wissensbasierten Systeme das Prinzip der Wiederverwendung etabliert, um nicht ressourcenaufwendig das sprichwörtliche Rad immer wieder neu erfinden zu müssen. Durch die Verwendung global eindeutiger Bezeichner in Standardsprachen zur Wissensrepräsentation, wie OWL und RDF, lassen sich Objekte, Konzepte und Relationen aus externen Quellen referenzieren und gegebenenfalls auf eigene Objekte, Konzepte oder Relationen abbilden (Mapping).

RDF (Resource Description Framework) und OWL (Web Ontology Language) sind Standards zur Beschreibung von Ressourcen im Web und deren Beziehungen zueinander. Ressourcen und Relationen werden mit einheitlichen Ressourcenbezeichnern (Unified Resource Identifiers, URIs) eindeutig identifiziert. RDF-Beschreibungen bestehen aus Subjekt-Prädikat-Objekt-Tripeln, die sich kombiniert zu (Wissens-)Graphen zusammenfügen. OWL fügt zusätzlich eine formal-logische Semantik (basierend auf sog. Beschreibungslogiken) hinzu, die komplexere Beschreibungen von Objekten und Klassen sowie komplexere Relationen (z. B. symmetrische, reflexive oder transitive Relationen) zulässt. Die standardisierte Semantik der Ontologiesprache erlaubt es standardkonformen Inferenzalgorithmen (sogenannte Reasoner), per logischer Schlussfolgerung implizites Wissen zu entdecken und explizit zu machen.

Es gibt, ähnlich wie in der Softwareentwicklung, verschiedene Entwicklungsmodelle für Ontologien (Schäfermeier und Paschke 2015). Ein typischer Entwicklungsprozess, der auf der Neon-Methodologie⁷ basiert, ist in Abb. 2.4 dargestellt. Es beinhaltet die Entwicklung eines eigenen, auf die Anwendung zugeschnittenen Schemas (in Abb. 2.4 mit T1.- beginnend), welches wiederum ähnlich wie in der Software- oder Datenbankentwicklung Anforderungsmanagement sowie Prozeduren zur Überführung

⁷http://neon-project.org/nw/NeOn_Book.html, aufgerufen am 24.03.2023.

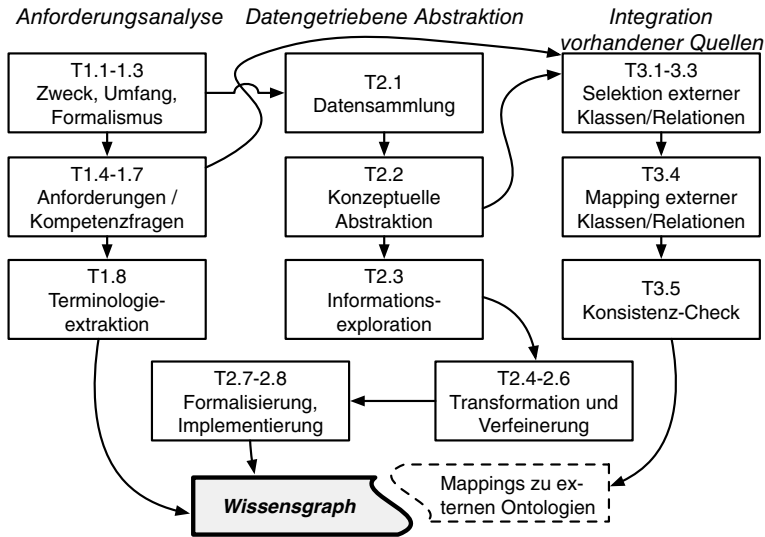


Abb. 2.4 Ontologieentwicklungsprozess, der die eigene Ontologieentwicklung, Wiederverwendung und Mappings verbindet. Quelle: Eigenkreation, basierend auf (Al Qundus et al. 2021)

von Anforderungen in eine formale Repräsentation (T2) beinhaltet. Im Anschluss daran folgen Auswahl und Integration externer Vokabularien, Terminologien und Ontologien (T3). Zum Abschluss findet das konzeptuelle Mapping auf das in Schritt T2 entwickelte Zielschema statt.

Folgende externe Ontologien sind für den Anwendungsfall der Auswahl von Studienteilnehmenden besonders relevant:

*Anatomical Therapeutic Chemical Classification (ATC)*⁸ stellt eine Taxonomie aktiver Inhaltsstoffe von Medikamenten bereit. Die Klassifikation geschieht anhand der Organe oder Systeme, auf die der entsprechende Inhaltsstoff wirkt, sowie der therapeutischen, pharmakologischen und chemischen Eigenschaften des Stoffes.

RxNorm Vocabulary (RX) stellt normalisierte Namen für Medikamente in klinischen Studien zur Verfügung und verlinkt diese mit zahlreichen Standardvokabularien, die bspw. in Anfragesystemen zu Wechselwirkungen und im Arzneimittelmanagement verwendet werden.

Concept Unique Identifiers (CUI) stellt eine Liste eindeutiger Bezeichner aus dem UMLS-Metathesaurus (einem umfangreichen Thesaurus, der synonyme Konzepte aus über 200 biomedizinischen Vokabularien integriert) bereit sowie eine Relation, mit der Konzepte aus eigenen oder anderen externen Begriffshierarchien oder Ontologien auf diese gemappt werden können.

⁸<https://bioportal.bioontology.org/ontologies/ATC>, aufgerufen am 24.03.2023.

UMLS Semantic Types (STY) wurden ursprünglich für den UMLS-Metathesaurus entwickelt. Die STY stellen eine Taxonomie genereller Begriffe (als „semantische Typen“ bezeichnet) und Relationen dar, die eine gemeinsame High-Level-Kategorisierung medizinischer Begriffe und Relationen aus unterschiedlichen Vokabularien ermöglichen.

Die *Clinical Trials Ontology (CTO)* stellt Begriffe und Relationen zur Beschreibung und Registrierung klinischer Studien bereit.

In Abb. 2.5 ist beispielhaft zu sehen, wie diese externen Quellen aufeinander abgebildet wurden. Auf der linken Seite im Bild ist ein Unterast der ATC-Klassifikation zu sehen, der den Inhaltsstoff *Indometacin* und dessen Oberkonzepte in Form von Wirkstoffklassen beinhaltet. Dasselbe Konzept ist (mit seiner eigenen ID) ebenfalls im RxNorms-Vokabular enthalten, hier mit dem Ziel, den Inhaltsstoff mit den Medikamenten, in denen er enthalten ist, in Verbindung zu bringen. Die beiden gleichen, aber unterschiedlich bezeichneten Konzepte werden mittels der Relation *umls:cui* (Concept Unique Identifier) auf den lexikalischen Identifier *C0021246* aus dem UMLS-Metathesaurus abgebildet. Das RxNorms-Vokabular wird wiederum mit den UMLS Semantic Types verbunden, indem manuell Sub-/Superklassenrelationen eingefügt werden. In diesem Beispiel sind das in *Term Indometacin* als Subklasse von *Pharmaceutical Substance* sowie die Einnahmeform *Indometacin 40 mg Oral Capsule* mit dem Markennamen *Tivorbex* als Subklasse von *Clinical Drug*.

Der so entstehende KG kann auf unterschiedliche Weise gespeichert und verwaltet werden. Im einfachsten Fall geschieht dies in der Form einer bzw. mehrerer OWL- oder RDF-Dateien. Bei großen Datenmengen wird in der Regel ein RDF-Triplestore verwendet, der den Graph in Tripelform speichert und über genormte Anfrageschnittstellen (z. B. SPARQL) effizient zugreifbar macht. Zur Verbindung der im KG enthaltenen Objekte und Konzepte mit deren Vorkommen in Textquellen bieten sich wiederum

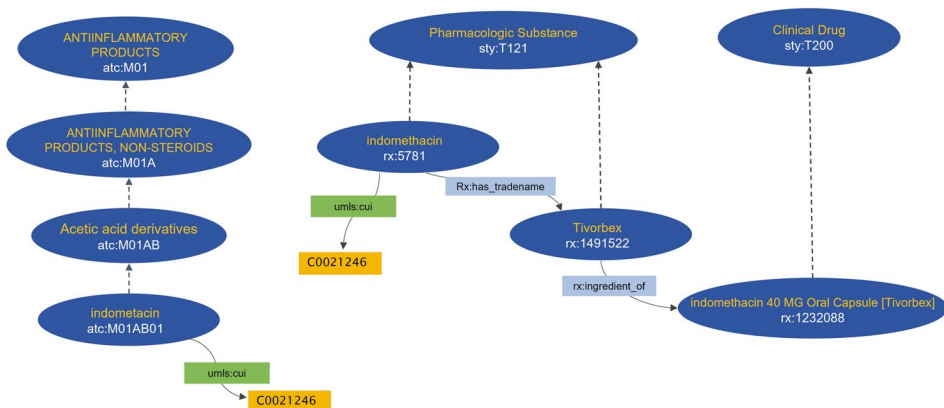


Abb. 2.5 Mapping von Konzeptrelationen und lexikalischen Bezeichnern

Terminologie-Services an, die speziell auf die Suche nach einzelnen Konzepten und deren Unter- und Ober- bzw. ähnlichen Konzepten optimiert sind.

2.4 Informationsextraktion

Ein Großteil der im Rahmen klinischer Studien zu verarbeitenden Daten liegt in unstrukturierter Form vor. Information-Extraction-(IE-)Pipelines strukturieren Informationen aus diesen Dokumenten und ermöglichen die Integration in den KG. Solche IE-Pipelines werden meist spezifisch für bestimmte Dokumententypen und Fragestellungen entwickelt. Die Analyseschritte einer solchen Pipeline umfassen Named Entity Recognition (NER), Coreference Resolution, Relation Extraction und Event Detection. Coreference Resolution dient der Zuordnung von Entitäten zu Pronomen, wie z. B. „er“, „sie“ oder „es“, die auf diese Entitäten verweisen. Eine Entität ist ein Objekt oder eine Sache im Text, welche benannt oder kategorisiert werden kann und in Relation zu anderen Entitäten stehen kann. Die Event Detection beinhaltet die automatische Identifikation von Ereignissen im Text anhand der W-Fragen: „was“, „wo“, „wann“ und „wer“. Die wichtigen Schritte der Named Entity Recognition und Relation Extraction werden in den folgenden Absätzen anhand einiger Beispiele genauer erklärt und dienen dazu, unstrukturierte Textdaten in ein strukturiertes Format zu überführen. In den Beispielen wird ein NER-Modell anhand medizinischer Textdaten trainiert und anschließend ein Distant-Supervision-Ansatz für die Relation Extraction benutzt. Die benutzten Anwendungen und Vorgehensweisen zur IE in den Beispielen sind der Arbeit von Efeoglu und Paschke (2023) entnommen.

Abb. 2.6 und 2.7 zeigen Beispiele für Entitäten und NER. Diese Daten werden genutzt, um das NER-Modell zu trainieren. Die in Abb. 2.6 gezeigten Auszüge aus den Trainingsdaten⁹ müssen danach in das CoNLL-Format (Tjong Kim Sang und De Meulder 2003) konvertiert werden. Bei NER-Prozessen werden Erwähnungen von Entitäten im Text gefunden und einer Kategorie zugeordnet. Entitäten können unter anderem Menschen, Orte und Organisationen oder domänenspezifisch auch Krankheiten und Medikamente umfassen. Beispielsweise werden im ersten Stichpunkt in Abb. 2.2 die Entitäten *CAD*, *PAD* und *Coronary Artery Stenosis* als Kategorie *Disease* annotiert.

Um ein NER-Verfahren zu realisieren, gibt es zwei unterschiedliche Ansätze: zum einen regelbasierte Verfahren und zum anderen trainierbare Modelle. Letztere haben durch ihre universelle Einsetzbarkeit sehr an Beliebtheit gewonnen. Um Modelle zu trainieren, werden allerdings Trainingsdaten benötigt, also in unserem Fall klinische Texte in der speziellen Domäne, welche mit NER annotiert wurden. Je nach Anwendungsfall

⁹Beispieldaten sind annotiert im CoNLL-Format verfügbar auf https://github.com/sefeoglu/clinical_trials_hybrid_ai/tree/master/annotated_training_data_for_ner/train.txt, aufgerufen am 15.03.2023.

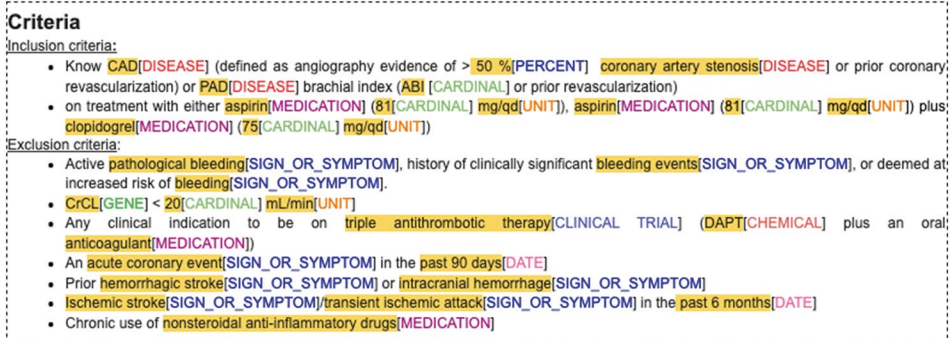


Abb. 2.6 Klinische Textdaten mit Entitätstypen, die mittels der Annotationspipeline generiert wurden

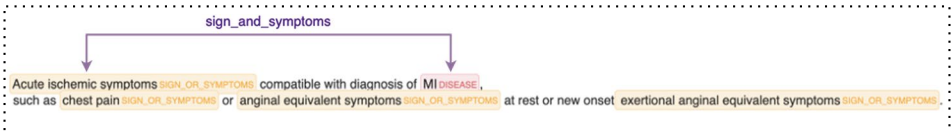


Abb. 2.7 Ein Beispiel für die binäre Relation „sign_and_symptoms“ zwischen einem Symptom oder Anzeichen und einer Krankheit

und Domäne existieren auch bereits vortrainierte Modelle oder regelbasierte Verfahren. Die klinischen Texte können allgemeine (Abb. 2.6: *81 (CARDINAL)*) sowie domänen-spezifische (Abb. 2.6: *CrCL (GENE)*) Entitäten enthalten, welche in den Trainingsdaten annotiert werden müssen. Es gibt Ansätze zur Vereinfachung des Annotationsaufwands solcher Texte. Diese machen sich existierende NER-Anwendungen für allgemeinere Bereiche (z. B. GATE, Apache OpenNLP und SpaCy) zunutze, um eine vereinfachte Annotationspipeline zu generieren. In unserem Beispiel wurde eine Annotationspipeline benutzt, welche mithilfe bekannter NER-Anwendungen und Instanzen aus ATC-, RX- und STY-Ontologien generiert wurde. Nach Generierung der Annotationen können diese mithilfe von existierenden NLP-Frameworks (in unserem Beispiel FLAIR (Akbik *et al.* 2019) und SpaCy¹⁰) zum Training und zur Auswertung eines NER-Modells verwendet werden.

Relation Extraction, eine weitere wichtige Komponente der semantischen Analyse von IE-Pipelines, identifiziert die Relationstypen zwischen den Entitäten in einem Dokument. Die Erkennung der Relationen zwischen Entitätspaaren kann mithilfe unterschiedlicher Verfahren erfolgen: überwachtes Lernen, Distant Supervision, Open Information Extraction oder mit regelbasierten Verfahren. In Abb. 2.7 ist ein Beispiel der Relation

¹⁰ <https://spacy.io/>.

Extraction zu sehen, welches durch ein Distant-Supervision-Verfahren realisiert wurde. Distant Supervision ist eine ML-Methode zur Relationsextraktion, die Techniken des selbstüberwachten oder unüberwachten maschinellen Lernens in Verbindung mit bestehenden KGs, wie beispielsweise Wikidata,¹¹ verwendet, um Relationen zu extrahieren (Smirnova et al. 2018). In Abb. 2.6 wird eine Relation vom Typ „sign_and_symptoms“ zwischen den Entitäten „Acute ischemic symptoms (SIGN_OR_SYMPTOMS)“ und „MI (DISEASE)“ gefunden. Es handelt sich dabei um ein Inklusionskriterium für Patienten, die unter Herzattacken¹² leiden. Im Beispiel wurde ein Extraktionsverfahren für verschiedene vordefinierte Relationstypen entwickelt. Dieses erkennt die Relationen „has cause“, „signs and symptoms“, „synonyms“ und „part of“.

Eine weitere Möglichkeit, Informationen zu extrahieren, besteht darin, textbasierte Dialogsysteme beziehungsweise Chatbots zu verwenden. Diese sind meist transformer-basierte KI-Systeme, welche dem Nutzer ermöglichen, Anfragen in natürlicher Sprache zu formulieren, und diese dann wiederum in natürlicher Sprache beantworten. Für eine Verwendung zur Informationsextraktion aus Texten muss das System die Möglichkeit bieten, einen Text entgegenzunehmen, und Fragen zu diesem Text beantworten können. Der Vorteil solcher Systeme besteht darin, dass sie meist sehr einfach zu bedienen sind und der Nutzer kein großes Vorwissen braucht, um diese effizient zu nutzen.

Im Folgenden wird zwischen lokalen Systemen und Cloud-basierten Systemen unterschieden. Cloud-basierte Systeme wie ChatGPT sind über das Internet zugänglich, wobei die Server meist von den Herausgebern des Modells zur Verfügung gestellt werden. Bei sensiblen Daten könnte das ein Sicherheits-, Datenschutz- oder Geschäftsgeheimnischutzproblem darstellen. Demgegenüber stehen die lokalen Systeme, welche auf einem eigenen Gerät oder lokalem Netzwerk laufen. Diese lokalen Systeme sind meist um ein Vielfaches kleiner als Cloud-basierte Systeme, von denen die größten Modelle mehr als 150 Mrd. trainierbare Parameter besitzen können. Dementsprechend sind lokale Systeme in ihrer Funktionsweise oftmals sehr eingeschränkt. Nachfolgend wird Informationsextraktion aus Texten mithilfe von Frage-Antwort-Systemen am Beispiel der Inklusions- und Exklusionskriterien aus Abb. 2.2 demonstriert.

Im vorliegenden Beispiel geht es um die Extraktion von In- und Exklusionskriterien aus Studienprotokollen. Dabei sollen alle Kriterien extrahiert werden, welche sich auf die Medikation beziehen, und dabei deren Parameter, wie das betroffene Medikament, dessen Dosierung und die Behandlungsdauer, strukturiert erfasst werden. Hierzu wird eine Kette von Fragen an ein Conversational-AI-System, wie ChatGPT, gestellt. Anfangs wird dem System der Eingabetext zur Verfügung gestellt. Danach werden vorgefertigte Anweisungen gestellt. Diese sind in Abb. 2.8 aufgelistet.

Am Ende gibt das System eine Tabelle mit den gewünschten Informationen aus, die anschließend in den KG eingespeichert werden können. Bei der Formulierung der Fragen

¹¹ <https://www.wikidata.org>.

¹² Das Beispiel ist zu finden unter <https://beta.clinicaltrials.gov>, aufgerufen am 15.03.2023.

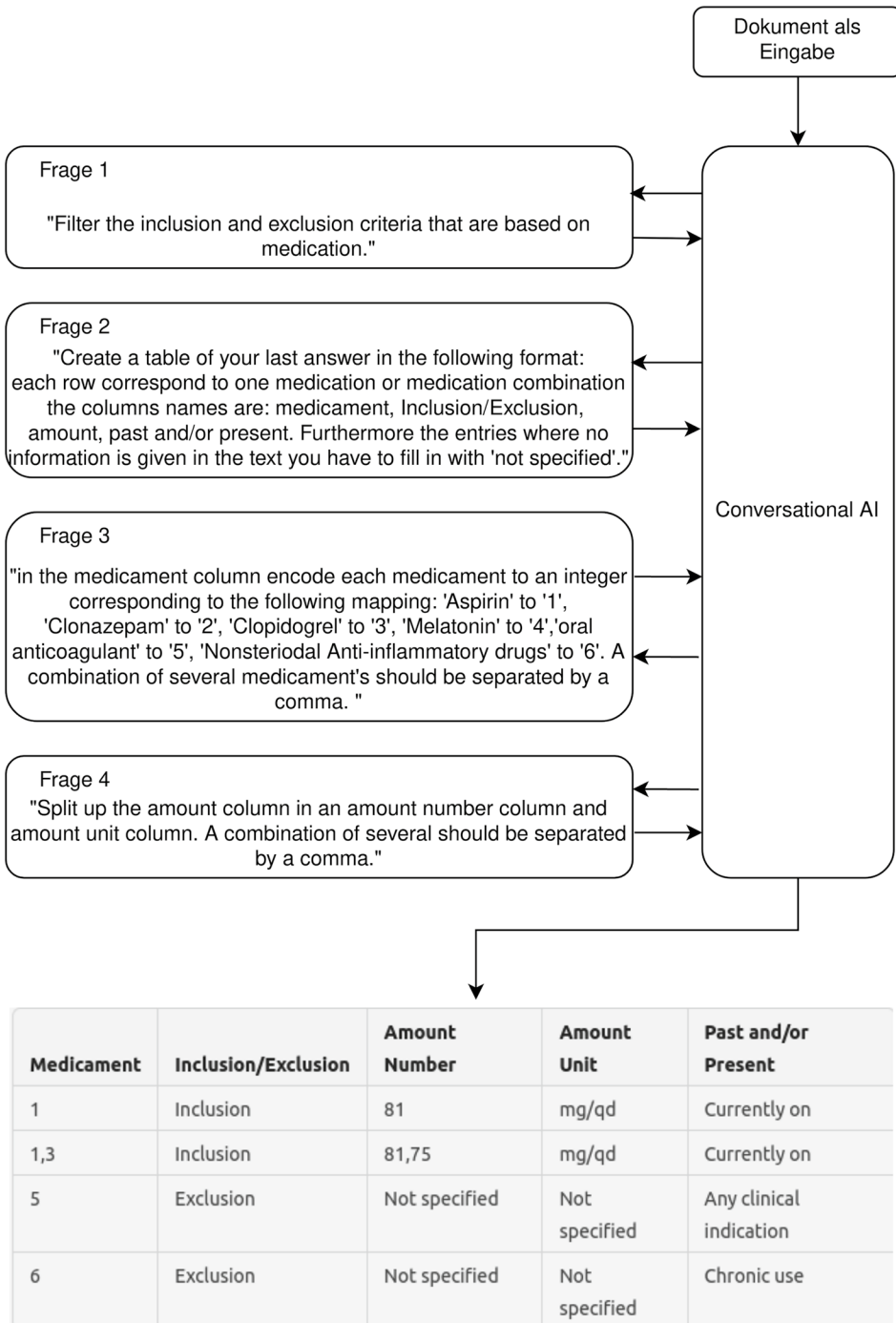


Abb. 2.8 Anweisungen zur Extraktion von In- und Exklusionskriterien mittels ChatGPT

sowie bei dem Format der Antworten muss auf die Eigenheiten des verwendeten Systems Rücksicht genommen werden. So gibt es Fragen, welche in manchen Systemen sehr gute Antworten liefern, aber bei anderen Systemen nicht sinnvoll verarbeitet werden können.

Das Format der Antworten spielt bei der Extraktion eine wichtige Rolle. Die Anweisungen sollten so formuliert werden, dass die Antworten möglichst einheitlich sind und somit besser weiterverarbeitet werden können. In Frage 3 in Abb. 2.8 werden die verschiedenen Medikamente in Zahlen kodiert. Das dient dazu, dass für jedes Medikament eine eindeutige Antwort geliefert wird. Wenn durch das System keine solche Kodierung möglich ist, könnte für ein und dasselbe Medikament beispielsweise entweder die Antwort *Nonsteroidal Anti-inflammatory Drugs* oder die Abkürzung *NAIDs* zurückgegeben werden. Ist dies der Fall, so muss bei der Formatierung der Antworten auf reguläre Ausdrücke (RegEx) zurückgegriffen werden. RegEx sind spezielle Zeichenfolgen, die verwendet werden, um nach bestimmten Mustern in Texten zu suchen.

Beachtet sollte auch werden, dass alle diese Systeme natürlich nicht fehlerfrei sind. Oftmals gibt das System falsche oder unpassende Antworten zurück. Beispielsweise haben viele Systeme Probleme damit, sinnvolle Antworten zu liefern, wenn die gestellte Frage sich nicht aus dem Text beantworten lässt. Ein Beispiel solch einer Frage zum Dokument in Abb. 2.2 wäre: „Wie hoch ist der Blutdruck?“ Einige Systeme würden hier eine Antwort generieren, obwohl dies offensichtlich unmöglich ist. Um Abhilfe zu schaffen, geben viele Systeme deshalb neben der Antwort auch Konfidenzwerte zurück. So lassen sich Antworten mit geringen Konfidenzwerten aussortieren. Jedoch sollte auch hier beachtet werden, dass diese Konfidenzwerte je nach System und Anfrage nicht vollständig verlässlich sind. Die einheitliche Formatierung der Antworten dient auch dazu, dass falsche Informationen aussortiert werden. Doch selbst nach der Formatierung sollte immer davon ausgegangen werden, dass ein gewisser Restteil an falschen Informationen übrig bleibt.

Um die Fehleranfälligkeit der Anwendung zu bestimmen, empfiehlt es sich, eine Evaluierung durchzuführen. Falls das verwendete System Konfidenzwerte zurückgibt, kann die Evaluierung auch dazu verwendet werden, einen Schwellenwert zu bestimmen, ab dem eine Antwort als gültig angesehen und somit weiterverarbeitet wird. Bei der Evaluierung müssen Testdaten vorbereitet werden, bei denen die Informationen, welche extrahiert werden können, schon bekannt sind. Die bekannten Informationen werden dann mit den Informationen abgeglichen, die von der Anwendung zurückgegeben werden. Die Evaluierung kann für Fragenketten sowie für einzelne Fragen durchgeführt werden. Fehleranfällige Fragen sollten abgeändert oder entfernt werden.

2.5 Informationsanfragen

Klinische Forscher benötigen schnelle und zuverlässige Antworten aus den verstreuten Studiendaten. So versuchen Forscher beispielsweise, die Sicherheit und Wirksamkeit einer Intervention zu bestimmen, indem sie bestimmte Ergebnisse bei den Teilnehmern messen und nach unerwünschten Ereignissen suchen, d. h. nach unbeabsichtigten und

unerwünschten Nebenwirkungen, die teilweise mit der Verwendung eines Arzneimittels in Verbindung gebracht werden. Diese unerwünschten Ereignisse werden in der Patientenakte des Studienteilnehmers dokumentiert. Dieser Prozess wird erleichtert, indem die medizinischen Daten der Teilnehmer aus verschiedenen Quellen integriert und geeignete Abfragen für eine kontinuierliche Sicherheitsbewertung eingerichtet werden. Hierzu sind komplexe Abfragen des zugrunde liegenden KG erforderlich, die mittels natürlicher Sprache und ohne Kenntnis von Datenbankabfragesprachen direkt vom Nutzer formuliert werden können.

Die meisten Datenanalysten im Umfeld klinischer Datenverarbeitung sind nicht in der Tiefe mit semantischen Technologien, KGs und Abfragesprachen wie SPARQL oder den genauen Vokabularen innerhalb des KG vertraut. Entsprechend wichtig sind nutzerfreundliche Zugangsschnittstellen für IT-Systeme in diesem fachlichen Umfeld. Unter dem Begriff Natural Language Interfaces werden Lösungsansätze zusammengefasst, die das Formulieren komplexer Anfragen an ein solches System mittels natürlicher Sprache ermöglichen und Fachexperten so in die Lage versetzen, selbstständig und effizient in den Datenbeständen zu recherchieren. Die Übersetzung natürlichsprachlicher Anfragen in technische Abfragesprachen ist aufgrund der vielfältigen Ausdrucksmöglichkeiten ein technisch schwieriges Problem (Baksi 2021). Die Lösungsansätze umfassen Controlled Natural Language (CNL), semantische Analyse und statistische Verfahren.

CNL ist ein Ansatz, der versucht, die Komplexität der Sprache zu reduzieren. CNLs haben sowohl eine formale Syntax als auch Semantik und können auf eine bestehende formale Sprache abgebildet werden. Die Benutzer sind auf eine vordefinierte Grammatik und ein Vokabular beschränkt, welches auf dem Inhalt und der Struktur des KG basiert.

Semantische Analyse kann mittels Templates oder regelbasiert erfolgen. Regelbasierte Ansätze verwenden eine vordefinierte Menge an Regeln. Diese können auf grammatischer Ebene oder auf Bedeutungsebene angewendet werden. So könnten beispielsweise Grammatikregeln verwendet werden, um ein Prädikat aus der Anfrage zu erkennen. Template-basierte Ansätze hingegen verwenden eine Vorlage, die Platzhalter für relevante Informationen in der Anfrage enthält. Die Platzhalter werden dann durch die entsprechenden SPARQL-Ausdrücke ersetzt, um die vollständige SPARQL-Abfrage zu erstellen. Ähnlich wie bei kontrollierter Sprache werden hier Anfragen durch festgelegte Schablonen eingeschränkt (Unger *et al.* 2012).

Statistische Ansätze versuchen die Abbildung von natürlichsprachigen Anfragen zu SPARQL-Abfragen auf Grundlage einer Datenbasis zu erlernen. Eine solche Datenbasis liegt praktisch jedoch meist nicht im erforderlichen Umfang vor.

Abb. 2.9 zeigt die Benutzeroberfläche unseres CNL-Anfragesystems „AskTONI“ (Karam *et al.* 2020). Die Konstruktion von Anfragen erfolgt interaktiv mittels eines endlichen Automaten. Dieser definiert zusammen mit dem KG die Abfragesprache. Während der Eingabe einer Frage werden, basierend auf dem Automaten und dessen Zuständen, Phrasen vorgeschlagen und Eingaben vervollständigt. Die Bestandteile der Eingabe werden farbig unterlegt: hellgrau für vordefinierte Phrasen, grün für Konzepte, dunkelgrau für Eigenschaften und gelb für Instanzen innerhalb des KG. Die Grenzen dieser

Bestandteile entsprechen Transitionen des zugrunde liegenden Automaten, der die Anfrage während der Eingabe in eine SPARQL-Abfrage übersetzt. Dabei generiert jede Transition des Automaten ein Fragment, das der SPARQL-Abfrage hinzugefügt wird. Sobald der Automat durch Abschluss der Anfrage durch Eingabe von „.“ oder „?“ seinen Endzustand erreicht, wird die generierte Abfrage ausgeführt und das Ergebnis angezeigt.

Abb. 2.10 zeigt die aus der konstruierten Anfrage generierte SPARQL-Abfrage. Die PREFIX-Deklaration ermöglicht es, Bezeichnungen in Abfragen zu verwenden, anstatt überall vollständige URIs zu nutzen. Die SELECT-Klausel gibt an, welche Variablen (in Form von Fragezeichen gefolgt von einem Variablennamen) im Ergebnis enthalten sein sollen. In diesem Beispiel wird die Variable „?c12“ zurückgegeben. Die WHERE-Klausel definiert die Bedingungen für die Abfrage. In der Klausel werden Muster angegeben, die die Beziehungen zwischen Entitäten und Eigenschaften beschreiben. In der SPARQL-Abfrage wird beispielsweise die Eigenschaft „treatedWith“ verwendet, um die Beziehung zwischen den Entitäten „EnrolledSubject“ und „PLACEBO“ zu beschreiben.

Zur semantischen Analyse sind regelbasierte Ansätze besonders geeignet. Sie benötigen keine spezielle Trainingsdatenbasis und ermöglichen mehr Variabilität als ein Template-basierter Ansatz. Dabei müssen Entitäten und Eigenschaften extrahiert werden, die dann in der WHERE-Klausel die Bedingungen für die SPARQL-Abfrage definieren. Die regelbasierte Anfrageanalyse kann in 3 Schritten erfolgen: Zunächst wird die



Abb. 2.9 Benutzeroberfläche des Anfragesystems AskTONI mit kontrollierter Sprache

```

PREFIX study: <https://w3id.org/phuse/study#>
PREFIX code: <https://w3id.org/phuse/code#>
SELECT distinct ?c12
WHERE {
?c12 rdf:type study:EnrolledSubject.
?c12 study:treatedWith study:PLACEBO.
?c12 study:afflictedBy ?c72.
?c72 rdf:type study:AdverseEvent.
?c72 study:serious code:NoYesResponse_Y.
}

```

Abb. 2.10 SPARQL-Abfrage zur Anfrage: „Give me enrolled subjects treated with placebo and afflicted by a serious adverse event?“

Anfrage tokenisiert, d. h. wortweise zerlegt, anschließend werden mittels sogenannter Part-of-Speech-Tagger, auch POS-Tagger, die Wortarten der einzelnen Token bestimmt und basierend darauf werden Konstituenten mit Shallow Parsing (Federici et al. 2002) bestimmt. Der aktuelle Entwicklungsstand nutzt POS-Tagger mit statistischen Modellen, bei denen neuronale Netze zum Einsatz kommen. Konstituentengrammatiken können als kontextfreie Grammatik modelliert werden. Eine kontextfreie Grammatik besteht aus Symbolen und Produktionsregeln. Abb. 2.11 zeigt beispielhaft, wie eine Phrase in Nominalphrasen zerlegt wird.

Für die Anfrage: „Give me adaptive design studies with participants aged between 30 and 60“, wird die SPARQL-Anfrage in Abb. 2.12 generiert. Abb. 2.13 zeigt die regelbasiert generierten Teilbäume und Tab. 2.1 zeigt, wie diese Fragmente schrittweise in Bestandteile der SPARQL-Abfrage übersetzt werden. Zusammengehörige Teile sind dabei in der Abfrage und Tabelle farblich hervorgehoben.

Um die SPARQL-Abfrage zu generieren, müssen neben der Grammatik auch die Begriffe formalisiert werden. RDF-Graphen verwenden Uniform Resource Identifier (URI) als eindeutige Referenz und mindestens ein Label als sprachliche Bezeichner für Entitäten. Um Worte aus der Anfrage in URIs zu übersetzen, können wörterbuchbasierte, lexikalische oder vektorielle Textähnlichkeiten zu den Labels des KG bestimmt werden.

2.6 Diskussion

Der KG-basierte Lösungsansatz zur Informationsverwaltung im Umfeld klinischer Studien birgt drei wesentliche Herausforderungen: Benötigt werden Mechanismen, die Informationen präzise und flexibel genug repräsentieren, Verfahren, um Informationen zuverlässig aus strukturierten und unstrukturierten Datenquellen in den KG zu überführen, und gleichermaßen Abfragemöglichkeiten, um das gespeicherte Wissen mit geringem technischen Aufwand für fachliche Fragestellungen nutzen zu können.

Formale Wissensmodelle in Form von KG, Ontologien oder Terminologien bieten den Vorteil, dass auf ihnen logisches Schlussfolgern möglich ist. Anfragen werden mit hoher (in der Regel absoluter) Genauigkeit beantwortet und Antworten sind, im Gegensatz zu

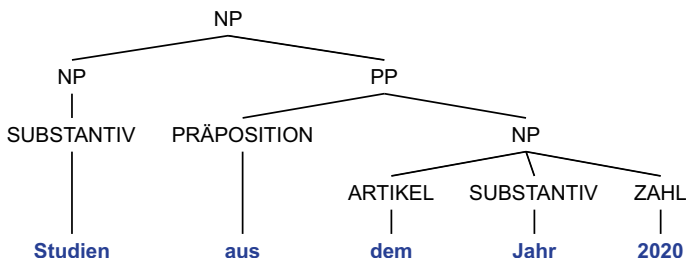


Abb. 2.11 Konstituentenbaum einer Phrase

```

PREFIX study: <https://w3id.org/phuse/study#>
PREFIX code: <https://w3id.org/phuse/code#>
SELECT distinct ?a
WHERE {
  ?a rdf:type study:Study.
  ?a study:adaptiveDesign code:NoYesResponse_Y.
  ?a study:hasStudyParticipant ?b.
  ?b code:hasAge ?c.
  FILTER (xsd:integer(?c) >= 30 && xsd:integer(?c) <= 60).

```

Abb. 2.12 SPARQL-Anfrage zu: „Give me adaptive design studies with participants aged between 30 and 60“

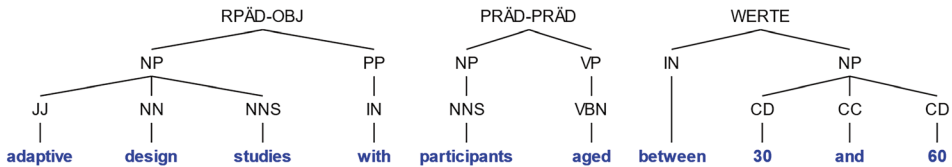


Abb. 2.13 Regelbasiert generierte Teilbäume zur Anfrage: „Give me adaptive design studies with participants aged between 30 and 60“

Tab. 2.1 Fragmentweise Übersetzung einer Anfrage in SPARQL

Wort	POS	Produktionsregel	Label	URI
adaptive design	JJ NN	PRÄD-OBJ	Adaptive Design	study:adaptiveDesign
studies	NNS	PRÄD-OBJ	Study	study:Study
participants	NNS	PRÄD-PRÄD	Study Participants	study:hasStudyParticipant
aged	VBN	PRÄD-PRÄD	Age	code:hasAge
30	CD	WERTE		
60	CD	WERTE		

denen subsymbolischer Ansätze, erklärbar. Dies ist insbesondere in Hochrisikodomänen wie der Medizin von großer Bedeutung. Durch standardisierte Identifizierungsmechanismen (URIs/IRIs) und eindeutig definierte Semantik ist die Wiederverwendung existierender Ontologien möglich. Im Gegensatz zu den Ansätzen des maschinellen Lernens werden zur Erstellung von Ontologien keine großen Mengen an Trainingsdaten benötigt.

Demgegenüber steht der personelle Aufwand, den die Sammlung relevanter Begriffe, Konsultation mit Domänenexperten und letztendlich die Modellierung der domänenrelevanten Konzepte und Relationen mit sich bringen. Die Erstellung eines Domänenmodells erfordert Experten sowohl aus der Fachdomäne als auch solche, die sich mit der verwendeten Formalisierungssprache und -methode auskennen. Das resultierende

Modell enthält am Ende exakt das, was die Domänenexperten modelliert haben. Sub-symbolische Ansätze, insbesondere solche, die auf neuronalen Netzen basieren, sind hier mitunter flexibler und können Features automatisch extrahieren. Darüber hinaus setzt die Integration passender externer Wissensmodelle natürlich deren Existenz voraus. Die Medizin ist in der glücklichen Lage, in diesem Bereich gut abgedeckt zu sein – in anderen Domänen sieht das unter Umständen anders aus. Das Wissensmodell muss strukturell zu den vorhandenen Daten passen. Ist dies nicht der Fall, muss entweder das Wissensmodell angepasst werden oder die Daten müssen einen Preprocessing-Schritt durchlaufen, der sie an das Wissensmodell anpasst. Subsymbolische Ansätze sind auch hier flexibler.

Insgesamt haben beide Ansätze ihre Vor- und Nachteile, die jeweils von der Domäne und der Datenlage bzw. dem Vorhandensein formaler Wissensmodelle abhängen. Das in diesem Kapitel beschriebene Szenario ist ein Beispiel dafür, dass der beste Weg oft in der Kombination beider Ansätze liegt. Kap. 6 beschreibt hier einen neuartigen Ansatz zur Lösung des auch in dieser Arbeit relevanten Problems der Ermittlung von Synonymen (mittels Word-Embeddings) und zur Ergänzung auf diesem Wege extrahierter Begriffe mit Wissen aus existierenden Vokabularen.

Die Anwendung einer IE-Pipeline, deren Hauptbestandteile RE und NER bilden, ist eine Möglichkeit, einen KG zu konstruieren. Was diesen Ansatz bedeutend macht, ist die Möglichkeit, große Mengen an unstrukturierten Daten zu verarbeiten und in ein strukturiertes Format zu übersetzen. Ein Nachteil von NER- und RE-Ansätzen, welche auf maschinellem Lernen basieren, ist, dass für das Training solcher Modelle annotierte Trainingsdaten benötigt werden. Um den Annotationsaufwand zu verringern, gibt es Verfahren, welche eine Annotationspipeline aus existierenden NER-Modellen und beispielsweise Entitäten aus bestehenden Ontologien generieren.

Eine weitere Möglichkeit der IE besteht darin, textbasierte Dialogsysteme beziehungsweise Chatbots zu verwenden. Der Vorteil hierbei besteht in der Allgemeinheit der Anwendungsgebiete sowie der einfachen Bedienung, welche kein großes Vorwissen benötigt. Je nach System und Anwendung kann die Qualität der Ergebnisse jedoch variieren. Es empfiehlt sich daher, das Verfahren in dem speziellen Anwendungsfall zu evaluieren. Ein weiteres Problem besteht darin, dass die leistungsstarken Systeme oftmals Cloud-basiert sind und eventuellen Datenschutz- und Sicherheitsstandards nicht genügen. Abhilfe kann durch lokale Modelle geschaffen werden, die im Allgemeinen allerdings deutlich schlechtere Ergebnisse liefern.

CNL erhöht die Benutzerfreundlichkeit, indem mehrdeutige und vage Ausdrücke in natürlicher Sprache vermieden werden. Dies führt zu besseren Antworten in Bezug auf Vollständigkeit und Genauigkeit, weshalb dieser Ansatz in einigen Fällen gegenüber Lösungen mit höherer Anfrageflexibilität zu bevorzugen ist. Ein Nachteil dieses Ansatzes besteht jedoch darin, dass Benutzer sich an eine bestimmte Grammatik und ein bestimmtes Vokabular halten müssen, um relevante Ergebnisse zu erhalten.

Ein Nachteil von semantischen Analyseansätzen liegt darin, dass die Genauigkeit der Übersetzung von natürlicher Sprache in SPARQL stark von der Qualität der verwendeten Grammatikregeln abhängt. Wenn die Regeln unvollständig oder unpräzise sind, können

falsche oder unvollständige Anfragen generiert werden, die zu ungenauen oder irrelevanten Ergebnissen führen können.

Insgesamt hängt die Wahl zwischen Shallow-Parsing- und Controlled-Vocabulary-Ansätzen von den spezifischen Anforderungen und Zielen der Anwendung ab. Eine Kombination beider Ansätze kann jedoch zu einer höheren Genauigkeit und Benutzerfreundlichkeit führen.

Für jede der drei Kernherausforderungen wurden verschiedene technische Lösungsansätze skizziert. Dabei profitieren insbesondere die IE aus unstrukturierten Textdaten und Anfragen an KG von neuen Entwicklungen im Bereich der KI. Maschinelle Lernverfahren ermöglichen wissensbasierten Systemen eine höhere Flexibilität, indem sie die Anpassbarkeit an neue Dokumententypen und Fragestellungen verbessern.

2.7 Empfehlungen

Abschließend können folgende Empfehlungen an IT-Architekten von wissensbasierten Systemen ausgesprochen werden:

1. Bei der Erstellung formaler Wissensmodelle sollte auf existierende Ontologien zurückgegriffen werden.
2. Um Zuverlässigkeit und Flexibilität zu vereinen, empfiehlt sich die Kombination aus symbolischer Wissensrepräsentation und Machine-Learning-Ansätzen.
3. Um einen KG aus unstrukturierten Daten zu generieren, kann eine IE-Pipeline benutzt werden.
4. Der Annotationsaufwand kann durch Nutzung bestehender Ontologien verringert werden.
5. Bei der Auswahl eines textbasierten Dialogsystems sollte eine Abwägung von Datensicherheit (lokale Systeme) und Performance (Cloud-basierte Systeme) stattfinden.
6. Die Kombination von Shallow-Parsing- und Controlled-Vocabulary-Ansätzen kann zur Entwicklung natürlichsprachiger Anfrageschnittstellen mit hoher Genauigkeit und Benutzerfreundlichkeit genutzt werden.

Literatur

Akbik, A. *et al.* (2019) 'FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP', in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 54–59. Available at: <https://doi.org/10.18653/v1/N19-4010>.

Al Qundus, J. *et al.* (2021) 'ROC: An Ontology for Country Responses towards COVID-19', in A. Paschke *et al.* (eds) *Proceedings of the Conference on Digital Curation Technologies (Quarator 2021), Berlin, Germany, February 8th – to – 12th, 2021*. CEUR-WS.org (CEUR Workshop Proceedings). Available at: https://ceur-ws.org/Vol-2836/quarator2021/_paper/_13.pdf.

- Baksi, K.D. (2021) ‘Recent Advances in Automated Question Answering In Biomedical Domain’, *ArXiv* [Preprint]. Available at: <https://www.semanticscholar.org/paper/Recent-Advances-in-Automated-Question-Answering-In-Baksi/7c5064305b8f8add57f211e185cdf9d68e6e31> (Accessed: 2 May 2023).
- Efeoglu, S. and Paschke, A. (2023) ‘Fine-Grained Named Entities for Corona News’, in. *14th International Conference for Semantic Web Applications and Tools for Health Care and Life Sciences (SWAT4HCLS)*, Basel, Switzerland.
- Federici, S., Montemagni, S. and Pirrelli, V. (2002) ‘SHALLOW PARSING AND TEXT CHUN-KING: A VIEW ON UNDERSPECIFICATION IN SYNTAX’, in. Available at: <https://www.semanticscholar.org/paper/SHALLOW-PARSING-AND-TEXT-CHUNKING%3A-A-VIEW-ON-IN-Federici-Montemagni/bc136ee666887713fa7c4642e48ee8cb3bf80443> (Accessed: 2 May 2023).
- Karam, N. et al. (2020) ‘Answering Controlled Natural Language Questions over RDF Clinical Data’, in A. Harth et al. (eds) *The Semantic Web: ESWC 2020 Satellite Events*. Cham: Springer International Publishing (Lecture Notes in Computer Science), pp. 129–134. Available at: https://doi.org/10.1007/978-3-030-62327-2_22.
- Schäfermeier, R. and Paschke, A. (2015) ‘Verteilte und agile Ontologieentwicklung’, in B. Ege, B. Humm, and A. Reibold (eds) *Corporate Semantic Web: Wie semantische Anwendungen in Unternehmen Nutzen stiften*. Berlin, Heidelberg: Springer (X.media.press), pp. 341–358. Available at: https://doi.org/10.1007/978-3-642-54886-4_24.
- Smirnova, A. and Cudré-Mauroux, P. (2018) ‘Relation Extraction Using Distant Supervision: A Survey’, *ACM Computing Surveys*, 51(5), p. 106:1–106:35. Available at: <https://doi.org/10.1145/3241741>.
- Tjong Kim Sang, E.F. and De Meulder, F. (2003) ‘Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition’, in *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003. CoNLL-HLT 2003*, pp. 142–147. Available at: <https://aclanthology.org/W03-0419> (Accessed: 6 September 2023).
- Unger, C. et al. (2012) ‘Template-based question answering over RDF data’, *Proceedings of the 21st international conference on World Wide Web*, pp. 639–648. Available at: <https://doi.org/10.1145/2187836.2187923>.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.


Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Retrospektives Klassifizieren und Annotieren von Geschäftsdokumenten im Tagesgeschäft der Denkmalbehörden

3

Christian Fillies , Andreas Giesenhausen und Frauke Weichhardt 

Kernaussagen

1. Durch die formale und fachliche Klassifikation von bestehenden Dokumenten mit Hilfe von Machine Learning (ML) in Kombination mit semantischen Technologien konnte auf eine nachträgliche manuelle Klassifikation der Bestandsdokumente der Berliner Bezirke verzichtet werden.
2. Klassifikation von Dokumenten mit maschinellen Lernverfahren ist ein häufig wiederkehrender Anwendungsfall.
3. Eine naive Anwendung von maschinellen Lernverfahren auf Texten führt oft zu schlechten Ergebnissen.
4. Geschäftsdokumente, wie z. B. Bescheide zu Bauanträgen, enthalten Informationen zu unterschiedlichen Themen, die beim ML alle berücksichtigt werden, ohne dabei jedoch zum Anwendungszweck beizutragen.
5. Semantische Technologien helfen die für den Anwendungszweck relevanten Informationen zu identifizieren und die Ergebnisse des Lernverfahrens zu verbessern.

C. Fillies (✉) · F. Weichhardt
Semtation GmbH, Potsdam, Deutschland
E-Mail: cfillies@semtalk.com

F. Weichhardt
E-Mail: fweichhardt@semtalk.com

A. Giesenhausen
Knowlogy Solutions GmbH, Potsdam, Deutschland
E-Mail: giesenhausen@knowlogy-ag.com

3.1 Einleitung

In vielen Unternehmen sind im Laufe der Jahre umfangreiche Ablagestrukturen entstanden, die nachträglich in Vorgangsverwaltungs- und Dokumentenmanagementsysteme integriert werden sollen, um das enthaltene Wissen weiter nutzen zu können.

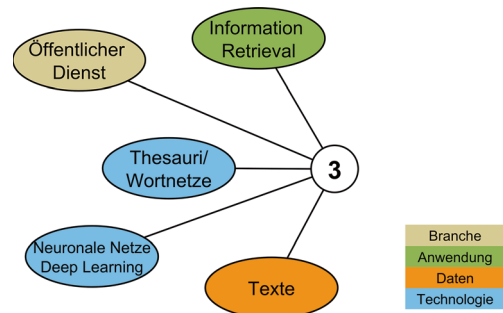
In unserem Kapitel geht es darum, bestehende Textdokumente für das Information Retrieval verfügbar zu machen. Wir setzen dazu sowohl Thesauri und Wortnetze als auch neuronale Netze ein, um Metadaten für die Dokumente aus den Texten zu ermitteln. Abb. 3.1 zeigt eine thematische Einordnung des Kapitels bezüglich der Branche, Anwendung, Daten und Technologien.

Im hier zu Grunde liegenden Anwendungsfall aus dem Bereich Denkmalschutz ist es wichtig, auch nach Jahren Informationen, Dokumente und Unterlagen zu einem Vorgang, einem Projekt oder Geschäftspartner finden zu können. Für eine nachträgliche manuelle Klassifikation oder ein Sortieren sind meistens keine Kapazitäten verfügbar. Die Namenskonventionen und Ablagestrukturen sind oft die einzigen Metadaten, die verfügbar sind. Es gibt bei den Dokument- und Ordnernamen aber individuell je nach Autor sehr große Unterschiede, wie Adressen, alphabetische Sortierung, Dokumenttypen, Datum oder z. B. Denkmalnummer. Auch fachliche Klassifikationen wie Baumfällung, Fassade und Dachausbau sind sowohl für Dokumentnamen als auch für Ordnernamen häufig zu finden. Insbesondere die fachliche Klassifikation der Dokumente ist wichtig und nicht trivial, es ist oft domänenspezifisches Fachwissen zum Verständnis der Klassifikationsstruktur erforderlich.

In einer herkömmlichen Ordnerstruktur muss sich ein Dokument in eine zweidimensionale Klassifikation einordnen. Häufig gibt es aber mehr als zwei Klassifikationsmerkmale, denen ein Dokument zugeordnet werden kann. Dies lässt sich nur mit der Zuordnung verschiedener Metadaten erreichen. Die Fragestellung ist an sich nicht neu, da sie bei jeder Überführung von Bestandsdokumenten in Dokumentenmanagementsysteme mit Metadaten im Mittelpunkt steht.

Abb. 3.2 zeigt die Aspekte hybrider KI, welche in diesem Kapitel beschrieben werden, als Boxology-Modell.

Abb. 3.1 Thematische Einordnung des Kap. 3



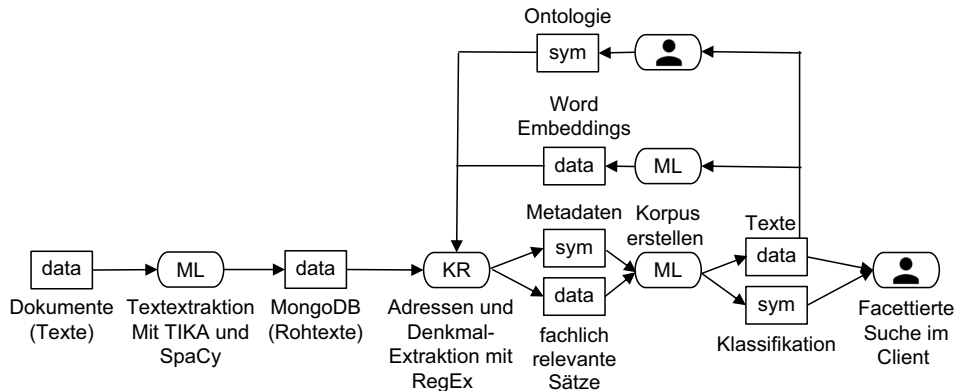


Abb. 3.2 Boxology-Modell des hybriden KI-Systems

Das hier betrachtete Projekt besteht im Wesentlichen aus zwei Teilen: einer klassischen Textanalyse mit einer Vorbereitungsphase mit Tika zur Textextraktion, spaCy zur Textaufbereitung und regulären Ausdrücken zum Erkennen von Adressen und Dokumenttypen wie „Bescheid“, „Versagung“ und „Stellungnahme“. Anhand der bestehenden Berliner Denkmaldatenbank HIDA¹ wurden die Adressen der Denkmale in den Dokumenten gesucht und damit die Denkmale den Dokumenten zugeordnet.

Häufig enthalten die verschiedenen Dokumenttypen unterschiedliche Textbausteine. Erkannte Textbausteine können zur formalen Klassifikation der Dokumente verwendet werden (z. B. Paragraphen von Gesetzen), sind aber für eine semantische (also fachliche) Klassifikation wie Dachausbau, Werbeschild etc. häufig eher störend.

Um eine weitergehende Kategorisierung zu ermöglichen, wurden im zweiten Teil des Projektes aus den Betreffzeilen von 5000 Dokumenten ca. 1000 Substantive extrahiert und im Rahmen eines Workshops 20 Begriffsclustern zugeordnet. Diese Cluster bilden die Dimensionen für eine facettierte Suche auf dem Dokumentenbestand. Aus den aufbereiteten Texten haben wir einen eigenen Korpus erstellt und mit sense2vec² und fast-text³ eigene Word-Embeddings erstellt. Diese ermöglichen eine unschärfere, „semantische“ Zuordnung einzelner Sätze zu Konzepten und damit zu Clustern der Ontologie. Am Ende konnten wir diese Verfahren auch verwenden, um fachlich relevante Zusammenfassungen zu generieren. Für die Ablage der Metadaten und die facettierte Suche wurde im Backend MongoDB verwendet.

¹ <https://www.berlin.de/landesdenkmalamt/denkmale/liste-karte-datenbank/denkmaldatenbank/>

² <https://spacy.io/universe/project/sense2vec>

³ <https://fasttext.cc/>

3.2 Anwendungsfall

Im Fokus der Mitarbeitenden der Berliner Denkmalschutzbehörden stehen die Bearbeitung und Dokumentation von Verwaltungsentscheidungen zu Bau-, Flächen- und Bodendenkmalen im Land Berlin. In den Denkmalschutzämtern sind im Verlaufe der letzten 25 Jahre eine große Anzahl von Dokumenten entstanden, die mit den bisher verfügbaren Methoden der Informationsbereitstellung nicht zu bewältigen sind. Wir beschränken uns auf digitale Dokumente, um den Aufwand in Grenzen zu halten. Die Informationen in den Dokumenten sind größtenteils unstrukturiert und enthalten einen hohen Anteil fachlich nicht relevanter Textbausteine, sodass sie sowohl manuell als auch maschinell nicht trivial zu Denkmalen oder Vorgängen zuzuordnen sind. Wir haben verschiedene Textanalyse (NLP)-Techniken evaluiert und prototypisch umgesetzt. Als effizienteste Vorgehensweise hat sich die Kombination von denkmalschutzspezifischen Worteinbettungen (Word-Embeddings) und einer einfachen Ontologie herausgestellt. Entstanden ist eine automatische Verschlagwortung, die eine intelligente semantische Filterung der Dokumente ermöglicht und damit die tägliche Arbeit der Denkmalschützer mit diesen wesentlich vereinfacht.

3.3 Aufbereitung und Analyse von Texten mit NLP-Methoden

Obwohl es sich hier nur um wenige Geschäftsprozesse wie Anträge und Stellungnahmen handelt, ergibt sich aus einer Geschäftsprozessanalyse eine beträchtliche Anzahl von Dokumententypen. Für jeden Dokumententyp müsste im klassischen ML jeweils ein Klassifikator anhand einer ausreichenden Zahl gelabelter Dokumente trainiert werden. Der Aufwand, auch nur 1000 Dokumente pro Dokumententyp durch Experten zu labeln, scheint nicht adäquat zu sein, da die Prozessbeteiligten (z. B. pensionierte Sachbearbeiter, damalige Antragsteller etc.) in der Regel nicht mehr verfügbar sind. Trotzdem haben wir einige hundert Genehmigungen und Versagungen durch Nicht-Fachexperten (KI-Berater) selbst „gelabelt“ (klassifiziert), um damit ML-Klassifikatoren testen zu können.

Abb. 3.3 vermittelt einen Eindruck über die zu erkennenden Dokumententypen.

Pro Berliner Bezirk liegen zwischen 10.000 und 30.000 Dokumenten in unterschiedlichen Formaten vor: doc, docx, pdf, ppt, jpg und ca. 30 weitere. Die Prozessdokumente liegen meist als Word Dokumente vor. Es finden sich aber auch viele Emails, Abschlussarbeiten, Zeitungsartikel, Studien und viele andere Inhalte, die beim aktuellen Vorgang entweder vorgangsbezogen oder denkmalbezogen relevant sein könnten. Auf die Analyse der Bilder (Fotos von Häusern, Steinen, Lampen, usw.) haben wir zunächst verzichtet, weil diese nur im Vorgangskontext Bedeutung haben.

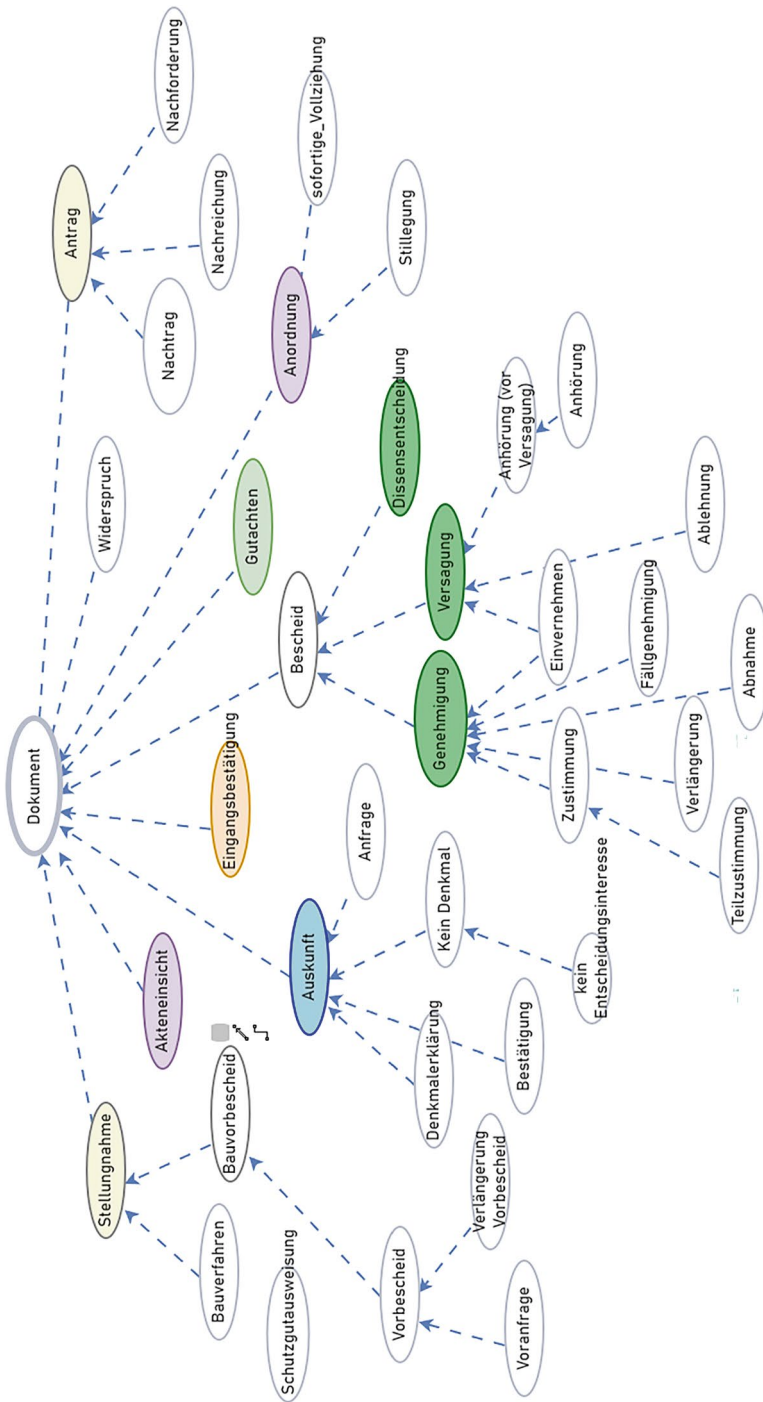


Abb. 3.3 Hierarchie der Dokumenttypen

Mithilfe des Tools Apache Tika⁴ werden Rohtexte und einfache Metadaten aus den Dokumenten extrahiert. Die Ergebnisse werden, wie auch alle später ermittelten Metadaten, in einer NoSQL Datenbank⁵ (MongoDB⁶) abgelegt, sodass weitere Analyse-schritte inkrementell optimiert werden können, ohne Zugriff auf die Originaldokumente zu haben.

Für den Prototyp haben wir Dokumente aus dem Dateisystem verwendet. Für ein Produktivsystem ist die Integration mit dem Dokumentenmanagementsystem der Behörde in Arbeit.

3.4 Analyse von Dokumenten im Denkmalschutz unter Verwendung von Regeln

Reguläre Ausdrücke⁷ beschreiben Suchmuster, die im Prototyp verwendet werden, um Dokumenttypen zu erkennen. Abb. 3.4 zeigt die Benutzeroberfläche für die Verwaltung von regulären Ausdrücken.

Der Aufwand zur Erstellung solcher regulären Ausdrücke ist im Vergleich zum Training mit gelabelten Dokumenten recht überschaubar und erzielt sinnvolle Ergebnisse für Prozessdokumente, die gerade im öffentlichen Bereich viele feste Phrasen und Gesetzesbezüge enthalten. Die regulären Ausdrücke werden sowohl zur Analyse der eigentlichen Texte, als auch zur Analyse der Dokument- und Ordnernamen verwendet. Es ist deshalb notwendig, die Originalordnerstruktur mit in den Metadaten zu kodieren, obwohl sie im Dokumentenmanagementsystem eigentlich nicht mehr verwendet wird.

Die Zuordnung von Denkmalen anhand von Denkmalbezeichnern erschien uns zunächst als ein Anwendungsfall für Named-Entity-Recognition (NER). In der Praxis erwies es sich als wesentlich erfolgreicher, die Adressdaten aus der Denkmaldatenbank HIDA⁸ zur Zuordnung von Dokumenten zu Denkmalen zu nutzen. Ein „Levenshtein distance“⁹-Werkzeug half, Tippfehler teilweise zu eliminieren. Aus der Denkmaluordnung lassen sich viele indirekte Selektionskriterien für die Dokumente ableiten wie Bezirk, Bauherr und der Denkmaltyp. Für die Denkmaltypen gibt es in der HIDA eine große Taxonomie aus architektonischer Sicht, die zur Filterung bei der Suche eingesetzt werden kann. NER kann eventuell in Zukunft eine Rolle spielen, um spezifische Geschäftspartner wie Antragsteller oder Behörden in den Texten zu identifizieren.

⁴<https://tika.apache.org/>

⁵<https://de.wikipedia.org/wiki/NoSQL>

⁶<https://www.mongodb.com/>

⁷https://de.wikipedia.org/wiki/Regul%C3%A4rer_Ausdruck

⁸<https://www.berlin.de/landesdenkmalamt/denkmale/liste-karte-datenbank/denkmaldatenbank/>

⁹https://en.wikipedia.org/wiki/levenshtein_distance

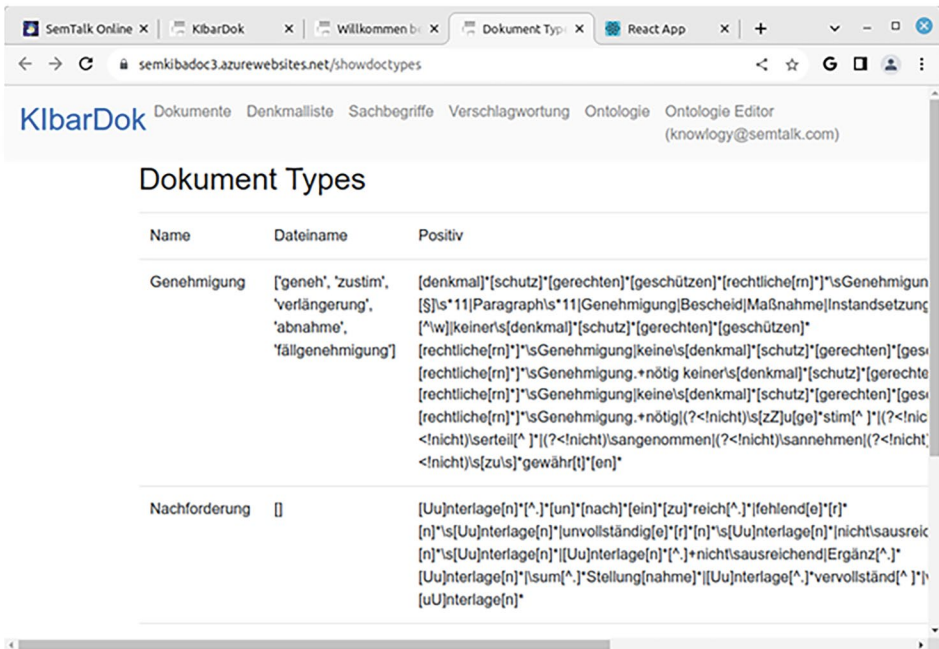


Abb. 3.4 Benutzeroberfläche für die Verwaltung von regulären Ausdrücken zum Erkennen von Genehmigung und Nachforderung

Falls Textbausteine (Abb. 3.5) im Text gefunden werden, geben sie entscheidende Hinweise auf den Dokumenttyp.

Textbausteine sind aber auch nützlich, um Hypothesen für die Vorhabensbeschreibung zu erkennen. Diese stehen an den durch XXX (resp. YYY) markierten, variablen Stellen im Textbaustein. Für eine fachliche Klassifikation der Dokumente mit ML sind die Textbausteine eher kontraproduktiv, da sie im Verhältnis zur Vorhabensbeschreibung einen großen Anteil des Textes ausmachen.

Für eine regelbasierte Extraktion gibt es in den Geschäftsprozessen der Denkmalschutzämter noch viele bisher ungenutzte Potenziale, die leicht genutzt werden könnten. Ein Beispiel ist der Umgebungsschutz: „Werbeschild X am Supermarkt Y verdeckt Denkmal Z“ ist ein häufig auftretender Fall, bei dem die Adresse des Vorhabens nicht identisch mit der Denkmaladresse ist. Weitere regelbasierte Analyseoptionen zwischen Dokumenten sind derselbe Ordner oder Unterordner, ähnliche Dokumentnamen (Vorhaben) oder ähnliches Datum.

Die Anwender suchen in Dokumentenmanagementsystemen nach Informationen in sogenannten „Vorgangsmappen“, also der Menge der bei einer Prozessausführung

The screenshot shows a web browser window with the URL `semkibadoc3.azurewebsites.net/showpattern`. The page title is "KibarDok" and the navigation menu includes "Dokumente", "Denkmalliste", "Sachbegriffe", "Verschlagwortung", "Ontologie", and "Ontologie Editor (knowlogy@semtalk.com)". The main content area is titled "Textbausteine" and displays a list of six text blocks, each with a yellow "Bearbeiten" button on the left. The text blocks contain legal or administrative information with variable placeholders (XXX, YYY).

Bearbeiten	Kategorie	Textinhalt
Bearbeiten	Versagung	Aus denkmalrechtlicher Sicht XXX nicht genehmigungsfähig.
Bearbeiten	Genehmigung	Aus denkmalrechtlicher Sicht wird XXX zugestimmt.
Bearbeiten		Das Einvernehmen gemäß § 12 Abs. 3 DSchG Bln wird in Verbindung mit § 11 Abs. 1 Denkmalschutzgesetz Berlin (DSchG Bln) vom 24.04.1995 (GVBl. 22 S. 274), zuletzt geändert durch Artikel II des Gesetzes vom 14. Dezember 2005 (GVBl. S. 755) XXX erteilt.
Bearbeiten		Das Einvernehmen gemäß § 12 Abs. 3 DSchG Bln wird in Verbindung mit § 11 Abs. 1 Denkmalschutzgesetz Berlin (DSchG Bln) vom 24.04.1995 (GVBl. 22 S. 274), zuletzt geändert durch Artikel II des Gesetzes vom 8. Juli 2010 (GVBl. S. 396) XXX erteilt.
Bearbeiten		Das beabsichtigte Vorhaben sieht XXX in der unmittelbaren Nähe YYY vor und ist als Veränderung der unmittelbaren Umgebung im Sinne von § 11 Abs. 2 DSchG Bln zu qualifizieren.
Bearbeiten		Das beantragte Vorhaben XXX ist nach DSchG Bln genehmigungsbedürftig.

Abb. 3.5 Beispiele für Textbausteine mit Variablen

entstandenen Dokumente. Zu einem Antrag gibt es zum Beispiel eine Anhörung und eine Versagung. Die Rekonstruktion des Vorgangs aus den vorhandenen unstrukturiert abgelegten Dokumenten ist das eigentliche Ziel des suchenden Anwenders.

Abb. 3.6 zeigt ein Modell des Ablaufs der Antragsbearbeitung in der denkmalschutzrechtlichen Prüfung. Es beschreibt, welche Dokumententypen an welchen Stellen des Vorgangs verwendet werden: Dokumente (hellgelb), Rollen (weiss), idealer Prozesspfad (blau) und Ausnahmen und Eskalation (grün).

Mithilfe solcher Prozessmodelle gelingt es, ein gemeinsames Verständnis zwischen dem KI-Team und den Anwendern herzustellen, wonach eigentlich gesucht bzw. trainiert werden soll und welche Art von Klassifikation (u. a. wirtschaftlich) sinnvoll zu erstellen ist.

Logische Abhängigkeiten zwischen Dokumenten können durch Vergleich mit einem maschinenlesbaren Prozessmodell analysiert werden, denn es beschreibt, welche Dokumente und Geschäftsobjekte bei welchen Arbeitsschritten erstellt werden. Beispielsweise kann eine Versagung nur nach einem Antrag erstellt werden. Hilfreich ist es auch zu untersuchen, welche Rollen welche Arten von Dokumenten erstellen. Was kommt vom

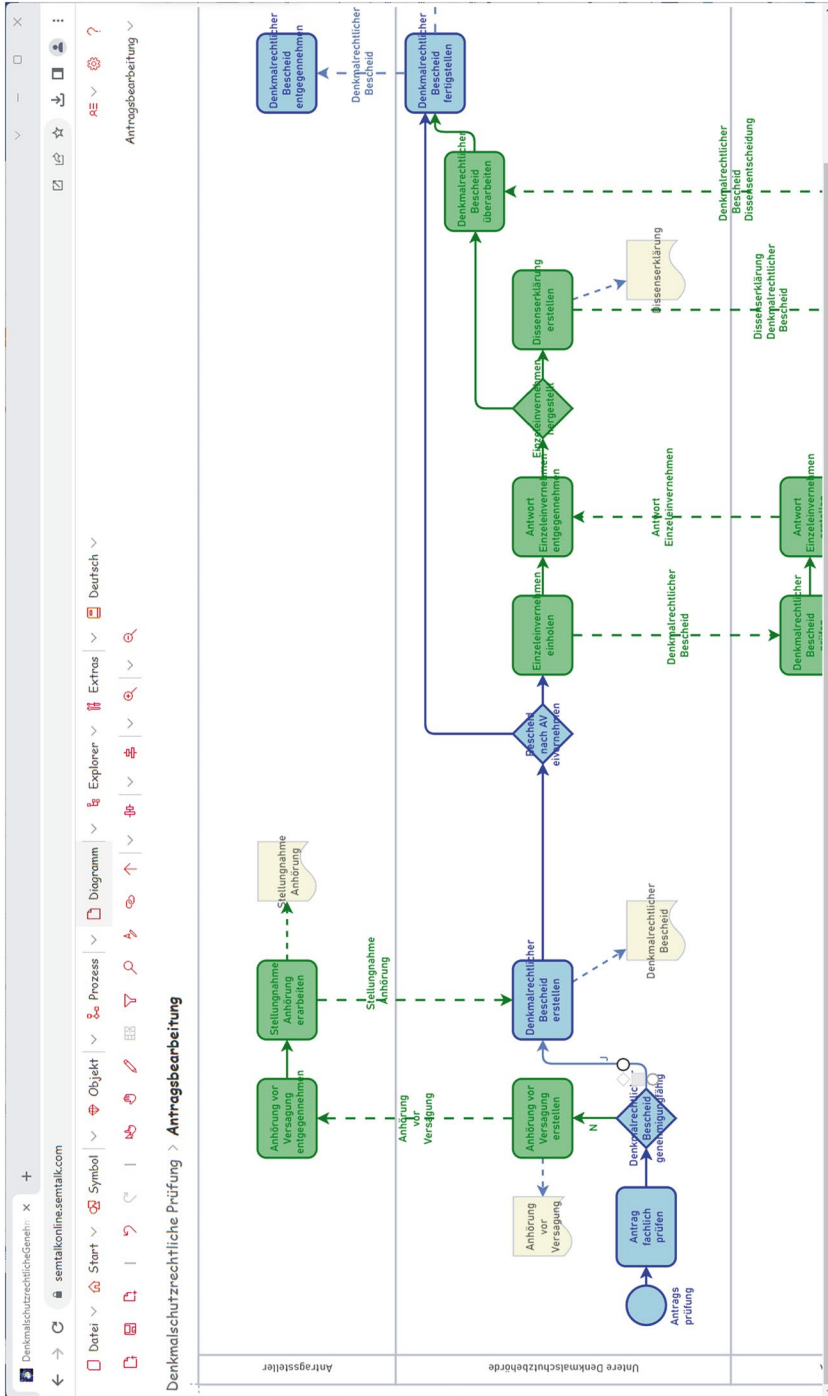


Abb. 3.6 Prozessmodell „Denkmalschutzrechtliche Prüfung“

Landesdenkmalamt und was von einer bestimmten Wohnungsbaugesellschaft? Aus den Prozessmodellen lassen sich viele fachliche Suchmuster oder Named-Entities erkennen, die bei der Klassifikation nutzbar sind.

3.5 Analyse der fachlichen Zuordnung auf der Grundlage ontologischer Informationen

Ergebnis des hier betrachteten Projekts war die Darstellung und Suche der Dokumente in einer facettierten Suche. Diese findet unter anderem auf Basis eines hierarchisierten Suchbegriffskatalogs statt. Beispiele hierfür findet man bei Lösungen auf Basis von Elastic Search¹⁰ oder im Microsoft SharePoint-Umfeld bei Verwendung des Terminologie-Speichers (TermStore)¹¹.

Die automatische Verschlagwortung mit Suchbegriffen aus dem Suchbegriffskatalog der Dokumente ist eine nicht triviale Aufgabe, da die Domäne denkmalschutzrelevanter Themen zwar begrenzt, aber sehr groß ist. Ob es sich bei einer Stellungnahme zu einer Baumfällung, zur Anbringung einer Werbetafel oder zur Umnutzung eines historischen Industriegebäudes handelt, ist ein sehr weites Feld. Davon abgesehen ist der Anteil des Textes in den Dokumenten, der die Vorhabensbeschreibungen enthält, sehr klein. Strukturiert man die Suchbegriffe, implizieren diese Suchbegriffe eine fachliche Klassifikation der Dokumente. Aus dem Clustering der Suchbegriffe ergeben sich Gruppierungen von Dokumenten.

Folgendermaßen sind wir vorgegangen: Die Texte werden zunächst mit spaCy in Sätze zerlegt, um Stoppworte bereinigt und lemmatisiert, d. h. auf Grundformen reduziert.

Wir haben dann versucht, mit LDA Topic Modelling Themen zu extrahieren, auch unter Zuhilfenahme externer Dokumentensammlungen. Die Ergebnisse trafen nicht unsere Erwartung, da zu viele fachlich nicht relevante Wörter Teil der Ergebnisse waren. Die Ergebnisse des Topic Modelling waren wenig hilfreich, da die relevanten Themengebiete im Grunde bekannt waren, es galt nur, die Dokumente diesen Themengebieten maschinell zuzuordnen, und nicht, diese zu entdecken.

Wir sind dann einen teilweise manuellen Weg gegangen: Aus dem Dokumentenbestand eines Bezirks haben wir aus den Betreffzeilen der Prozessdokumente ca. 1000 Substantive ausgeleitet und diese mit Fachexperten innerhalb von 2 Tagen in 10 Gruppen geclustert. Der entstandene Graph ist mehr ein „hat-zu-tun-mit“-Beziehungsnetz als eine „echte“ Ontologie, obwohl er Ober- und Unterbegriffe enthält (siehe Abb. 3.7).

Wir vergleichen die Texte auf Satzebene mit dieser „Ontologie“, um fachlich relevante Teile der Texte zu ermitteln. Aus den gefundenen Begriffen können wir auf die

¹⁰ <https://www.elastic.co/>

¹¹ <https://docs.microsoft.com/en-us/sharepoint/managed-metadata>

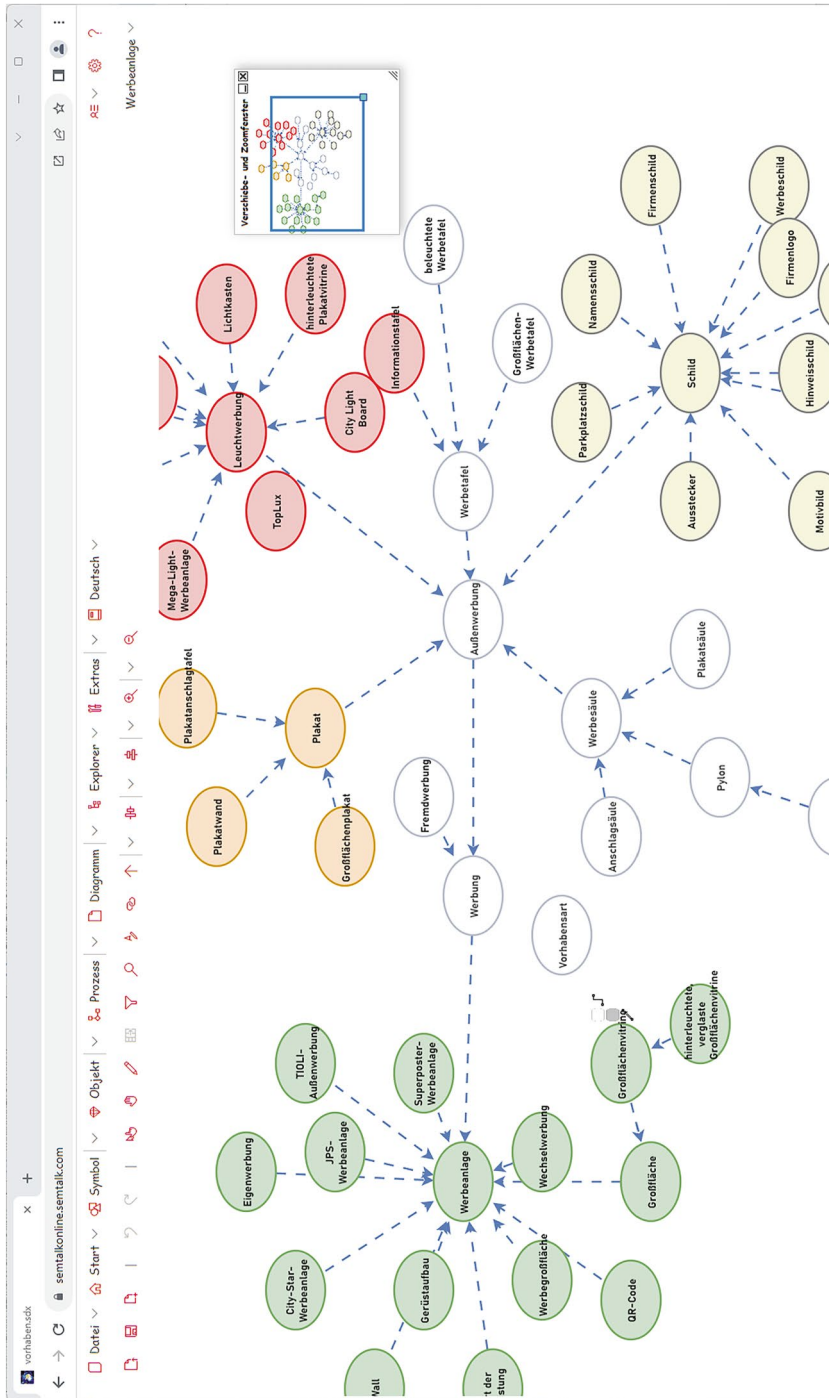


Abb. 3.7 Erste Begriffscluster

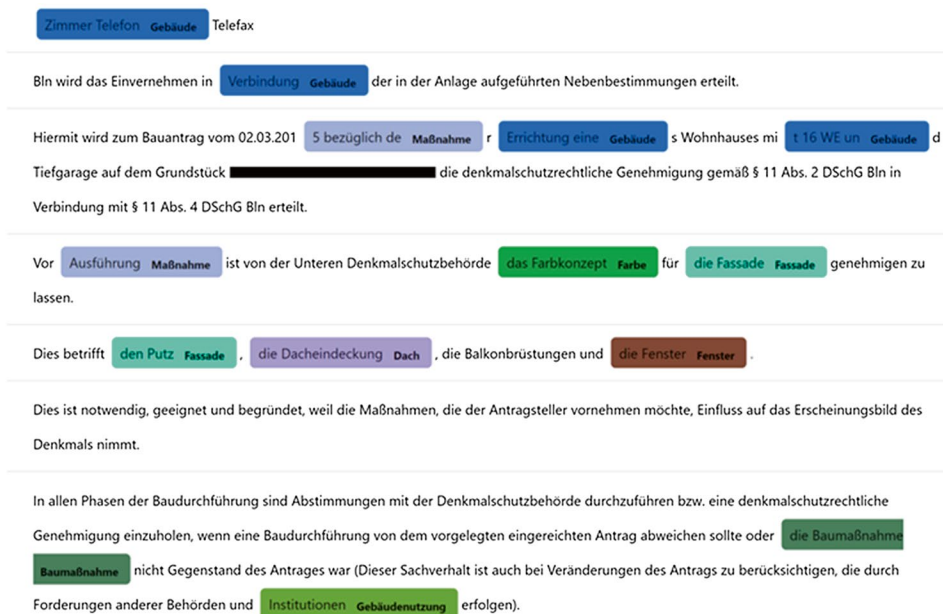


Abb. 3.8 Beispiele für analysierte Textstellen

Kategorie, z. B. „Werbemittel“, schließen und somit Metadaten auf Satzebene zuordnen. In Abb. 3.8 werden die erkannten Zuordnungen im Text visualisiert.

Schnell zeigte sich, dass die reine Verschlagwortung mit Suchbegriffen nicht ausreichend ist. Wir haben dann Worteinbettungen (Word-Embeddings) aus dem deutschen Standardkorpus in spaCy („core_news_md“, TIGER Corpus) verwendet um auch semantisch ähnliche Begriffe zu erkennen. Ein ähnliches Vorgehen allerdings mit wesentlich mehr NLP Technologie findet sich bei Melanie Siegel (Siegel 2018) bei der Analyse von betriebswirtschaftlichen Jahresabschlüssen.

Word-Embeddings sind ein Verfahren, ursprünglich von Google entwickelt, bei dem ein neuronales Netz (Autoencoder) so lange trainiert wird, bis die semantische Nähe zweier Begriffe ihrem Abstand in einem multi-dimensionalen Vektorraum entspricht und somit quantifizierbar ist. Mit Word-Embeddings können wir semantisch ähnliche Worte den Suchbegriffen aus unserem Katalog zuordnen. Dieses Verfahren zeigte teilweise überraschend gute Zuordnungen wie z. B. „Kubatur“ => „Dachausbau“, die nicht in unserem Suchbegriffskatalog enthalten sind. Eine detaillierte Beschreibung der Verwendung von Word-Embeddings in der semantischen Suche findet sich in diesem Buch in Kap. 6.

Nun kann man von der semantischen Nähe zweier Begriffe in deutschen Zeitungsnachrichten nicht unbedingt auf die semantische Nähe in denkmalschutzrelevanten

Dokumenten schließen. Es gibt bei dieser Vorgehensweise noch recht viele falsche Zuordnungen.

Wir haben alle verfügbaren Dokumente – zu dem Zeitpunkt etwa 50.000 Dokumente – verwendet, um eigene Word-Embeddings auf der Basis eines Denkmalschutzkorpus zu trainieren. Nach einer Bereinigung um die Textbausteine, bis auf die wichtigen Variablenwerte „XXX“ (siehe Abb. 3.5), und einer Reduktion auf Nominalphrasen mit SpaCy’s POS Tagger haben wir das Modell mit dem spaCy-addon sense2vec und fast-Text (facebook AI) in einigen Stunden trainiert und dann in unsere spaCy-Infrastruktur integriert.

Daraus resultiert ein denkmalschutzspezifischeres Ähnlichkeitsmaß für Begriffe, das wir für die Verschlagwortung nutzen. Der Vergleich von Textstellen mit Begriffen aus der Ontologie wird so bei Erstellungen von Metadaten fachspezifischer und damit besser. Auch mit der geringen Anzahl von Dokumenten aus fünf Bezirken hat sich durch das Training die Qualität der zugeordneten Metadaten aus Sicht unserer Anwender deutlich verbessert. Eine quantitative Aussage über die Qualität von Ähnlichkeit ist schwer zu treffen.

Das so ermittelte fachliche Relevanzmaß einzelner Sätze können wir leicht nutzen, um damit automatische Zusammenfassungen der Dokumente zu generieren, die später in der Suchoberfläche dem Anwender einen schnellen Eindruck vermitteln, worum es in dem Dokument eigentlich geht.

Damit ist für unsere Anwender eine wertvolle Lösung zum Filtern und Suchen in Bestandsdokumenten entstanden ohne den Aufwand, Beispiele für alle Dokumententypen zu labeln oder gar Beispiele für fachliche Metadaten wie „Werbeshire“, „Fassade“ usw. in großer Zahl zu erstellen. Der manuelle Aufwand wäre immens und der Dokumentenbestand am Ende dann doch wahrscheinlich zu klein gewesen. Eine seriöse Abschätzung des Verhältnisses von Aufwand und Nutzen erscheint kaum möglich. Der wesentliche Unterschied zu einem klassischen ML Klassifizierer ist dementsprechend bei unserem Ansatz, dass die Optimierung über die Anpassung der Ontologie und die Verbesserung der Word-Embeddings erfolgt und nicht über eine aufwendige Auszeichnung von Beispieldokumenten.

Ursprünglich geplant war es, die automatisch generierten Metadaten zu verwenden, um damit klassische ML-Klassifizierer zu trainieren, also statt manuellem Labelling die automatisch generierten Metadaten zu verwenden. Diese Option wurde zurückgestellt, da Suche und Filterung mit den gegebenen Metadaten als hinreichend erschienen. Die Generierung solcher Klassifizierer mag aber sinnvoll sein, wenn man sie in der operativen Digitalisierung von Geschäftsprozessen einsetzen möchte: Der Workflow kann einen anderen Prozesspfad beim Antrag auf Baumfällung wählen als beim Anbringen eines Werbeschildes.

Nächster Schritt in Richtung Produktivsystem wäre hier die Möglichkeit der manuellen Nachbearbeitung der automatisch erstellten Verschlagwortung sowie die inkrementelle Verbesserung von Stoppwortlisten, Textbausteinen und insbesondere der Ontologie.

3.6 ML gestützter Aufbau einer Ontologie

Die initiale Suchbegriffsliste oder Ontologie stammt aus den Betreffzeilen eines relativ kleinen Teils der Dokumente. Die grobe Gliederung der Dimensionen, nach denen später in der facettierten Suche gruppiert werden kann, wurde mit den Anwendern abgestimmt. Um die Domäne leichter erschließen zu können, haben wir das Onlinemodellierungs-Tool SemTalk¹² verwendet, mit dem sich Wissensgraphen leicht erstellen lassen. Eine überschaubare Anzahl von Konzepten und ihre Beziehungen werden dabei im Stile einer Mindmap je Szenario visualisiert und modelliert.

Das Ziel der Weiterentwicklung der Ontologie ist es, die Dokumente möglichst präzise den Themenfeldern zuzuordnen. Es treten sicher einige „false-positives“ auf. Um diese zu minimieren, kann die Ontologie zum Beispiel durch mehr modellierte Begriffe verbessert werden.

Um die Ontologie-Entwicklung möglichst gut zu unterstützen, nutzen wir wieder unser trainiertes Domänenmodell mit den Word-Embeddings, allerdings im umgekehrten Sinne.

Zu einem gewählten Konzept im Graph wird eine Liste von 20 möglichst semantisch ähnlichen Begriffe aus dem trainierten Modell angezeigt. Abb. 3.9 zeigt, wie auf der rechten Seite zum ausgewählten Konzept „Abdichtungsarbeiten“ verwandte Konzepte wie „Dachabdichtungsarbeiten“, „Beschichtungsarbeiten“, „Absteifungsarbeiten“ usw. vorgeschlagen werden. Die Entscheidung, in welcher Beziehung die Begriffe zueinander stehen, wird dabei dem Modellierer überlassen.

Konzepte, die Word-Embeddings haben, werden schattiert dargestellt. Die gestrichelten Linien deuten in die Richtung des explizit spezifizierten Oberbegriffes. Die durchgezogenen Linien stellen ähnliche Konzepte im trainierten Modell dar. Die Dicke der Kanten steht in Relation zur semantischen Nähe. Durch die parametrisierbare Visualisierung der semantischen Nähe unterstützt das trainierte Domänenmodell die Erstellung der Ontologie auf Basis der im Training verwendeten Dokumente.

3.7 Prototypische Suchoberfläche

Um die erzeugten Metadaten zu prüfen, ist es notwendig, eine Filter- und Suchfunktionalität auf Basis der ermittelten Metadaten zur Verfügung zu stellen. Die MongoDB Query Language stellt eine leistungsstarke NoSQL Abfragesprache zur Verfügung, mit der eine inkrementelle Filterung nach den verschiedenen Kategorien und somit eine facettierte Suche möglich wird.

Im Projekt wurden drei verschiedene Arten von Suchdimensionen verwendet (Beispiele siehe Abb. 3.10):

¹²<http://www.semtalk.de/>

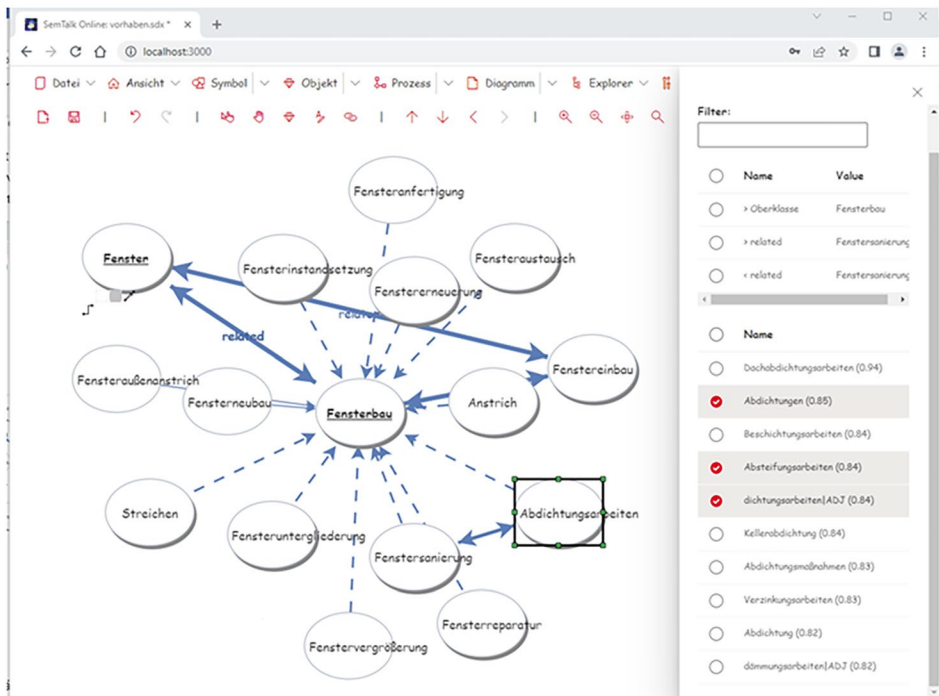
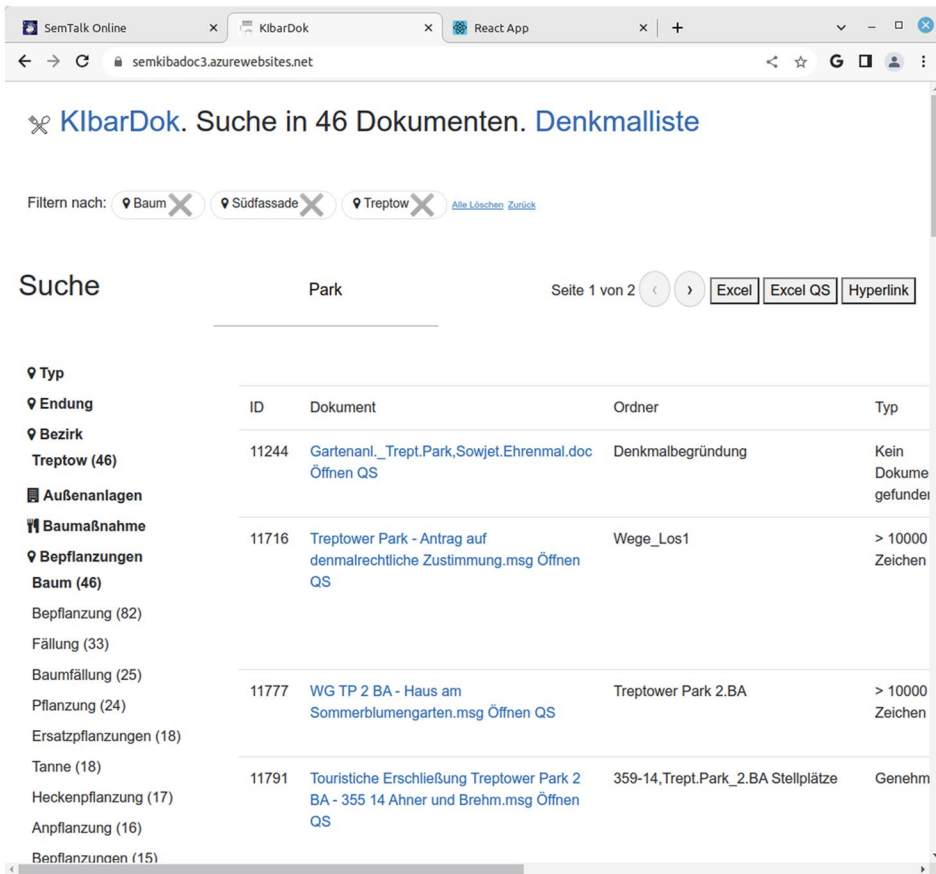


Abb. 3.9 Ontologie-Editor mit Semantikcheck

- Kategorien, die direkt vom Dokument abgeleitet werden, wie Name, Ordner, Dateityp;
- Kategorien, die indirekt vom evtl. gefundenen Denkmal abgeleitet werden, wie Denkmalart oder der sogenannte „Sachbegriff“, der die Einordnung in die HIDA-Taxonomie beschreibt;
- Fachliche Dimensionen wie Farbe, Dach, Bepflanzungen usw. aus der Ontologie.

Um auch nach Oberbegriffen filtern zu können, werden sowohl bei den sogenannten „Sachbegriffen“ aus der HIDA-Taxonomie als auch bei der von uns erstellten Ontologie alle Oberbegriffe hinzugefügt. Wenn also „Akazie“ in den Metadaten steht, kommen noch aus der Ontologie „Baum“ und „Bepflanzungen“ hinzu. Somit wird bei der Suche nach „Baum“ auch ein Dokument gefunden, das „Akazie“ aber nicht „Baum“ enthält. Das Verfahren ist eine Standardvorgehensweise zur semantischen Suche Hoppe (2020) und wird u. a. auch bei Microsoft Sharepoint empfohlen. Die HIDA-Taxonomie wird nur zur nachträglichen Filterung der Dokumente verwendet, nicht im Rahmen der hier beschriebenen technischen Generierung von deren Metadaten.

Schnell zeigt sich, dass zur Evaluierung ein Zugriff auf die Dokumente und eine Volltextsuche im Textkörper und in den verschiedenen Spalten notwendig ist. MongoDB erstellt auf Basis von Apache Lucene einen Volltextindex.



Suche in 46 Dokumenten. Denkmalliste

Filtern nach: Baum Südfassade Treptow [Alle Löschen Zurück](#)

Suche Park Seite 1 von 2 [Excel](#) [Excel QS](#) [Hyperlink](#)

ID	Dokument	Ordner	Typ
11244	Gartenanl._Trept.Park,Sowjet.Ehrenmal.doc Öffnen QS	Denkmalbegründung	Kein Dokume gefunden
11716	Treptower Park - Antrag auf denmalrechtliche Zustimmung.msg Öffnen QS	Wege_Los1	> 10000 Zeichen
11777	WG TP 2 BA - Haus am Sommerblumengarten.msg Öffnen QS	Treptower Park 2.BA	> 10000 Zeichen
11791	Touristische Erschließung Treptower Park 2 BA - 355 14 Ahner und Brehm.msg Öffnen QS	359-14,Trept.Park_2.BA Stellplätze	Genehm

Typ

Endung

Bezirk
Treptow (46)

Außenanlagen

Baumaßnahme

Bepflanzungen
Baum (46)
Bepflanzung (82)
Fällung (33)
Baumfällung (25)
Pflanzung (24)
Ersatzpflanzungen (18)
Tanne (18)
Heckenpflanzung (17)
Anpflanzung (16)
Bepflanzungen (15)

Abb. 3.10 Erster Prototyp zur facettierten Suche

Mit einer externen Adressensuche haben wir die Anwender schließlich endgültig überzeugt: Mit Reverse Geocoding z. B. von Bing haben wir die in den Dokumenten gefundenen Adressen genutzt, um die aktuell gefilterten Dokumente in OpenStreetMap mithilfe des Mapbox APIs zu visualisieren (Ausschnitt daraus siehe Abb. 3.11)

Es gibt in diesem Prototyp eine semantisch generierte Zusammenfassung der Texte und direkte Links zu Google Street View und dem Geoinformationssystem der Denkmalschützer, dem FIS-Broker¹³.

Die Oberste Berliner Denkmalbehörde (Senatsverwaltung für Kultur und Europa) wurde für das beschriebene Projekt „KibarDok – KI-Einsatz zur automatisierten und

¹³ <https://www.stadtentwicklung.berlin.de/geoinformation/fis-broker/>

KlbarDok. Suche in 1419 Dokumenten. Denkmalliste

Am Tip P 70.Trept Park Voranfrag KITA.doc

2. Dem Ausbau des Daches mit seiner Konsequenz eines 2. Rettungsweges, wird nicht zugestimmt. 3. Der Errichtung der Dachgaube wird in Konsequenz aus 1. und2. nicht zugestimmt. v. 14. 06. 2001) als Gartendenkmal aufgeführt. Erweiterung im Gartendenkmal Dies gilt auch für das Zubehör und die Ausstattung eines Denkmals. Der Volkspark dient auch heute in erster Linie noch den einst von seinem Schöpfer Gustav Meyer konzipierten Erholungsformen, wie der ruhigen Erholung, dem kontemplativen Naturgenuss bei Spaziergängen oder beim Lagern auf ausgedehnten Wiesenflächen sowie der spielerischen und sportlichen Betätigung, ebenso der Unterhaltung und Kommunikation. Alle Nutzungsplanungen haben sich dem Denkmalcharakter des Treptower Parks unterzuordnen, um Verluste der Substanz oder Wirkungen der Denkmals auszuschließen. Nutzungen, die sich nicht ohne Beeinträchtigung des Denkmalcharakters des Treptower Parks einordnen, können keine Zustimmung finden. Das Gebäude der KITA wurde als Gebäude für d ...

ID	Datel	Typ	Vorhaben	Zusammenfassung	Adressen	Denkmal
1199	Trept Park Voranfrag KITA.doc	Anfrage	Umbau und Erweiterung einer KITA		am treptower park 70	Treptower Park
96	Am Trip P 70.Trept Park Voranfrag KITA.doc	Versagung	Umbau und Erweiterung einer KITA	2. Dem Ausbau des Daches, mit seiner Konsequenz eines 2. Rettungsweges, wird nicht zugestimmt. 3. Der Errichtung der Dachgaube wird in Konsequenz aus 1. und2. nicht zugestimmt. v. 14. 06. 2001) als Gartendenkmal	am treptower park 70	Gartendenkmal

Abb. 3.11 Facettierte Suche mit Kartensansicht

barrierefreien Erschließung von Dokumenten“ mit dem Berliner Verwaltungspreis 2022¹⁴ ausgezeichnet.

3.8 Empfehlungen

Basierend auf unseren Projekterfahrungen geben wir nachfolgende Empfehlungen für ähnliche Projekte.

1. Komplexe Informationsextraktionsaufgaben können von einer Domänen-Ontologie profitieren.
2. Eine NLP-Analyse von Texten der Domäne ist nützlich, um die Ontologie zu organisieren und relevante Wörter und Konzepte zu finden. Methoden der natürlichen Sprachverarbeitung sind für diesen Zweck von großem Wert.
3. Es ist sehr hilfreich, schon in frühen Phasen des Projektes echte Anwendungen statt Mockups zu entwickeln, um die Potenziale von KI-Lösungen gemeinsam mit dem Anwender bewerten zu können.
4. Es ist von großer Bedeutung, die Geschäftsprozesse, Fallzahlen, Dokumenttypen und Dokumentstrukturen daraufhin zu prüfen, welche Art von ML mit welchem Aufwand auf welche Weise sicher zum Ziel führt.

3.9 Zusammenfassung

Im Hinblick auf die Frage „Können KI-Methoden Denkmalschützer bei ihren Analysen effektiv unterstützen?“ haben wir mithilfe von NLP-Systemen prototypische Implementierungen zur Analyse von Bestandsdokumenten vorgenommen.

Wir haben die Dokumenttypen und Adressen in den Texten mit Regeln analysiert, Informationen daraus extrahiert und sie in eine strukturierte Form gebracht. Denkmalschützer erhalten über Dokumenttyp und Denkmal einen schnelleren Zugriff auf die bisher nicht klassifizierten Dokumente. Schon diese Informationsextraktion ist also für sich genommen wertvoll.

Ein weiterer Schwerpunkt lag auf der Extraktion fachlicher Informationen aus den Texten. Um die Suche nach für den aktuellen Geschäftsvorfall relevanten Informationen effektiv zu unterstützen, wurde eine Ontologie erstellt. Für die Entwicklung dieser Ontologie wurden verschiedene NLP-Techniken angewandt, wie z. B. Terminologieextraktion.

Es wurde eine Vorgehensweise zur Erstellung von fachlichen Metadaten aus Bestandsdokumenten entwickelt und evaluiert, die auf eine manuelle Durchsicht und

¹⁴<https://www.berlin.de/rbmskzl/aktuelles/pressemitteilungen/pressemitteilung.1277873.php>

Klassifikation der Dokumente verzichtet und trotzdem die Denkmalschützer effektiv unterstützen kann. Nun gilt es, diese Techniken weiterzuentwickeln, anzupassen und mit bestehenden Dokumentenmanagement-Systemen zu integrieren.

Die wichtigste offene Frage am Ende ist: Wie steht es um die Übertragbarkeit des Ansatzes, mithilfe von Worteinbettungen und einer kleinen Ontologie sinnvolle fachliche Metadaten zu erzeugen? Für Bauanträge erscheint es auf den ersten Blick sinnvoll. Aber für welche Arten von Dokumentenablagen in der Verwaltung und anderen Unternehmen gilt dies noch?

Literatur

- Hoppe, T. (2020) Semantische Suche – Grundlagen und Methoden semantischer Suche von Textdokumente, Springer Vieweg, Wiesbaden, 2020. ISBN 978-3-658-30426-3 (mit Beiträgen von Bernhard Humm).
- Siegel, M. (2018), Text mining in economics, in A. Reibold, T. Hoppe & B. Humm, ed., 'Semantic Applications. Methodology, Technology, Corporate Use', Springer-Verlag, Berlin, Heidelberg.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Text-getriebener Aufbau von domänenspezifischen Wissensgraphen mit neuronalen Netzen

4

Felix Hamann 

Kernaussagen

1. Die Erstellung von Wissensgraphen für spezifische Domänen ist ein kostenintensiver Prozess (Personal).
2. Spezifisches Domänenwissen kann aus Textdokumenten extrahiert werden, die im operativen Geschäft anfallen.
3. Die Extraktion von Entitäten und Fakten aus diesen Daten ist schwierig, kann aber mit Unterstützung moderner Deep-Learning-Modelle für Text-Mining geleistet werden.
4. Dieses Kapitel schlägt einen Annotationsprozess vor, in dem Deep-Learning-basierte Faktendetektion interaktiv erfolgt.
5. Der Prozess sieht ein Vorschlags-Tooling vor, welches es dem Domänenexperten erlaubt, niederschwellig neue und relevante Fakten, die durch die KI vorgeschlagen werden, in den Wissensgraphen zu übernehmen.
6. Die vorgestellten Techniken und eingesetzten Technologien basieren auf dem aktuellen Stand der Forschung und sind noch in einem prototypischen Zustand.
7. Ihre Anwendbarkeit wird anhand zweier praktischer Use Cases im technischen Service und in der Analyse von Schadensbeschreibungen aus dem Versicherungswesen diskutiert.

F. Hamann (✉)
Hochschule RheinMain, Wiesbaden, Deutschland
E-Mail: felix.hamann@hs-rm.de

4.1 Einleitung

Ein effektives Wissensmanagement ist nach wie vor eine große Herausforderung für Unternehmen, gleich welcher Größe. Informationen, die die tägliche Arbeit innerhalb einer Organisation betreffen – seien es technisches Know-how, geschäftliche Abläufe oder Daten des Kundensupports –, sollen einfach wiederverwendbar und niederschwellig abrufbar sein. Dieses Themenfeld gewinnt angesichts wachsenden Fachkräftemangels massiv an Bedeutung und moderne, häufig softwaregetriebene Ansätze benötigen zugreifbare digitale Informationen. Dies ist, quer durch alle Branchen, notwendig, um im digitalen Zeitalter konkurrenzfähig zu bleiben.

Die Idealvorstellung effizienten Wissensmanagements scheidet jedoch häufig an den vielen Herausforderungen, die sich beim Aufbau einer Wissensbasis stellen. Diese sind einerseits organisatorischer Natur: Häufig sind Geschäftsprozesse und ihre Daten nicht durchgängig digitalisiert und Wissensinseln müssen zusammengeführt werden, die von einzelnen Mitarbeitenden oder Abteilungen bis zu heterogenen unternehmensweiten Datensilos reichen. Das übergeordnete Problem ist aber der erhebliche Aufwand der Wissensstrukturierung und -erfassung, insbesondere da hierfür Domänenexperten (welche die Relevanz verschiedener Informationen beurteilen können) und Digitalisierungsexperten (welche das Know-how für die technische Wissensrepräsentation besitzen) zusammenarbeiten müssen.

Gegenstand dieses Kapitels ist deshalb ein Überblick über Technologien, die den Prozess der Wissenserfassung unterstützen und effizient gestalten. Ein besonderes Augenmerk liegt hierbei auf neuronalen Modellen des Textverstehens (Large Language Models – LLM). Diese sind seit einigen Jahren in der Forschung etabliert und ihre Ausdrucksmächtigkeit ist (im Zuge von ChatGPT) inzwischen einem breiten Publikum bekannt. Das Potenzial dieser Technologie für die Erfassung von Wissensbasen in spezifischen Unternehmenskontexten wird hier beleuchtet. Das Kapitel beginnt mit einer Einführung in Wissensgraphen. Anschließend wird ein Prozess zur opportunistischen, datengetriebenen Erweiterung von Wissensgraphen vorgestellt.

Abb. 4.1 ordnet das Kapitel thematisch ein bezüglich der Branche, Anwendung, Daten sowie Technologien.

Abb. 4.1 Thematische Einordnung des Kap. 4

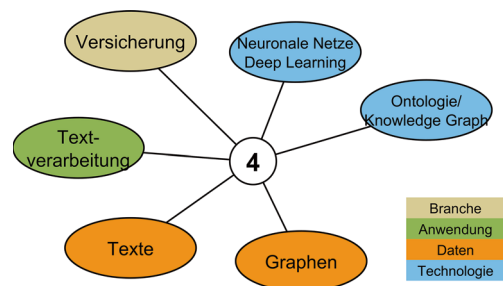


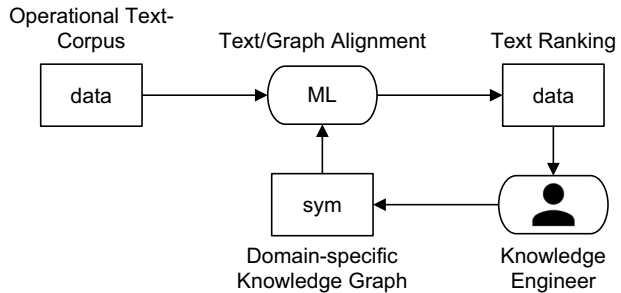
Abb. 4.2 Boxology-Modell des hybriden KI-Systems

Abb. 4.2 zeigt das Boxology-Modell¹ des hybriden KI-Systems. Es gibt auch einen Überblick über den präsentierten Arbeitsablauf: Text und Graphen werden miteinander verknüpft und einem Knowledge Engineer präsentiert, um neues Wissen dem Wissensgraphen hinzuzufügen. Nach einer kurzen Erläuterung gängiger Deep-Learning-Ansätze für Text-Mining und Wissensgraphvervollständigung schließt das Kapitel mit der Präsentation von zwei konkreten Use Cases im betrieblichen Umfeld.

4.2 Stand von Forschung und Technik

Eine weitverbreitete, niedrighschwellige Möglichkeit, um textuelle Informationen zugreifbar zu machen, ist die Verwendung von Suchmaschinen. Mit frei verfügbaren Plattformen wie Elasticsearch² ist das auch verhältnismäßig schnell gemacht. Derartige Software arbeitet schlagwortbasiert, d. h., man muss die Nomenklatur eines Themenbereichs kennen, um zielgerichtet suchen zu können, und sich darauf verlassen, dass die indextierten Dokumente einigermaßen frei von Rechtschreibfehlern und Paraphrasierungen sind. Sowohl in der akademischen Forschung als auch industriell bei den IT-Riesen dominieren aktuell hingegen subsymbolische Ansätze für die Informationsverarbeitung, typischerweise auf Basis von Sprachmodellen (Language Models), die die sogenannte Transformer-Architektur verwenden.

4.2.1 Sprachmodelle

Solche Sprachmodelle, wie die (recht medienwirksam vermarktete) GPT-Reihe (Radford 2018, 2019; Brown 2020) und die verschiedenen BERT-Varianten (Devlin 2018; Clark 2020; Lan 2019), bilden den Ausgangspunkt nahezu aller modernen Deep-Learning-Ansätze für Natural Language Processing (NLP). Sie bilden den Grundstein für

¹Für eine Einführung in die Boxology siehe Kap. 1.

²<https://www.elastic.co>.

verschiedene Anwendungsgebiete. Grundlegend ist ein Sprachmodell so definiert, dass es die Plausibilität einer gegebenen Textsequenz einschätzen soll. Trainiert wird es, indem es eine große Menge natürlichsprachlicher Texte gezeigt bekommt. Wenn ausreichend Trainingsdaten gegeben sind, wird das Modell eine hohe Resilienz gegenüber Paraphrasierungen und Rechtschreibfehlern entwickeln. Einerseits liegt das daran, dass ähnliche Begriffe in ähnlichen Kontexten auftauchen – also sowohl Synonyme als auch falsch geschriebene Varianten von Worten in ähnlichen Sätzen zu finden sind. Andererseits arbeiten moderne LM mit einem speziellen Encoding, bei dem Worte aus ihren Teilen zusammengesetzt werden. Ist einer dieser Teile „kaputt“, so ist ein Großteil der Information nach wie vor aus den anderen Wortteilen reproduzierbar.

Methodisch funktioniert dies, indem Text durch sogenannte Embedding-Vektoren kodiert wird. Deep Learning zeichnet sich dadurch aus, dass die bestimmenden Eigenschaften, die zum Beispiel zum Klassifizieren verwendet werden, eigenständig von den Modellen gelernt werden. Es hat sich gezeigt, dass dies ein sehr erfolgreicher Ansatz ist, wenn dieser auch mit fehlender Transparenz der Entscheidungsfindung eines Modells einhergeht. Die gelernten Embeddings sind Vektoren, genauer gesagt Punkte in hochdimensionalen Räumen, und können dann von weiteren Modellen als Eingabe verwendet werden. Man nennt diese Repräsentationen **Feature-Vektoren**. Modelle, die spezifische Aufgaben zu lösen haben, lernen Korrelationen dieser gelernten Features zu identifizieren und auf ihrer Basis eine Entscheidung zu treffen.

Moderne Sprachmodelle sind zwar deutlich resistenter gegenüber den zuvor angesprochenen Problemen – haben allerdings auch Nachteile: Es sind Machine-Learning-Blackboxen, die einerseits für spezifische Domänen getunt werden müssen und andererseits intransparent funktionieren. Ein anschauliches Beispiel dafür sind moderne Interaktionssysteme wie Chatbots, wenn sie für die Suche verwendet werden. Diese können unter Umständen flexibler als die schlagwortbasierten Systeme Antworten liefern und einen Kontext über mehrere Anfragen hinweg aufrechterhalten. Jedoch verliert der Verwender nahezu komplett die Kontrolle über den Suchfluss, da Chatbots inhärent opak funktionieren. Zudem können sie Fakten herbeifantasierern, die, wie ChatGPT (<https://openai.com/research/gpt-4>) aktuell eindrucksvoll demonstriert, nicht einmal mehr offensichtlich von der Wahrheit unterschieden werden können. Sowieso ist es schwierig, ein System wie ChatGPT im Unternehmenskontext einzusetzen: Es kann gut mit allgemeiner Information (welche, die im Internet frei verfügbar ist), aber schwerlich mit Spezialwissen umgehen. Außerdem stellen sich immer Fragen des Datenschutzes. Zuletzt sind moderne KI-Modelle ausgesprochen ressourcenhungrig, eine Anforderung, die gerade für kleinere Unternehmen oder sehr spezifische Domänen ein effektives Training verhindert.

Trotz aktueller Fortschritte sind dem Einsatz rein neuronaler Ansätze zum Wissensmanagement im Unternehmen harte Grenzen gesetzt: In vielen Szenarien reicht es nicht, „nur“ eine Suchmaschine aufzusetzen oder eine Chatbot-Blackbox zu trainieren, wenn man effektiven Zugang zu der Information seiner Domäne bereitstellen möchte. Es wird ein strukturiertes und kuratiertes Wissensmodell benötigt. Hier ist wichtig, die Kontrolle

über diese Daten zu haben. So liegt also nahe, die beiden Welten miteinander zu verknüpfen: moderne Machine-Learning-Modelle als Werkzeuge einzusetzen, um strukturierte, verknüpfte, von Mensch und Maschine verwendbare Information zu heben.

4.2.2 Wissensgraphen

Beflügelt von der Euphorie um den Entwurf eines „Semantic Web“ in den frühen 2000er-Jahren entstand das moderne Bild von Wissensgraphen. Diese bieten eine strukturierte Sicht auf verknüpfte Informationen. Prominent wurde diese Technologie vor allem durch den Google Knowledge-Graph,³ der zu einer Suchanfrage hilfreiche strukturierte Informationen anbietet. Andere bekannte Vertreter sind Wikidata (<https://www.wikidata.org>), DBPedia (<https://www.dbpedia.org>) und YAGO (<https://yago-knowledge.org>). Sie alle verwenden halbautomatische Prozesse, um Wissen aus Wikipedia (und teilweise anderen Quellen) zu extrahieren und einzupflegen. Die Definition eines Wissensgraphen ist dabei erst einmal recht breit gefasst und beschreibt viele verschiedene Möglichkeiten, Wissen abzubilden. Grundsätzlich ist ein Wissensgraph aber immer ein Datenmodell, in dem *Dinge* (die Entitäten) über *Relationen* miteinander verknüpft werden. Eigenschaften der Entitäten werden also über die Relationen beschrieben, die sie mit anderen Entitäten haben. Man spricht hier dann von **Knoten** (den Entitäten), die über **Kanten** (den Relationen), als Tripel, kodiert werden. Ein Tripel ist ein 3-Tupel aus zwei Entitäten und einem Relationstyp: zum Beispiel (**Germany, Currency, Euro**). Wichtig zu verstehen ist aber, dass es sich dabei um eine abstrakte Sicht auf die Daten handelt – ob sie „klassisch“ in einer relationalen Datenbank oder in einer dafür spezialisierten Graphdatenbank gespeichert wird, ist dabei unerheblich. Vorteile dieser Sicht auf Daten liegen primär in der für Menschen sehr intuitiven Art, Wissen darzustellen, und den Möglichkeiten einer gut nachvollziehbaren Visualisierung. Zeitgleich ist es eine von Maschinen gut verarbeitbare Informationsdarstellung.

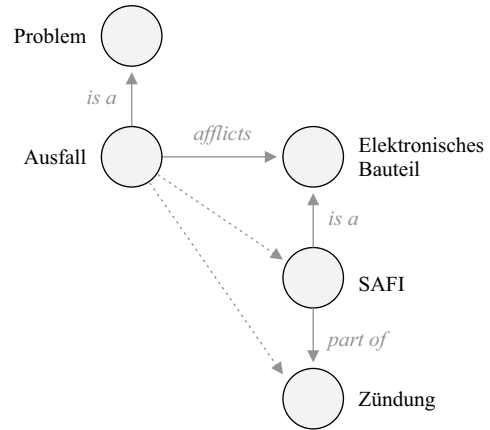
Eine der wichtigsten Eigenschaften von Wissensgraphen ist die Möglichkeit, formale Schlussysteme anzuwenden. In Abb. 4.3 ist ein kleiner Graph abgebildet, in dem man Bauteile und sie betreffende Probleme taxonomisch abgebildet und miteinander verknüpft hat. Eine einfache Schlussfolgerung, die sich in diesem Graph anbietet: Wenn etwas, das ein Problem ist, eine Oberkategorie von Bauteilen betrifft, dann wird dieses Problem auch Kindknoten betreffen können. Diese Art strukturelles Wissen kann nun genutzt werden, um etwa, wenn ein Kunde einen „Ausfall der Zündung“ meldet, vorzuschlagen, sich das „SAFI“ anzuschauen, da es als elektronisches Bauteil dieses Problembild zeigt.

Der klassisch verbreitetste Standard für die Codierung solcher Graphen ist das Resource Description Framework (RDF).⁴ Konzeptuell beschreibt dieses die Graphdarstellung in der oben eingeführten Tripel-Variante. Um RDF-Graphen mithilfe eines Schemas

³<https://cloud.google.com/enterprise-knowledge-graph/docs/search-api>.

⁴<https://www.w3.org/RDF>.

Abb. 4.3 Ein kleiner Bauteilgraph



zu formalisieren, existieren das RDF-Schema (RDFS)⁵ und darauf aufbauend die Web Ontology Language (OWL).⁶ Diese Standards bieten ein spezielles RDF-Vokabular an, welches den Rahmen für die Anwendung fertiger Schlussysteme bietet. So wird es möglich, Verifikation und Inferenz auf den Daten zu betreiben. Der De-facto-Standard für strukturierte Anfragen an RDF-Daten (das Pendant zu SQL für relationale Datenbanken) heißt SPARQL.⁷

Es gibt aber auch andere Ansätze, wie die verschiedenen Ausprägungen von Property Graph Models (z. B. Labeled Property Graphs (LPG)). In diesen Modellen kann man Knoten und Kanten des Graphen beliebig Attribute zuordnen. Dies vereinfacht das Modell insofern, als nicht wie bei RDF für jedes Attribut eine weitere Kante mit Attributknoten hinzugefügt werden muss. Ein bekannter Vertreter eines Datenbanksystems für Property Graphs ist Neo4j.⁸ LPGs werden in der Regel mit der Abfragesprache Cypher angesteuert.

4.2.3 Taxonomien

Gegenstand dieses Kapitels ist ein spezieller Typ Graph, die Taxonomie. In dieser wird über die Relationen zusätzlich eine Hierarchie abgebildet. Ein prominentes Beispiel einer Taxonomie ist WordNet,⁹ eine lexikalische Datenbank mit dem Ziel, Beziehungen

⁵<https://www.w3.org/TR/rdf-schema>.

⁶<https://www.w3.org/OWL>.

⁷<https://www.w3.org/TR/sparql12-query>.

⁸<https://neo4j.com>.

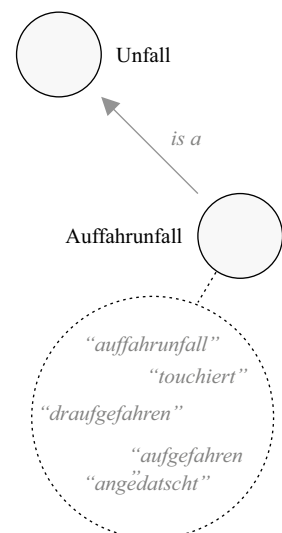
⁹<https://wordnet.princeton.edu>.

zwischen Wörtern abzubilden. Wörter werden hier zu Gruppen mit gleicher Bedeutung (den Synsets) zusammengefasst und in Beziehung, wie z. B. Hyperonym (Unterkategorie: Kuchen ist eine Backware) und Meronym (Teilganzes: Mehl ist Teil eines Kuchens), gesetzt. Die Hierarchie der Taxonomie bildet dabei den Detailgrad ab: Dinge, die in der Taxonomie „höher“ verortet sind, sind allgemeiner und abstrakter.

Im Gegensatz zum allgemeinen Ansatz von WordNet soll es in diesem Kapitel darum gehen, domänenspezifische Taxonomien aufzubauen, welche relevantes Wissen im täglichen Betrieb eines Unternehmens abbilden. Bei einem Maschinenbauer sind dies die Maschinen/Teile und Probleme/Lösungen, die diese betreffen. Bei einem Versicherer sind dies beteiligte Personen, Schäden, Schadensumstände etc. Diese Taxonomien müssen entweder komplett neu entworfen werden oder entstehen durch die Anpassung einer allgemeineren, übergreifenden Taxonomie an die spezifische Umgebung. Im obigen Beispiel wäre dies das Hinzufügen weiterer Bauteile für Zündungen oder andere relevante Problemklassen. In unserer Definition einer Taxonomie ist jedem Knoten eine Menge von Phrasen zugeordnet, welche das Konzept textuell beschreiben (Abb. 4.4).

Eine solche Taxonomie aufzubauen, bietet einige Vorteile: Das Wissensmodell, als ein Teil eines übergeordneten Wissensmanagements, dient als aktive Dokumentation der Domäne. Die Verwender der Taxonomie und die Domänenexperten können sie als Diskussionsgrundlage und geteiltes Glossar verwenden. Ein Graph, wenn er gut visualisiert wird, stellt eine verhältnismäßig niedrige technische Hürde dar und kann intuitiv von Menschen verstanden werden. So wird der Zugang zum Wissen für alle Beteiligten vereinfacht. Aber auch Maschinen haben nun eine deutlich bessere Möglichkeit, strukturiert auf das Wissen zuzugreifen, z. B. um sie zum Schlussfolgern im Rahmen von Such- oder Empfehlungsvorgängen zu nutzen.

Abb. 4.4 Einer Schadensfalltaxonomie werden Phrasen zugeordnet, die den Sachverhalt beschreiben



Die Kombination von einem übergeordneten Metamodell wie einer solchen Taxonomie und den konkreten Ausprägungen (Instanzen), mit denen die Entitäten der Taxonomie beobachtet werden können, bildet eine Wissensbasis. Man spricht bei der Taxonomie dann auch von der **T-Box** (Terminology). Ihr können weitere Regeln und Einschränkungen hinzugefügt sein: zum Beispiel, dass Bauteile nur in andere Bauteile oder Maschinen eingebaut sein können. Die **A-Box** (Assertions) hingegen beschreibt die konkreten Instanzen, also die beobachtbaren Ausprägungen der Taxonomieentitäten.

4.2.4 Wissenserfassung

Das Erstellen einer solchen Wissensbasis ist äußerst kostspielig. Es kostet viel Zeit, sich umfassend Gedanken darüber zu machen, welche Anforderungen der Wissensgraph bedienen soll und welche Informationen dementsprechend abgebildet werden sollen. Zudem muss er immer wieder angepasst werden, wenn sich Anforderungen verändern oder die Realität vom Modell divergiert (z. B. aufgrund der Einführung neuer Bauteile). Sowie so ist es schwierig, alle relevanten Konzepte einer Domäne a priori ganzheitlich zu erfassen.

Ganz allgemein sind die zwei wichtigsten Kriterien für die Aufnahme von Dingen in die Taxonomie wohl *Frequenz* und *Potenzial*. *Frequenz* ist als quantitative Eigenschaft trivial: Wenn etwas häufig im täglichen Geschäft in Erscheinung tritt, wird es wohl von Relevanz sein. Wenngleich es sich um eine unwichtige Information handelt, so kann das Wissen darüber allein schon hilfreich sein, um, zum Beispiel in einem Suchszenario, einen Filter zu implementieren, der den Zugriff auf wertvolle, aber seltenere Informationen erleichtert. *Potenzial* (im Sinne der Relevanz für den konkreten Einsatz im täglichen Geschäft) als qualitative Eigenschaft ist schwieriger einzuschätzen und hängt von dem konkreten Use Case ab. Vielleicht ist es wichtig, eine ganze Klasse von Entitäten zu identifizieren, um statistische Kennzahlen aus vielen Beobachtungen abzuleiten, vielleicht muss die sprichwörtliche Nadel im Heuhaufen gefunden werden.

Nun muss jedoch entschieden werden, wie die Vorgehensweise für den Aufbau der Taxonomie aussehen soll. Grundsätzlich bietet sich einerseits die Möglichkeit, den Wissensgraphen händisch durch Experten zu erstellen oder ihn automatisiert und datengetrieben durch Modelle inferieren zu lassen. Bei der händischen Erfassung definiert ein Domänenexperte Fakten und Entitäten für den Wissensgraphen. Die Grundlage bildet hierfür sein Erfahrungsschatz. Da Domänenexperten aber in den seltensten Fällen in der Lage sind, derartige Graphen technisch aufzubauen, arbeiten sie im Tandem mit einem Wissensmodellierer. Beide nehmen im beidseitigen Austausch strukturiert Wissen auf und verknüpfen es. Ein großer Vorteil dieses Ansatzes ist die hohe Qualität des gewonnenen Wissens. Nachteile bestehen darin, dass der obige Prozess aufwendig, zeitintensiv und damit teuer ist. Zudem besteht die Gefahr, dass das Modell aus dem sprichwörtlichen Elfenbeinturm definiert wird: Dinge, die hypothetisch wichtig, aber in der Praxis kaum relevant sind, bekommen vielleicht sehr viel mehr Aufmerksamkeit, als ihnen gebühren sollte.

Auf der anderen Seite steht das vollautomatische Erstellen des Graphen mithilfe von KI-Modellen. Die Grundlage hierfür bilden unstrukturierte Daten (häufig in Textform), die im Rahmen von Geschäftsprozessen ohnehin erhoben werden, wie z. B. Tickets, Handbücher, Ablaufbeschreibungen, FAQs, Mails, Foren und Chats. Diese Daten sind häufig in großen Mengen vorhanden. Die offensichtlichen Vorteile des Ansatzes bestehen in der Zeit- und Kosteneffizienz, der Skalierbarkeit, der Flexibilität und Aktualität und in der Tatsache, dass Informationen, die im Graphen landen, unmittelbar aus einer echten Beobachtung des Faktus in den Daten abgeleitet sind. Nachteile sind, dass erst einmal genügend (in der Regel annotierte) Daten für ein Modelltraining vorhanden sein müssen. Außerdem sind Echtweltdaten häufig sehr verrauscht. Obwohl aktuelle Deep-Learning-Verfahren oft eine überraschende Resilienz demgegenüber haben, so sind die inferierten oder extrahierten Informationen inhärent fehlerbehaftet. Im praktischen Einsatz liefern auch moderne neuronale Extraktionsmodelle noch einiges an Datenmüll, der entweder toleriert oder von Menschen nachträglich gesäubert werden muss.

Man kann festhalten, dass der Aufbau einer Taxonomie mit Experten hauptsächlich Qualität auf Kosten von Quantität liefert, wohingegen ein modellgetriebener Aufbau Quantität bringt, aber oft mit mangelnder Qualität einhergeht. Es liegt also nahe, dass eine Verknüpfung dieser beiden Ansätze gewinnbringend sein kann. Experten verwenden Text-Mining-Modelle, um schneller, zielgerichteter und datengetrieben Wissensgraphen aufzubauen. Dadurch, dass die Erweiterung der Taxonomie durch den domänenspezifischen Text getrieben wird, ist die abgebildete Information authentisch und die Taxonomie ist in den operativen (textuellen) Daten *verankert*. Ein solcher interaktiver, KI-gestützter Wissensgraphaufbau ist Kerngegenstand dieses Kapitels.

4.3 Interaktiver Aufbau von Wissensgraphen

Der diskutierte Prozess einer interaktiven, KI-gestützten Wissensgewinnung wird in Abb. 4.5 illustriert: Ziel ist es, auf Grundlage eines Textkorpus (grün) eine Taxonomie zu erstellen (violett). Hierzu findet (im Zentrum der Abbildung) ein neuronales Modell Anwendung, welches das Textkorpus nach neuen Entitäten und Fakten durchsucht. Diese werden in einem Tool dem Domänenexperten präsentiert, welcher die vorgeschlagenen Fakten kuratiert. Dieses Kapitel konzentriert sich auf diesen Erfassungsprozess. Die Rahmenprozesse dieser Aufnahme bilden (am Rand der Abbildung) die initiale Definition einer kleinen Basis-taxonomie, die Anbindung der Datenquellen aus betrieblichen Informationssystemen oder die nachgelagerte Anwendung der Taxonomie für Such- oder Empfehlungsszenarien.

Wie bereits angemerkt, soll datengetrieben gearbeitet werden: Das heißt, es müssen Daten vorliegen, die im betrieblichen Alltag anfallen und maschinell verarbeitet werden können. In unserem Fall ist das Freitext. Dieser Text kann alles Mögliche sein: Tickets aus Supportsystemen, Transkriptionen von Anrufen, Meeting-Protokolle, E-Mail-Korrespondenzen etc. Wichtig ist nur, dass es genug davon gibt, sodass sich der Einsatz einer komplexeren Toolchain überhaupt lohnt.

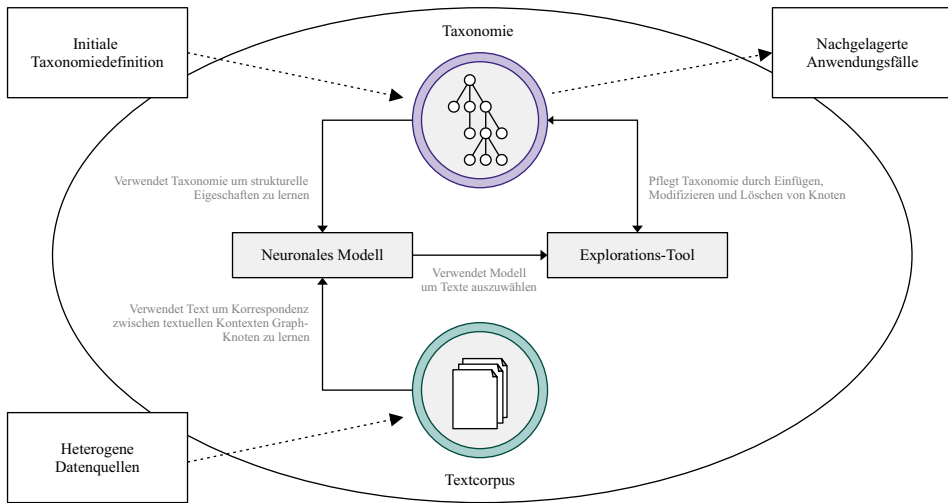


Abb. 4.5 KI-gestützte Wissensgewinnung

4.3.1 Erstellung der initialen Taxonomie

Um den Prozess starten zu können, muss eine initiale Taxonomie erstellt werden. Dies ist der Ausgangspunkt, von dem aus der Prozess angestoßen wird, der sukzessive den domänenspezifischen Wissensgraphen liefert. Zu Beginn muss geklärt werden, welche Anforderungen an die Taxonomie kurz- und mittelfristig gestellt werden. Das ist ein wichtiger Schritt mit dem Ziel, das Modell klein und scharf zu halten. Bei dem Aufbau von Graphen (wie dieser Taxonomie) wird gerne empfohlen, einen allgemeinen, aber qualitativ hochwertigen Graphen wie Wikidata als Grundlage zu nehmen und von diesen zu selektieren. Dies ist bei sehr engen Domänen und hoher Fachspezifität aber häufig kein besonders hilfreicher Ansatz, da (primär für die deutsche Sprache) solche Daten gar nicht oder nur selten vorhanden sind. Ist man sich nun im Klaren über das Ziel, so kann man relevante Entitäten der Domäne eingrenzen: Gibt es z. B. in Anwendungsfällen des technischen Services verschiedene Maschinentypen? Sind in einem juristischen Use Case Personen involviert, die eine juristische Relevanz haben? Wenn im Kontext einer Versicherung Schäden auftreten, welche fallen einem ad hoc ein? Hier sollte pragmatisch gearbeitet werden – die Taxonomie wird im Laufe der Zeit ohnehin wachsen. Am besten orientiert man sich an einem Szenario und fokussiert sich auf dessen relevante Aspekte. Zusätzlich empfiehlt es sich, stichprobenartig anhand von Rohdaten den Nutzen des Wissensgraphen zu beurteilen. So kann geprüft werden, wie viel Potenzial tatsächlich vorhanden ist und wie viele nutzbare Entitäten man realistisch extrahieren können wird. Hierzu baut man einen Suchindex, um die Textdaten (oder einen Teil von ihnen) durchsuchbar zu machen. Etablierte Anwendungen wie Elasticsearch (<https://www.elastic.co>) bieten sich an. Sucht man nun nach Begriffen seiner initialen Taxonomie, geben die zurückgelieferten Texte Aufschluss über Menge

und Güte der textuellen Kontexte, die einer Entität zugeordnet werden können. Wie häufig kommt der Begriff vor? Gibt es ausreichend Suchergebnisse? Ist der Begriff spezifisch genug gewählt? Ist eine Disambiguierung wichtig?

Wenn man sich auf ein paar grundlegende Begrifflichkeiten geeinigt hat, sollte man sie miteinander verknüpfen. Synonyme lassen sich zu Gruppen zusammenfassen. Artverwandte Begriffe lassen sich unter Oberbegriffen subsumieren. Wenn man „bottom-up“ arbeitet und immer abstraktere Oberbegriffe sucht, fügt sich ganz natürlich die initiale Taxonomie zusammen.

4.3.2 Erweiterung der Taxonomie

Nachdem die initiale Taxonomie erstellt wurde, kann die eigentliche Arbeit beginnen. Der Prozess der Verfeinerung ist in Abb. 4.6 dargestellt. Er basiert auf einer interaktiven Exploration der Textsammlung mit dem Ziel, die initiale Taxonomie substanziell zu erweitern: Hat man sich z. B. erste relevante Maschinen, Symptome oder Schadensklassen sowie Beziehungen zwischen ihnen definiert, ist das Ziel, nun möglichst effizient weitere solcher Entitäten/Beziehungen zu entdecken und dem Wissensgraphen hinzuzufügen.

Die Grundlage hierfür bildet das Textkorpus: Wir nehmen an, dass neue Entitäten/Beziehungen in ähnlichen textuellen Kontexten wie die bereits bekannten auftreten werden. Suchen wir also nach ähnlichen Textstellen wie jenen, die bekannte Entitäten enthalten, stoßen wir auf vermeintlich neue interessante Instanzen und können diese dem Wissensgraphen hinzufügen. Wir formulieren also Wissensgrapherweiterung als ein Suchproblem mit folgenden Schritten:

1. Vorkommen der bekannten Entitäten werden per Schlagwortsuche in der Textsammlung gefunden. Alle gefundenen Texte werden den jeweiligen Knoten zugeordnet und dienen als erste Verankerung des Wissensmodells in den Daten.

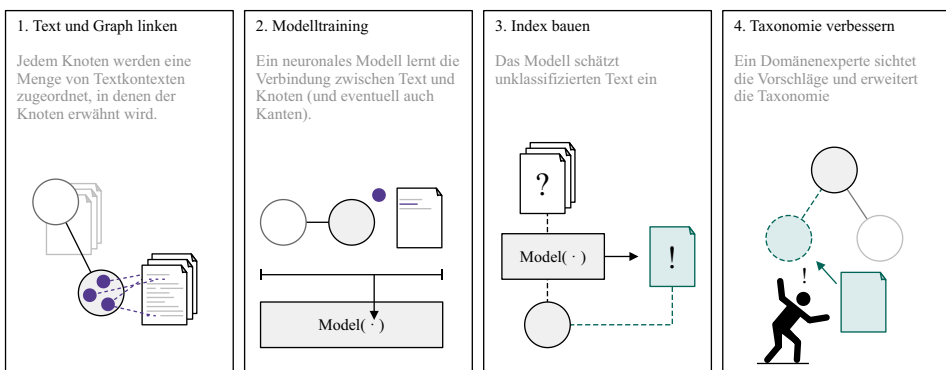


Abb. 4.6 Prozess zur Gewinnung neuer Taxonomieknoten

2. Diese Kombination aus Graph und Text dient als Trainingsmenge für Mining-Modelle. Die resultierenden Modelle sollen Textkontexte finden, in denen sich potenziell Entitäten oder Fakten von Interesse verbergen. Um dies zu realisieren, sind verschiedene Techniken aus dem Bereich des NLP denkbar, auf die im Abschnitt „Neuronaler Werkzeugkasten“ eingegangen wird: Named Entity Recognition (NER), Relation Extraction (RE) oder semiinduktive Knowledge Graph Completion (OW-KGC).
3. Das oder die Mining-Modelle werden auf Text angewendet, in dem bisher keine der bekannten Entitäten/Fakten gefunden wurden. Da die Modelle vorschlagen, welcher Knoten am ehesten zum Text „passt“, werden die Vorschläge pro Knoten gesammelt und nach Modellkonfidenz sortiert.
4. Die Ergebnisse eines Minings auf dem Textkorpus werden per grafisches Interface den Domänenexperten und Wissensmodellierern aufbereitet zur Verfügung gestellt. Die Experten sichten die vorgeschlagenen Texte und Fakten und entscheiden, welche Vorschläge wo der Taxonomie hinzugefügt werden sollen. Während dieser Arbeit wächst das Verständnis der Domäne durch die Konfrontation mit den vielen dokumentierten Ereignissen der Vergangenheit. Dieses wachsende Verständnis wird unmittelbar und aktiv durch die Verbesserung der Taxonomie dokumentiert.

Mit dem verbesserten Wissensmodell kann nun zu Punkt (1) zurückgekehrt und der Prozess iteriert werden. So wachsen mit allen folgenden Iterationen die Taxonomie, das Verständnis über die Eigenschaften der abgebildeten Domäne, die Nützlichkeit der Taxonomie für nachgeordnete Anwendungsfälle, die Trainingsmenge und die Güte der trainierten Mining-Modelle.

4.4 User Interface

Ob der oben geschilderte Explorations- und Adaptionprozess erfolgreich ist, hängt maßgeblich davon ab, ob die verfügbaren Texte effektiv zugreifbar sind. Ein Domänenexperte muss schnell und begründet die Entscheidung treffen können, ob eine Information es wert ist, dem Graphen hinzugefügt zu werden. Dazu wird eine Oberfläche benötigt, die den folgenden Anforderungen genügt:

- **Zielgerichtet suchen:** Vor allem, wenn die Taxonomie größer wird, interessieren den Domänenexperten nur noch einzelne Aspekte, die verbessert werden müssen. Zum Beispiel kann sich herausstellen, dass in einer Fachanwendung für Schadensklassifizierung die Granularität der Schadenstypen nicht fein genug ist und der Experte die Taxonomie an dieser Stelle verfeinern muss. Die Oberfläche sollte dann nur Inhalte anzeigen, die für die aktuell interessante Taxonomieregion von Belang sind.
- **Effektiv suchen:** Ergebnislisten für einen Taxonomiekontext werden, wenn die zugrunde liegenden Daten qualitativ mangelhaft sind, ebenfalls ein hohes Rauschen

aufweisen. Kein Suchsystem ist perfekt und jedes Modell ist maximal so gut wie die Daten, auf denen es arbeitet. Daher ist es wichtig, dass Ergebnislisten effektiv durchstöbert und gefiltert werden können.

- **Adaption ermöglichen:** Jede gefundene hilfreiche Information kann der Taxonomie hinzugefügt werden. Dies allein ist jedoch nicht ausreichend. Wenn der Vorschlag hilfreich, aber für den konkreten Kontext nicht relevant ist, dann sollte er ggf. an einer anderen Stelle der Taxonomie hinzugefügt oder zumindest für eine spätere Abarbeitung zurückgelegt werden können. Ferner können die neu gewonnenen Informationen auch dazu führen, dass ganze Teile der Taxonomie überdacht und neu geordnet werden müssen.

Die Abb. 4.7 und 4.8 zeigen ein prototypisches User Interface, welches diese Anforderungen bedient: Auf der linken Seite ist die Taxonomie (T-Box) abgebildet. Knoten können aufgeklappt werden, um ihre jeweiligen Hyperonyme zu zeigen. Für ausgewählte Knoten werden dann zentral die wichtigsten Informationen zusammengefasst: In wie vielen Texten sind Erwähnungen des Knotens auffindbar, welche Synonyme (also die A-Box) sind ihm zugeordnet und wie viele Vorschläge macht das Mining-Modell? Des Weiteren

Upload Taxonomy

Taxonomy Maintenance

Taxonomy

- teil ist beschaedigt
- konfigurationsfehler
- auslaufen
- wert zu hoch / niedrig / schwankt
- ausserhalb der toleranz**
- wert schwankt
- drehzahl schwankt
- temperaturschwankungen
- wert zu hoch
- wert zu niedrig
- zu dunkel
- zu fest
- zu frueh
- zu hart
- zu heiss
- zu hell
- zu kalt
- zu kuehl
- zu leicht
- zu schwer
- zu spaet
- zu tief
- zu warm
- zu weich
- softwarefehler
- bedienfehler
- diagnose und behebung
- geraet zeigt fehler

[Add Node](#)

Node: ausserhalb der toleranz (id=144)

[Show Predictions](#)

Synonyms	Count
ausserhalb der toleranz	12383
Children	23422

Phrases (2)

- ausserhalb der toleranz (92 matches)
- nicht im toleranzbereich (7 matches)

[Add](#)

Associated Texts

Direct	Count
Direct	99
Children	46
Total	145

[Add Child Node](#) [Delete Node](#)

Already Matched Texts

ausserhalb der toleranz

Ausserdem sind einig der Kolben **ausserhalb der Toleranz** (grosser):

BEI ist **ausserhalb der Toleranz**.

Die Abweichung zwischen der KL5151 Zaehlerkarte und dem Umrichter waren **ausserhalb der Toleranz**

21. 06. 2011 20:16 „ Fernserviceaktivitaet Gemessenes Mass war **ausserhalb der Toleranz**.

Die Anlage wurde am 31. 03. 2003 mit Laser ausgerichtet , wobei keine Abweichungen **ausserhalb der Toleranz** festgestellt wurden .

Problem mit kardanischen Fuefachkopf Mittensersatz ist **ausserhalb der Toleranz** Umkehrspiel der A Achse ist ebenfalls ausserhalb der Toleranz

[load more](#)

nicht im toleranzbereich

Spur A und Spur B liegen **nicht mehr im Toleranzbereich**.

Die Übergänge der Leisten (Ende Bandstahl) Pos. 010 und 015) sind **nicht im Toleranzbereich**

Die Klebstoffverteilung ueber die gesamte Folienbreite ist **nicht im Toleranzbereich**.

Meiner Meinung muss jemand von uns vor Ort und die Platte pruefen , kann mit nur vorstellen das die Bohrungen **nicht im Toleranzbereich** sind.

Positionen der Bohrungen sind **nicht im Toleranzbereich** Hr. Rochner wuenscht im Zeitraum 28. 30:09 einen Techniker zur Maschineneinstellung bitte mit Hr. Rochner terminieren

Konnte vor Ort **nicht** nachvollzogen werden Masse waren **im Toleranzbereich**

[load more](#)

Abb. 4.7 Übersichtsseite und Auswahl des Einstiegspunktes

Upload Taxonomy

Predictions for "ausserhalb der toleranz"

Related Symptoms	Synonym Predictions	Child Predictions
<ul style="list-style-type: none"> + ausserhalb der toleranz <p>Other Symptoms</p> <ul style="list-style-type: none"> + teil ist beschaedigt + konfigurationsfehler + auslaufen + wert zu hoch / niedrig / schwankt + softwarefehler + bedienfehler + diagnose und behebung + geraet zeigt fehler 	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>x Score: 1.00 (-2.60) Prediction: 985763</p> <p>Wenn die C Achse 2/10 mm ausserhalb der Position ist , versucht sie weiterhin in Position zu kommen .</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>x Score: 1.00 (-2.32) Prediction: 497517</p> <p>Der Sollwert (0) fuer die neuen X Achsen wird als fehlerhaft im der Matext angezeigt , obwohl der Wert innerhalb der Toleranz ist .</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>x Score: 0.99 (-1.56) Prediction: 413932</p> <p>3 Zehntel unwuechtig aushalb der Toleranz</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <p>x Score: 0.99 (-5.02) Prediction: 615239</p> <p>Fehler bei Resterueckfuehrung ; Rest liegt ausserhalb der Position der Saugtraverse</p> </div>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>x Score: 0.87 (-7.19) Prediction: 44153</p> <p>Formelfehler im Makro IF ZV < = 16.3 THEN 16.3 ELSE ZV hier wird immer 16,3 gefraest</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>x Score: 0.79 (-6.06) Prediction: 1288471</p> <p>Fehlersuche ueber Simens Starter und SPS , Temperaturfehler am Sensor aufgefunden .</p> <p>temperaturfehler am sensor Filter</p> <p>child of Annotate</p> <p>ausserhalb der toleranz</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <p>x Score: 0.54 (-5.63) Prediction: 559200</p> <p>Vermutlich steigungsfehler von Spindel</p> </div>

Abb. 4.8 Textvorschläge zur Annotation

Abb. 4.9 Annotationsdialog im Detail

x Score: 0.99 (-12.41) | Prediction: 2907509

Gewinde von Gasdruckfeder ist abgebrochen

gewinde abgebrochen Filter

synonym of Annotate

gewinde beschaedigt

können neue Synonyme und Kindknoten hinzugefügt oder der Knoten sowie seine Kinder gelöscht werden. Auf der rechten Seite werden Textstellen angezeigt, in denen Erwähnungen des Knotens gefunden wurden. Klickt man zentral auf „Show Predictions“, wird man auf die Seite weitergeleitet, auf der die Modellvorschläge gezeigt werden.

Hier wird auf der linken Seite die lokale Nachbarschaft des ausgewählten Knotens angezeigt. Des Weiteren ist auch ein Zugriff auf den Rest der Taxonomie möglich. Auf der rechten Seite sind die Vorschlagslisten mit ausgewählten Textstücken zu sehen. In den einzelnen Boxen mit Textstücken können nun per Klick auf die einzelnen Worte Phrasen vorausgewählt und nach Belieben im Freitextfeld angepasst werden (Abb. 4.9). Der Platz für den neuen Knoten in der Taxonomie ist per Modellvorhersage vorausgewählt und kann, falls gewünscht, angepasst werden. Um Redundanz zu vermeiden, werden bereits neu hinzugefügte Phrasen aus den Vorschlägen entfernt. Ebenso kann der Nutzer Vorschläge des Modells, die unpassend sind, aus der Liste entfernen. Wenn unnütze Vorschläge redundant auftauchen, kann man per „Filter“ die Listen um alle Vorschläge verringern, die die gefilterte Phrase beinhalten.

4.5 Neuronaler Werkzeugkasten

Kernstück der Exploration nach neuen Entitäten und Fakten ist – wie oben geschildert – ein Mining-Modell, welches Passagen mit neuen, potenziell interessanten Entitäten und Fakten entdeckt. Hierzu bieten sich verschiedene Modelltypen aus NLP und ML an. Die Hauptanforderung ist Robustheit, denn Daten in Geschäftskontexten enthalten häufig orthografische Fehler, viele domänenspezifische Begriffe und Abkürzungen, Redundanz und Datenmüll. Viele in der Industrie verbreitete, regelbasierte Ansätze tun sich damit schwer. Ein wörterbuchbasierter Lemmatizer kann nicht lemmatisieren, wenn das Wort falsch geschrieben ist. Auch Suchmaschinen mit invertierten Indexen haben damit größere Schwierigkeiten. Wenn dazu noch die Interpunktion nicht stimmt, dann wird ein POS-Tagger wenig Chancen haben, korrekt zu taggen. Und natürlich stellt Paraphrasierung („der Schlauch hat ein Leck“ versus „es tritt Öl aus“) eine Herausforderung dar.

In den vergangenen Jahren konnte der NLP-Bereich mithilfe von Deep-Learning-Verfahren große Fortschritte erzielen. Modelle, vorrangig jene, die die zu Beginn angesprochenen aktuellen Transformer-Architekturen implementieren, zeichnen sich insbesondere durch Robustheit gegen die oben genannten Probleme aus. Es lohnt sich also, einen Blick darauf zu werfen, wo sich der Stand der aktuellen Forschung befindet. Die hier vorgestellten Methoden korrespondieren mit den zuvor skizzierten Anforderungen an den Aufbau von Wissensgraphen: Wissensextraktion aus Text mithilfe von Text-Mining-Ansätzen und Wissensverfeinerung mithilfe von Knowledge Graph Completion (KGC).

4.5.1 NLP-Tasks

Um die beschriebene Pipeline umzusetzen und vom Rohtext bis zum Fakt im Wissensgraphen zu gelangen, gibt es eine Reihe verschiedener NLP-Techniken. Diese beschäftigen sich unter anderem damit, Erwähnungen von Entitäten in Textstellen zu lokalisieren, zu identifizieren, welche Entität mit einer Erwähnung gemeint ist, und dem Graphen Fakten hinzuzufügen. Eine Auswahl dieser Techniken wird im Folgenden vorgestellt. Abb. 4.10 zeigt einen Überblick, wo und wann diese Techniken zum Einsatz kommen können.

4.5.2 Named Entity Recognition (NER)

Eine wichtige Komponente bei der Wissensakquise mithilfe von Text-Mining ist Named Entity Recognition (NER – einen Überblick gibt es in Li (2020)). Die Aufgabe eines NER-Modells ist, in einem Stück Text zu entscheiden, ob dort Entitäten von Interesse beschrieben werden. Ein Beispiel:

„Safi Ausfall Trigger Pickup: Nach diverser Fehlersuche und Austausch fast aller Teile der Zündung weitergeleitet an XYZ-Betrieb.“

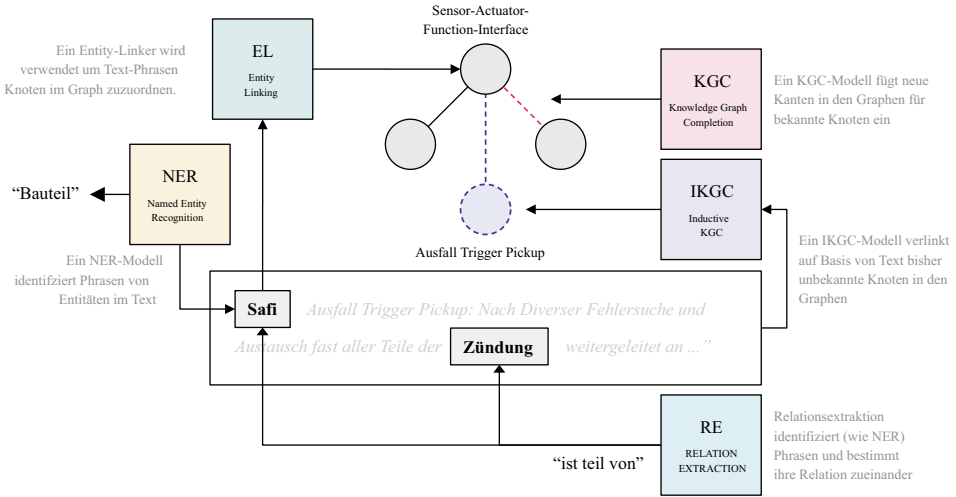


Abb. 4.10 Übersicht verschiedener neuronaler Techniken zur Informationsextraktion und Graphvervollständigung

Tab. 4.1 Mögliche Vorschläge eines domänenspezifischen NER-Modells

Markierter Bereich	Klassen
„Safi“	Firma, elektrisches Bauteil
„Ausfall“	Maschinelles Problem
„Trigger Pickup“	Kommunikationsfunktion
„Zündung“	Bauteil
„XYZ-Betrieb“	Firma

Hier hat ein Techniker dokumentiert, wie bei einem Problem mit der Zündsteuerung einer Maschine zuerst, um das Problem anzugehen, Teile der Zündung getauscht wurden und bei ausbleibendem Erfolg das Problem eskaliert wird. Ein NER-System markiert im Text die Bereiche, bei denen es Entitäten vom Typ einer vorher definierten Typmenge vermutet. Ein allgemeiner NER-Tagger arbeitet mit generischen Typen wie „Person“, „Firma“ etc. Ein domänenspezifischer Tagger hingegen arbeitet mit Tags, die an der initialen Taxonomie orientiert sind – für dieses Beispiel könnte ein gut funktionierender Tagger die in Tab. 4.1 aufgeführten Entitäten klassifizieren.

4.5.3 Entity Linking (EL)

Hat man durch seine NER-Pipeline Mengen von Erwähnungen gesammelt, muss man nun entscheiden, wie und wo sie in die Taxonomie einsortiert werden müssen. Hier kann man schon entscheiden, ob die Taxonomie erweitert werden sollte (vielleicht hat

man eine neue Kategorie von Bauteiltypen gefunden) oder ob eventuell nur neue Paraphrasierungen einer bekannten Entität entdeckt wurden.

Diese Verbindung automatisiert herzustellen, ist der Anwendungsbereich von Entity Linking (EL – einen Überblick gibt Sevgili (2022)). Hier sollen Erwähnungen von Dingen bereits existierender Entitäten in einem Graphen zugeordnet werden. Das EL-Modell kennt den Graphen (also auch die Verknüpfung der Entitäten untereinander) und weiß auch, welcher Bereich eines Strings die Erwähnung enthält, z. B. hat ein NER-Modell diese Information geliefert. Um das vorherige Beispiel aufzunehmen:

„Safi Ausfall Trigger Pickup: Nach diverser Fehlersuche und Austausch fast aller Teile der Zündung weitergeleitet an XYZ-Betrieb.“

Der Entity Linker hat die Aufgabe, für die Erwähnung „Safi“ die korrekte Entität im Graphen zu bestimmen – soweit diese existiert. Hierbei darf der Linker sowohl den textuellen Kontext der Erwähnung als auch die Vernetzungseigenschaften des Graphen verwenden. Das ist etwa hilfreich, um zu entscheiden, ob es sich in dem genannten Kontext um die Firma **SAFI** oder das **Sensor Actuator Functional Interface** (das konkret gemeinte Bauteil) handelt.

4.5.4 Knowledge Graph Completion (KGC)

Da Entitäten und Fakten teuer zu erfassen sind, werden Wissensgraphen in der Praxis nur unvollständig erfasst. Dies bedeutet einerseits, dass Entitäten nicht vorhanden sein können und (auch mit der Unterstützung von NER- und EL-Systemen) eingefügt werden müssen. Andererseits betrifft dies aber auch die Kanten, die die Entitäten miteinander verbinden. Hier setzen Knowledge-Graph-Completion-Modelle (KGC) an. Diese arbeiten, unabhängig von zugeordnetem Text, auf den strukturellen Graphdaten, wie weiter oben beschrieben, in Tripel-Darstellung. Ein Modell bestimmt die Plausibilität gezeigter Tripel in Form eines Scores. Zum Beispiel würde ein gutes KGC-Modell dem Tripel (SAFI, Teil_von, Zündung) einen hohen und (SAFI, Teil_von, Ausfall) einen niedrigen Score zuweisen. Dies kann verwendet werden, um fehlende Kanten im Graphen einzuziehen, indem man dem Modell einfach für ein Entitäts-Relations-Tupel (z. B. SAFI, Teil_von) alle möglichen Kombinationen mit anderen Entitäten des Graphen zeigt. Das Modell errechnet dann einen Score für jedes potenzielle Tripel. Die Eingabe besteht, wie auch schon beim Sprachmodell, aus gelernten Embeddings, also Feature-Vektoren – diesmal für Entitäten und Relationstypen. Wenn man die Tripel nach Score sortiert, kann man die obersten Kandidaten inspizieren, um zu entscheiden, ob die Relation eingefügt werden soll:

1. (SAFI, Teil_von, Zündung),
2. (SAFI, Teil_von, SAFI),

3. (SAFI, Teil_von, XYZ_Betrieb),
4. (SAFI, Teil_von, Problem),
5. (SAFI, Teil_von, Ausfall),
6. ...

Eine interessante Einordnung aktueller Triple-Scorer-KGC-Modelle findet sich in Ruffinelli (2020).

4.5.5 Semiinduktives KGC

Knowledge Graph Completion, wie im letzten Abschnitt geschildert, kann erst effektiv eingesetzt werden, wenn ausreichend strukturelle Daten vorhanden sind. Dieses Verfahren funktioniert dann besonders gut, wenn der Graph einigermaßen dicht besetzt ist. Bei kleinen, unvollständigen Taxonomien zu Beginn unseres Explorationsprozesses hilft KGC deshalb nicht besonders: Es sind nicht fehlende Kanten zwischen den Entitäten das Problem, sondern unverlinkte Erwähnungen von bislang vollkommen unbekanntem Entitäten im Text. Mit dem Verlinken von (bisher, zur Trainingszeit) unbekanntem Entitäten setzt sich die noch sehr junge Disziplin der *semiinduktiven* KGC auseinander.

Hier stellt sich ein offensichtliches Problem: Anhand welcher Merkmale kann ein Modell entscheiden, wo der beste Platz für die unbekannte Entität ist, wenn sie ansonsten bisher nicht im Graphen verlinkt wurde? Wir haben dafür den Text: Das NER-Modell liefert uns die Information, wo sich in einer Passage die Erwähnung der fraglichen neuen Entität befindet. Die Kombination aus NER und EL lässt uns gruppieren, welche Textkontexte wahrscheinlich den gleichen Entitätstyp beschreiben. Der Kniff ist nun, die oben beschriebenen Repräsentationen des Sprachmodells mit den Repräsentationen, die bei der KGC trainiert werden, zu kombinieren. Die Anfrage an das Modell verändert sich dadurch: Statt (SAFI, Teil_von, Zündung)? ist es jetzt („Text der SAFI beschreibt“, Teil_von, Zündung)? Trainiert man Repräsentationen auf dieser Art von Anfrage, entsteht ein *Alignment* zwischen textueller und struktureller Repräsentation. Das ermöglicht sogar Aussagen über bisher komplett unbeobachtete Entitäten zu machen – sofern das Sprachmodell gut generalisiert.

4.5.6 Relation Extraction (RE)

Eine weitere Disziplin von Informationsextraktion ist die Relationsextraktion. Ziel ist es, Entitäten im Text zu identifizieren und miteinander zu verknüpfen. In der Regel passiert dies durch die Verknüpfung von NER, welche Phrasen benannter Entitäten identifiziert, und durch Relationsklassifikation, welche für ein Paar dieser Entitäten entscheidet, ob und, wenn ja, welche Relation zwischen den beiden besteht. Im obigen Beispiel sollte ein gut arbeitendes Modell in der Lage sein, die Verbindung von der Phrase „SAFI“ als

Bauteilentität mit der „Zündung“ in Verbindung zu bringen. Der aktuelle Forschungsstand von RE wird in Bassignana (2022) beschrieben. Vorteil der Relationsklassifikation ist, dass Tripel direkt in den Graphen eingeführt werden können. Jedoch setzt dies auch voraus, dass Entitäten von Interesse auch in (relativ kurzen) Textabschnitten gemeinsam erwähnt werden.

Gerüstet mit den neuronalen Werkzeugen und der Bedienoberfläche soll der Ansatz mit Industriedaten erprobt werden. Hierzu werden im Folgenden zwei anwendungsnahe Use Cases präsentiert, die in kooperativen Forschungsprojekten mit Industriepartnern laufen. Diese sind in zwei sehr unterschiedlichen Domänen angesiedelt: Der Erste behandelt den technischen Service für den Maschinenbau und der Zweite dreht sich um das Versicherungsgeschäft.

4.6 Use Case 1: Taxonomie für den technischen Service

In diesem Use Case stellt die Empolis Intelligent Views GmbH die Daten zur Verfügung. Die Empolis-Gruppe bietet Dienstleistungen zur Entscheidungsunterstützung und zum Wissensmanagement von Unternehmen an. Ein Kunde von Empolis ist die HOMAG Group AG, ein mittelständisches Unternehmen, welches hauptsächlich Holz verarbeitende Maschinen produziert und vertreibt. Um nachgelagerte Anwendungen im Support, wie Suche oder Entscheidungsunterstützung, zu verbessern, soll hier eine allgemeine Taxonomie aus dem Maschinenbau auf die HOMAG zugeschnitten werden. Diese initiale Taxonomie bildet allgemeine Schäden und Probleme ab, die Maschinen betreffen können, z. B. „senkt nicht“ und „zieht nicht“ als Unterbegriffe von „keine Funktion“. Das zur Verfügung stehende Textkorpus enthält etwa 1,4 Mio. Support-Anfragen. Diese wurden über mehrere Jahre gesammelt, sind mehrheitlich auf Deutsch verfasst, allerdings nicht weiter klassifiziert oder kuratiert. Verfasser sind meistens Wartungstechniker. HOMAG exportiert seine Maschinen weltweit und die fortschreitende Globalisierung ist auch hier erkennbar: Es ist keine Seltenheit, dass etwa ein italienischer Wartungstechniker in einer polnischen Fabrik ein Problem an der deutschen HOMAG-Maschine beheben soll. Die textuelle Verewigung des Prozesses im Support-Ticket geht dann mit den erwartbaren Problemen in der Orthografie einher.

Wie trotzdem zielgerichtet aus diesem großen Textfundus Wissen extrahiert werden kann, wird in diesem Abschnitt beschrieben. Wir folgen hierbei dem in „Erweiterung der Taxonomie“ geschilderten Prozess: Zuerst wird eine Pipeline aufgebaut, die die Rohtexte so prozessiert, dass sie besser verarbeitbar sind. Anschließend werden Erwähnungen von bekannten Begriffen den passenden Knoten in der allgemeinen Taxonomie zugeordnet. Dieses bildet die Trainingsmenge für ein semiinduktives KGC-Modell. Das Modell lernt ein Alignment zwischen den Textkontexten und den Taxonomieknoten. Ziel ist, dass das Modell die Plausibilität einschätzen kann, ob ein Textkontext ein Synonym oder ein Hyperonym eines bekannten Knotens enthält. Um die Bedienoberfläche mit Vorschlägen zu befüllen, werden Texte, in denen keine bekannten Entitäten erwähnt wurden, mit dem

semiinduktiven KGC-Modell geprüft. Die, denen eine ausreichende Plausibilität zugeschrieben wird, kommen in die Vorschlagsliste für einen Domänenexperten. Dieser sichtet diese Ergebnislisten und fügt der Taxonomie interessante Phrasen hinzu.

4.6.1 Vorverarbeitung

Der erste Schritt betrifft das Säubern der Rohdaten. Ziel der Vorverarbeitung ist es, offensichtlich unnütze Information zu entfernen und die Daten in Sätze zu zerlegen. Im konkreten Fall stecken in den Tickets häufig Metainformationen wie Mail-Header, Zeitstempel und andere Nutzinformation des Ticketsystems. Diese werden per regulärem Ausdruck identifiziert und gelöscht. In einem weiteren Schritt wird automatisch die Sprache erkannt, um nur die Tickets in deutscher Sprache zu behalten. Wir verwenden hierfür polyglot.¹⁰ Die deutschen Tickets werden dann in ihre Worte zerlegt und Satzweise getrennt. Hierzu bieten sich SoMaJo,¹¹ Haystack¹² oder spaCy¹³ an. Wir verwenden spaCy. Das so vorverarbeitete Textkorpus kann nun verwendet werden, um Vorkommen bekannter Phrasen zu identifizieren. Hier gibt es eine Besonderheit des Use Case: Probleme an Maschinen stellen oft Multiwortphrasen dar, deren Ordnung eine wichtige Rolle spielt. Möchte man etwa „**fährt nicht hoch**“ finden, soll auch „**fährt morgens nicht korrekt hoch**“ getroffen werden, nicht jedoch „... finde ich **nicht**, aber **fährt** wieder korrekt **hoch**“. Also sowohl die Ordnung muss eingehalten werden als auch Abstand zwischen den Worten erlaubt sein. Eine weitere Anforderung ist, dass nicht über Satzteile hinweg getroffen werden soll: „... kam nicht (Achse **fährt nicht** aus) **hoch**“. Um die Reichweite zu erhöhen, vergleichen wir die Phrasen sowohl mit der Stammform (wir verwenden CISTEM¹⁴) als auch mit dem Lexem¹⁵. Um flexibel auf übliche Wortstellungen zu reagieren, wurde von uns eine Menge von Regeln definiert, nach denen die Phrasen expandiert werden. Im oberen Beispiel greift die Regel „A **nicht** B“ → „**nicht** B A“ („fährt **nicht** hoch“ wird erweitert durch „**nicht** hoch fährt“).

Um dieses Matching abzubilden, wurde eine spaCy-Komponente entwickelt: SCAPHRA.¹⁶ Mithilfe dieses Systems haben wir nun einen Datensatz, bei dem jedem Knoten eine Menge von Text zugeordnet ist. Diesen Datensatz verwenden wir, um ein neuronales Modell zu trainieren.

¹⁰<https://github.com/aboSamoor/polyglot>.

¹¹<https://github.com/tsproisl/SoMaJo>.

¹²<https://haystack.deepset.ai>.

¹³<https://spacy.io>.

¹⁴<https://github.com/LeonieWeissweiler/CISTEM>.

¹⁵<https://spacy.io/api/lemmatizer>.

¹⁶<https://github.com/lavis-nlp/scaphra>.

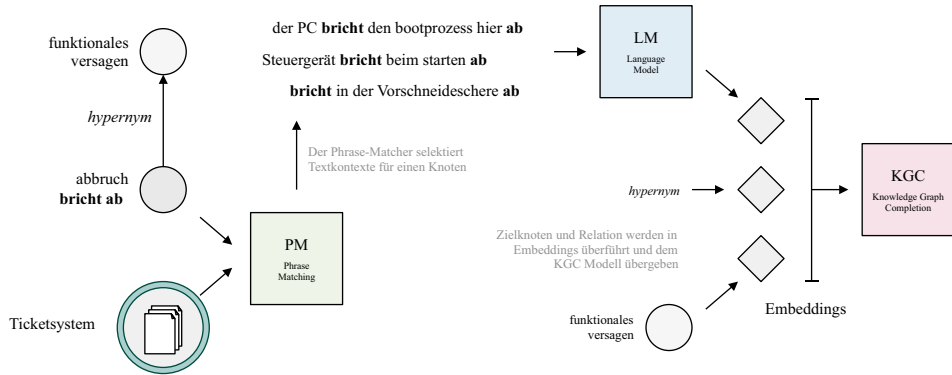


Abb. 4.11 Übersicht über das Modelltraining des semiinduktiven KGC-Modells

4.6.2 Modelltraining

Für diesen Anwendungsfall haben wir uns entschieden, mithilfe eines semiinduktiven KGC-Modells Texte auszuwählen, die im Tool angezeigt werden sollen. Bisher gibt es zwei öffentliche Modelle, BLP¹⁷ (Daza 2021) und IRT2M¹⁸ (Hamann 2022).

Das konkret eingesetzte Modell, mit dem auch die später vorgestellten Ergebnisse erzielt wurden, ist eine Variante von IRT2M. Konzeptuell funktioniert das Modell folgendermaßen (Abb. 4.11): Ein LLM (in diesem Fall ein BERT) kodiert den Text, der im vorigen Schritt einem Knoten zugeordnet wurde, und produziert einen den Eingabesatz beschreibenden Feature-Vektor. Ein KGC-Modell nimmt dieses Embedding entgegen und errechnet einen Score für das Tripel aus Eingabesatz-Embedding, Vektorrepräsentation des Relationstyps und der Zielentität. Dieser Score drückt die Wahrscheinlichkeit aus, dass der gegebene Satz eine neue Entität enthält, die über die betrachtete Relation mit der Zielentität verknüpft werden sollte. Im Training wird also ein Tripel aus dem Graphen ausgewählt, dann wird eine der Entitäten durch den ihr zugeordneten Text ersetzt. Nachdem die Vektorrepräsentationen per KGC-Modell verrechnet wurden, belohnt die eingesetzte Loss-Funktion hohe Scores für korrekte Text-Relation-Entitäts-Tripel. Die Menge an zugeordneten Textkontexten je Taxonomieknoten variiert sehr stark (zwischen 100 und 10.000). Für das Training werden immer zufällige Texte aus den zugeordneten Mengen ausgewählt.

¹⁷ <https://github.com/dfdazac/blp>.

¹⁸ <https://github.com/lavis-nlp/irt2m>.

4.6.3 Inferenz

Mit dem trainierten Modell wird nun die große Menge an Tickets nach neuen Fakten durchstöbert. Hierzu produziert das Modell für jede Textstelle einen Score für jeden Relationstyp (Hyperonym, Synonym) und alle möglichen Entitäten der bekannten Taxonomie. Für jeden bestehenden Knoten der Taxonomie entsteht so im Umkehrschluss eine Liste an zugeordneten Texten, jeweils mit dem Score, den das Modell errechnet hat. Sortiert man absteigend nach Score, so findet man die Texte, in denen das Modell sich am sichersten ist, dass sich in ihnen entweder ein neues Synonym oder ein Unterbegriff verbirgt. Diese Listen werden per Webtool einem Domänenexperten gezeigt. Dieser sichtet die Listen und wählt neue Phrasen aus, die der Taxonomie hinzugefügt werden sollen.

4.6.4 Expertenevaluation

Um die Güte des Ansatzes einzuschätzen, arbeitete ein Domänenexperte ca. zwei Stunden lang mit der prototypischen GUI. Der Explorationsprozess war dabei komplett frei: Es gab keine Vorgaben, welche Teile der Taxonomie erweitert werden sollten, und die Ergebnislisten enthielten ungefilterte Modellvorschläge. Die Qualität der Vorschläge variiert erwartbar sehr stark, je nach Entität. Zusammengesetzte Hauptwörter, wie sie im Deutschen häufig auftreten, werden mit hoher Trefferrate gefunden („Stromschwankungen“, „Druckschwankungen“ ... als Unterbegriffe von „Wert schwankt“). Des Weiteren findet das Modell zuverlässig Phrasen, die eine Überlappung aufweisen („Drehmomentüberlast“, „Bleederüberlast“ ... als Synonyme für „Überlast“). Schwerer tut sich das Modell mit kompletten Abweichungen wie „Formschwabbeln“ als ein Unterbegriff für „unerwünschter Effekt“. Aber auch diese Begriffe tauchen in den Ergebnislisten auf. In unserem Fall fügt der Domänenexperte mithilfe des Tools der Taxonomie etwa 50 neue Phrasen pro Stunde hinzu. Das Annotieren von Phrasen wurde dabei als besonders niedrigschwellig und damit angenehm beurteilt. In einer Stichprobe der Daten (zufällig ausgewählte Texte) wurde ermittelt, dass sich nur in etwa 18 % der Fälle verwertbare Information befindet. Das Modell erhöht die Rate auf diesem Datensatz auf bis zu 40 % (Hamann 2023).

4.7 Use Case 2: Taxonomie für einen Versicherer

Ziel des zweiten Use Case ist der Aufbau einer Taxonomie zur Modellierung von Schadenmeldungen für einen Versicherer. Hier wird beschrieben, wie eine Taxonomie auf Basis einer NER-ähnlichen Anwendung initial definiert und anschließend erweitert wird. Des Weiteren wird untersucht, wie auf Basis des Span-Detektionsmodells eine Transformation der textuellen Schadenmeldungen in eine Graphrepräsentation umgesetzt wird.

Im Gegensatz zum ersten Use Case gibt es hier kein strukturiertes Wissen als Ausgangspunkt. Wie eingangs beschrieben, ist es in einem solchen Fall sinnvoll, mit einem konkreten Anwendungsszenario zu starten, um die initiale Taxonomie zu definieren. Das langfristige Ziel ist die Extraktion von Graphrepräsentationen einzelner Schadenmeldungen, um diese besser vergleichbar und klassifizierbar zu machen. Konkret ließen sich diese Repräsentationen für Clustering und Klassifizierung des Themenbereichs (welche Abteilung ist zuständig), Bewertung der Dringlichkeit oder für Plausibilitätsprüfungen verwenden. Die notwendigen Entitäten und Kanten im Graphen müssen daher Beteiligte (Versicherungsnehmer, Antragsteller, Zeugen ...), beschädigte Objekte, äußere Umstände (Regenwetter, Glatteis, unübersichtliche Baustellen ...) und den Verlauf von Ereignissen abbilden können. Dieser Graph muss aber auch situativ anpassbar sein, um auf veränderte Anforderungen oder zur Verbesserung bestehender Anwendungsszenarien reagieren zu können.

Wir starten datengetrieben und opportunistisch, indem wir ein Szenario ausgewählt haben, bei dem Phrasen zur Klassifikation identifiziert werden müssen. Ein paar Schadenmeldungen (etwa 300) werden zufällig ausgewählt und von Domänenexperten mit einer Grundmenge von Labels annotiert (wir verwenden hierfür LabelStudio¹⁹). Die Festlegung dieser Labelklassen stammt aus mehreren Runden iterativer Anpassung auf Basis einiger Beispiele. Diese kleine Menge annotierter Texte dient nun als Ausgangsbasis für zwei Anwendungen: 1) den Aufbau und die Erweiterung der domänenspezifischen Taxonomie und 2) die automatisierte Extraktion von Graphen auf Basis der vergebenen Labels für nichtannotierte Fälle.

4.7.1 Initiale Definition des Wissensmodells

Die Definition der Labels und der Taxonomie sollen hier anhand eines kleinen Beispiels illustriert werden. Angenommen, man betrachtet diese kleine Schadenmeldung aus dem Kfz-Bereich (Fall 123):

„Bin beim Einfahren in die Hauptstraße bei Nässe leider zu weit nach rechts gedriftet und habe dabei den Wagen schon leicht berührt mit meinem Lkw.“

Der erste Schritt in Richtung einer strukturierten Definition ist die Modellierung der Abfolge der geschilderten Ereignisse, wie sie im Text auftreten – hier beispielsweise, dass das Subjekt an einen Ort gefahren ist, einen spezifischen äußeren Umstand beschreibt („Nässe“), der zu der Beschädigung („schon leicht berührt“) eines Objektes („den Wagen“) geführt habe. Gerüstet mit mehreren dieser Beispiele werden nun im Tandem

¹⁹<https://labelstud.io>.

mit einem Domänenexperten konkrete Entitäten benannt und die ersten Graphen skizziert. Im Beispielfall könnte ein solcher Fallgraph aussehen wie in Abb. 4.12.

Jedoch ist es so, dass die direkte Annotation von derartigen Entitäten und ihrer Relation zueinander, selbst in einem sehr ausgereiften und anpassbaren Tool wie Label-Studio, schnell sehr komplex und unübersichtlich wird. Damit auch Fachexperten (die keine Wissensmodellierer sind) eine Annotation vornehmen können, muss pragmatischer annotiert werden. In diesem Fall werden „flache“ Label benutzt, die oftmals eine bestimmte Relation direkt mit abbilden. So gibt es in jeder Fallbeschreibung immer eine explizit oder implizit erwähnte Person: den Antragstellenden (AS). Wenn etwa der AS das Objekt „den Stand“ beschreibt, das ihm oder ihr gehört und beschädigt wurde, so kann dieses Faktum als Tripel (AS, *objekt_beschaedigt*, „den Stand“) abgebildet oder einfach als Span mit dem Label *as_objekt_beschaedigt* markiert werden. Dies erleichtert zwar den Annotationsprozess, aber skaliert natürlich nicht besonders gut, wenn mehr Subjekte dazukommen (bei mehreren Antragstellenden würden Labels wie *as2_...*, *as3_...* usw. verwendet). Zudem muss ein Graph dann aus dem flachen Label rekonstruiert werden.

Die Menge an Labeln definiert damit auch den Ausgangspunkt für die initiale Definition der Taxonomie. Es werden, unter anderem, übliche Subjekte wie der Antragssteller, Zeugen etc. (die sich als Personen gruppieren lassen), verschiedene Arten von Schäden, verschiedene Orte etc. benannt. Nun sind alle Teile für eine Wissensbasis gegeben: Die Taxonomie (T-Box) definiert die Entitäten, die sich als konkrete Instanzen (A-Box)

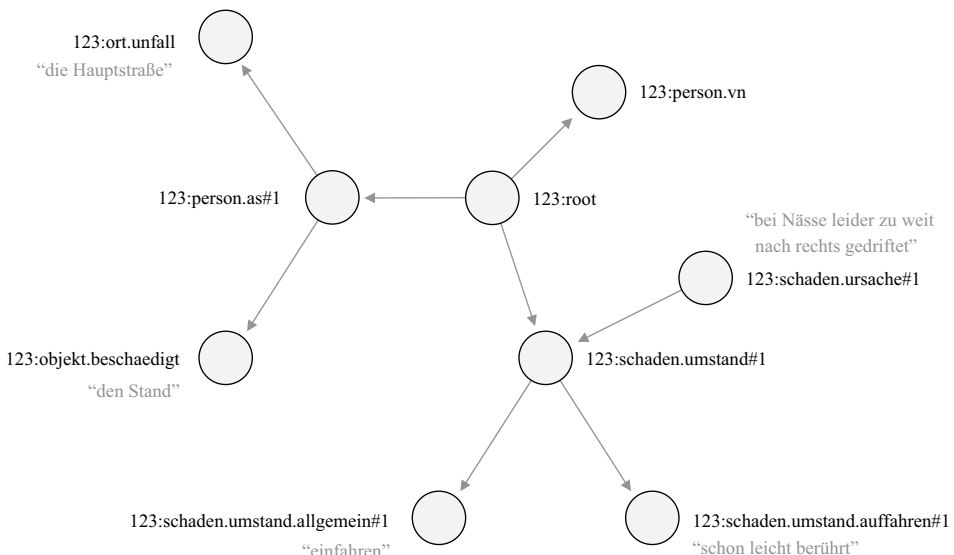


Abb. 4.12 Ein möglicher Fallgraph für eine Schadenmeldung

per Fall wiederfinden. Die Instanzen haben eine Menge von Erwähnungen (Spans) zugeordnet, mit denen sie im Text ausgedrückt werden.

4.7.2 Graphextraktion

Um indessen unbekannte Fälle in Graphen zu überführen, gibt es zwei Möglichkeiten:

1. Ein Span-Klassifizierer-(NER-Modell-)System lernt die flachen Labels auf Basis der bereits annotierten Fälle (Abb. 4.13). Bei einem neuen Fall wird der Klassifizierer verwendet, um den Text mit flachen Labeln zu markieren. Anschließend arbeitet ein Regelsystem auf Basis der Taxonomie und verwandelt die vorhergesagte Labelmenge in einen Graphen. Dieser Ansatz hat jedoch einen gravierenden Nachteil: Die Annotationen sind, vor allem zu Beginn, rar gesät. Das Modell weiß nicht, dass Label wie *as2_schaden_allgemein* und *as3_schaden_allgemein* prinzipiell die gleichen Spans abbilden (und unterschiedliche Personen betreffen). Da nur selten mehrere Antragsteller erwähnt werden, gibt es für diese Label nur wenige Trainingsdaten und die Güte des Modells leidet.
2. Das Regelsystem kann aber auch verwendet werden, um die Trainingsmenge für ein Relationsextraktionssystem (RE) bereitzustellen (Abb. 4.14). In erster Linie bildet es die flachen Label auf Taxonomieinstanzen ab und zieht die Information über die Zuordnungen aus den Labels heraus und bildet sie als Relationen ab. Das RE-Modell kann nun teilen und herrschen: Erst identifiziert es direkt Entitätsinstanzen auf Taxonomieebene (anstelle der flachen Label) und inferiert ihre Beziehung zueinander.

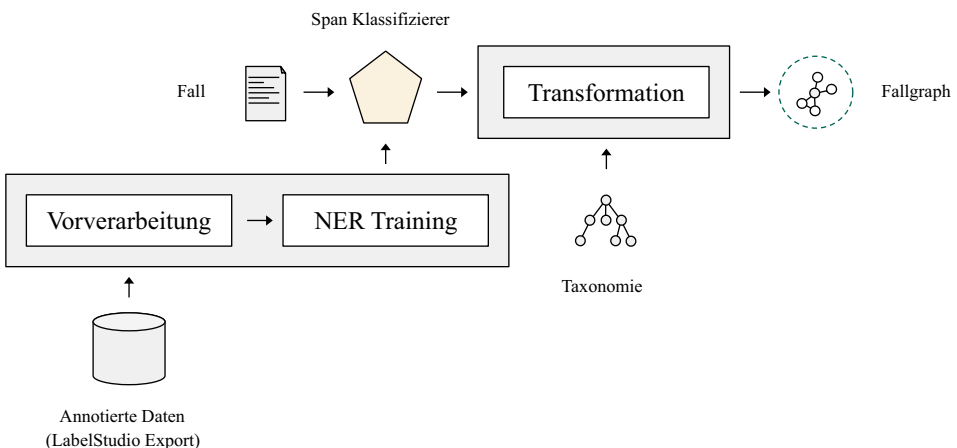


Abb. 4.13 Verwendung eines NER-Modells mit anschließender heuristischer Grapherstellung

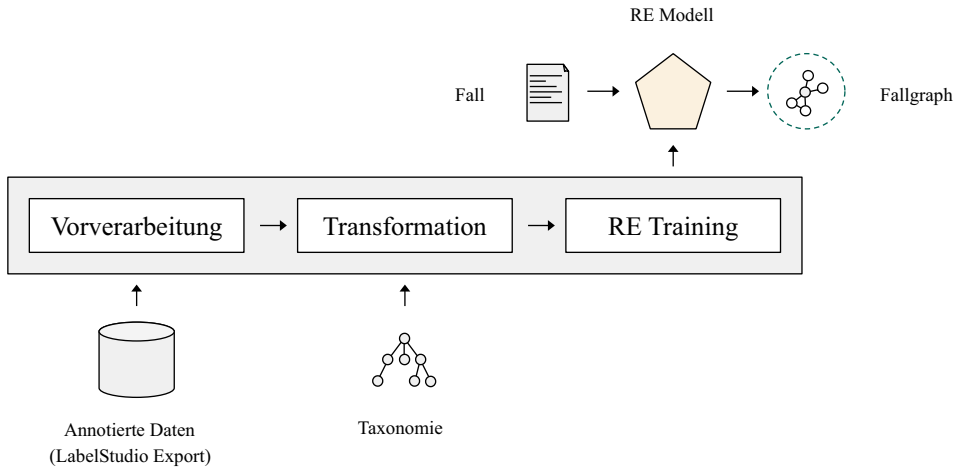


Abb. 4.14 Verwendung eines Relationsextraktionsmodells auf Basis von heuristisch generierten Fallgraphen

Hier ist die Trainingssituation mit wenigen Annotationen schon deutlich besser, da nur die Entität „schaden.allgemein“ klassifiziert werden muss, egal, welches Subjekt es dann tatsächlich betrifft.

4.7.3 Taxonomieerweiterung

Schon bei diesem Anwendungsfall ist schnell beobachtbar, dass die Menge an annotierten Daten sehr unterschiedlich und breit gefächert ist: Sammelbegriffe, wie „Schadensumstand: allgemein“ weisen sehr viel mehr Beispiele auf als spezifische Umstände, wie zum Beispiel „Schadensumstand: abkommen“. Dies ist ein Nebeneffekt, wenn die Annotation auf zufällig angeordneten Fällen abgearbeitet wird. Zudem wird sich herausstellen, dass die Güte von nachgelagerten Anwendungen stark vom Extraktionsprozess abhängig ist. Manche Anwendungsfälle erfordern weitere explizite Verfeinerungen (wie zum Beispiel spezifischere Schadensumstände) oder, um die Güte für unterrepräsentierte Labelklassen zu verbessern, es muss nach ähnlichen Fallbeschreibungen gesucht werden.

In Zukunft wird eine Weiterentwicklung des in Use Case 1 verwendeten GUI-Tools eingesetzt werden, um dies zu ermöglichen. Auch hier werden die bestehende Taxonomie und ihr zugeordnete Textpassagen für ein Retrieval-Modell verwendet, welches große Mengen von unannotierten Fällen durchstöbert. Domänenexperten können dann zielgerichtet in der Fallbasis suchen („finde Auffahrunfälle, die auf Parkplätzen bei Regenwetter stattgefunden haben“) und die Taxonomie, je nach Anforderung erweitern („Streifschäden sind interessant, waren aber bisher nicht explizit abgebildet“).

4.8 Schlusswort

Die in diesem Kapitel vorgestellten Technologien für interaktive Wissenserfassung befinden sich noch im prototypischen Zustand. Sie zeigen einiges an Potenzial, um in einem betrieblichen Kontext beim Aufbau von domänenspezifischen Wissensbasen zu unterstützen. Es gibt jedoch noch keine fertigen Pipelines dafür und der Prozess muss – wie in zwei Use Cases diskutiert – aus Einzelteilen selbst zusammengebaut werden. Die Anwendungsfelder sind sehr unterschiedlich und damit auch die Anforderungen. Die in diesem Kapitel gezeigten Use Cases unterscheiden sich auch dementsprechend: Im ersten Fall ist das Entdecken von buchstäblichen Phrasen in einer großen Textbasis für eine schon existierende, allgemeine Taxonomie relevant. Hingegen muss im zweiten Use Case auf der grünen Wiese begonnen werden und die Taxonomie ist eng mit ihrem ersten Einsatzzweck – einer Span-Klassifizierung – verbunden. Auf der Modellseite soll das Kapitel einen Einblick vermitteln, wo sich der momentane Stand der Forschung befindet. Hier stellen sich dann in der betrieblichen Praxis leider häufig noch Probleme kaufmännischer und administrativer Natur: Teams mit sehr unterschiedlichen Qualifikationen müssen zusammengestellt werden und eng zusammenarbeiten. Domänenexperten werden gebunden. Das birgt auch wegen des schwer bezifferbaren Nutzens ein finanzielles Risiko. Zudem benötigt man spezielle Hardware, wie z. B. Grafikkarten, und entsprechend ausgebildete Entwickler, die sie programmieren können. Im Laufe der Zeit werden aber auch für diese Technologien die Einstiegshürden sinken und damit die Chancen, diesen Prozess einmal selbst auszuprobieren, steigen. Die Vorteile eines gut gepflegten Wissensmodells liegen darin, dass es für viele Anwendungsfälle (wieder-) verwendet werden kann. Für die konkreten Use Cases entsteht also die Chance, sich einen Wettbewerbsvorteil durch schnellere und präzisere Verarbeitung von Service- und Schadensfällen zu verschaffen.

4.9 Empfehlungen

Wir haben folgende Empfehlungen für technische Entscheider, die überlegen, solche Wissensmodelle aufzubauen:

- Die Entwicklung von KI-Modellen kann unabsehbar viel Potenzial haben. Das gilt sowohl für schlechte als auch gute Performance. Generell gilt: Ein Modell wird schlechte Daten nicht magisch in wertvolle Informationen verwandeln. Eine realistische Einschätzung des Potenzials ist wichtig.
- Man sollte sich im Klaren sein, welche konkreten Probleme gelöst werden sollen und welche Anforderungen daraus an das Wissensmodell erwachsen. Am besten beginnt man mit einem konkreten Use Case und erweitert die Wissensbasis opportunistisch für weitere Anwendungsfälle.

- Bevor man mit der Entwicklung eines eigenen Modells beginnt, sollte man Ansätze „von der Stange“ probieren. Das hilft, das Potenzial einzuschätzen und ein tieferes Verständnis von der Datenlage zu bekommen.

Literatur

- Bassignana, E. and Plank, B., 2022. What Do You Mean by Relation Extraction? A Survey on Datasets and Study on Scientific Relation Classification. arXiv preprint [arXiv:2204.13516](https://arxiv.org/abs/2204.13516).
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. and Agarwal, S., 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33, pp.1877–1901.
- Clark, K., Luong, M. T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint [arXiv:2003.10555](https://arxiv.org/abs/2003.10555).
- Daza, D., Cochez, M. and Groth, P., 2021, April. Inductive entity representations from text via link prediction. In *Proceedings of the Web Conference 2021* (pp. 798–808).
- Devlin, J., Chang, M.W., Lee, K. and Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805).
- Hamann, F., Ulges, A. and Falk, M., 2022. IRT2: Inductive Linking and Ranking in Knowledge Graphs of Varying Scale. *Proceedings of the 1st Workshop on Text Mining and Generation co-located with the 45th German Conference on Artificial Intelligence KI 2022*
- Hamann, F. and Ulges, A., 2023, July. Domain-Specific Knowledge Graph Adaption with Industrial Text Data. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (pp. 281–293). Cham: Springer Nature Switzerland.
- Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. and Soricut, R., 2019. Albert: A lite bert for self-supervised learning of language representations. arXiv preprint [arXiv:1909.11942](https://arxiv.org/abs/1909.11942).
- Li, J., Sun, A., Han, J. and Li, C., 2020. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1), pp.50–70.
- Radford, A., Narasimhan, K., Salimans, T. and Sutskever, I., 2018. Improving language understanding with unsupervised learning.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I., 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8), p.9.
- Ruffinelli, D., Broscheit, S. and Gemulla, R., 2020. You can teach an old dog new tricks! on training knowledge graph embeddings.
- Sevgili, Ö., Shelmanov, A., Arkhipov, M., Panchenko, A. and Biemann, C., 2022. Neural entity linking: A survey of models based on deep learning. *Semantic Web, (Preprint)*, pp.1–44.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Das Projekt OdeNet: Aufbau eines semantischen Netzes für die deutsche Sprache

5

Melanie Siegel 

Kernaussagen

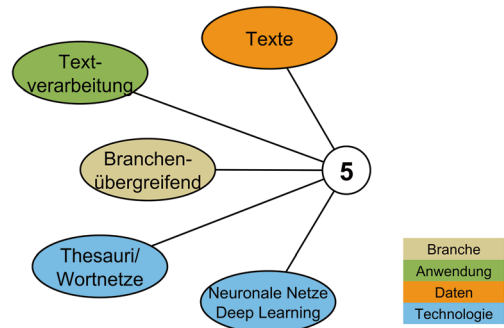
1. Mit Verfahren der Sprachtechnologie ist es möglich, in begrenzter Zeit und mit begrenzten Ressourcen ein Wordnet aufzusetzen.
2. Mit dem neuen Wordnet steht relevante Information zur Verfügung, die für NLP-Anwendungen nützlich ist.
3. Mit dem konsequenten Open-Source-Ansatz wird die Community in die Entwicklung (Korrekturen und Erweiterungen) eingebunden.
4. Die Einbindung in einen multilingualen Kontext im Rahmen der Initiative „Open Multilingual Wordnet“ erlaubt es, Informationen aus Wordnets anderer Sprachen zu nutzen und zu übertragen.
5. Gleichzeitig macht die multilinguale Zugriffsmöglichkeit auf Konzepte interessante Forschungsergebnisse und Anwendungen möglich.

5.1 Einleitung

Dieses Kapitel stellt die Entwicklung eines deutschen Wordnet dar. Abb. 5.1 zeigt die thematische Einordnung des Kapitels in das Gesamtkonzept des Buchs. Das Wordnet selbst stellt eine Technologie der Wissensrepräsentation dar. Wir nutzen u. a. neuronale Netze und Methoden aus dem Deep Learning, um Einträge für das Wordnet zu generieren. Als Inputdaten dienen Textdaten. Das entstandene Wordnet wird für die Textverarbeitung genutzt, die branchenübergreifend relevant ist.

M. Siegel (✉)
Hochschule Darmstadt, Darmstadt, Deutschland
E-Mail: melanie.siegel@h-da.de

Abb. 5.1 Thematische Einordnung des Kap. 5



Informationen über Wörter und Bedeutungen werden traditionell in Wörterbüchern gespeichert. Mit der zunehmenden Bedeutung der automatischen Verarbeitung von Sprache entstand ein Bedarf an maschinenlesbaren Wörterbüchern. In diesem Kontext entstanden Wordnets, die lexikalische Informationen in einem Format speichern, das von sprachverarbeitenden Systemen genutzt werden kann.

Ein Wordnet ist eine lexikalische Datenbank mit semantischen Beziehungen zwischen Wörtern in einer bestimmten Sprache. Basis von Wordnets sind Synsets: Gruppen von Synonymen in der Sprache, die für die Bedeutungskonzepte stehen.

Das erste Wordnet wurde für die englische Sprache an der Princeton University erstellt (auch bekannt als Princeton WordNet, Fellbaum 1998). Als die Nützlichkeit von Wordnets als lexikalische Ressourcen für unterschiedlichste Sprachtechnologieanwendungen deutlich wurde, wurde das Princeton WordNet (PWN) erweitert und Wordnets in anderen Sprachen wurden von Grund auf neu erstellt.

Das PWN wird in elektronischer Form u. a. als Teil von NLTK (Natural Language Toolkit) verbreitet und kann mit einer entsprechenden Python-Bibliothek genutzt werden. NLTK bietet Übersetzungen für Synsets in verschiedene Sprachen, obwohl diese Übersetzungen unvollständig sind. Das bedeutet, dass nicht jedes Synset im Englischen eine äquivalente Übersetzung in einer anderen Sprache hat. Es gibt auch Wordnets in anderen Sprachen, die völlig unabhängig von PWN entwickelt wurden, wie z. B. GermaNet (Hamp und Feldweg 1997). Viele dieser Wordnets enthalten qualitativ hochwertige Daten, die ressourcen- und zeitaufwendig manuell erstellt wurden. Daher sind diese Wordnets zum Teil kommerziell lizenziert und nicht frei nutzbar (und auch nicht z. B. Teil von NLTK).

OMW ist ein Open-Source-Projekt, das mit dem Ziel ins Leben gerufen wurde, die Nutzung von Wordnets in mehreren Sprachen mit Open-Source-Lizenzen zu erleichtern (Bond und Foster 2013). OMW hat den zusätzlichen Vorteil, äquivalente Synsets in verschiedenen Sprachen zu verbinden (Bond et al. 2016). Diese Verbindung entsteht durch einen interlingualen Index mit dem Namen „ILI“. Die englische Version des OMW (EWN, McCrae et al. 2020, 2019) ist im Grunde eine Kopie des PWN, mit eini-

gen Verbesserungen und Ergänzungen, vor allem der Hinzufügung eines interlingualen Indexes für jedes Synset. Viele der OMW-Wordnets in anderen Sprachen wurden unter Verwendung der bereits vorhandenen Übersetzungen in NLTK entwickelt. Diese Übersetzungen wurden extrahiert und in neue Wordnets verpackt. Folglich wurden die entsprechenden Synsets in den resultierenden Wordnets über den ILI miteinander verknüpft. Goodman und Bond (2021) entwickelten die WN-Python-Bibliothek, die für den Zugriff auf die Wordnets des OMW-Projekts in Python verwendet werden kann.

Obwohl NLTK Übersetzungen in vielen Sprachen anbietet, ist Deutsch bisher nicht enthalten. Daher wurde eine Initiative zur Erstellung eines quelloffenen deutschen Wordnet (OdeNet) gestartet, das als Teil des OMW-Projekts entwickelt wurde. Die Entwicklung bestand bisher aus einem grundlegenden Entwicklungszyklus und mehreren Optimierungszyklen, die im Folgenden beschrieben werden. Ziel der Beschreibung ist es, einen Weg aufzuweisen, um neue lexikalische Ressourcen mit multilingualer Anbindung zu erstellen. Ein aktuelles Projekt beschäftigt sich zum Beispiel damit, ein ukrainisches Wordnet auf ähnliche Weise zu erstellen wie OdeNet.

Mit Boxology haben van Harmelen und ten Teije (2019) und Witschel et al. (2021), wie in der Einleitung dargestellt, einen Ansatz zur vereinheitlichten Beschreibung von maschinellen Lernverfahren und wissensbasierten Systemen vorgestellt. In Abb. 5.2 wird gezeigt, wie die Arbeiten an OdeNet in der Boxology-Grafik skizziert werden können. Bestehende Lexika werden als Datenbasis genommen. Dazu kommen multilinguale Wordnets, eine symbolische Repräsentation von Sprache. Beide Informationsquellen werden mit verschiedenen Methoden des maschinellen Lernens verarbeitet, sodass daraus eine erste Version eines deutschsprachigen Wordnet im XML-Format entsteht. Dieses wird von Expert*innen analysiert und dann mit wissensbasierten Methoden weiterentwickelt, sodass OdeNet entsteht.

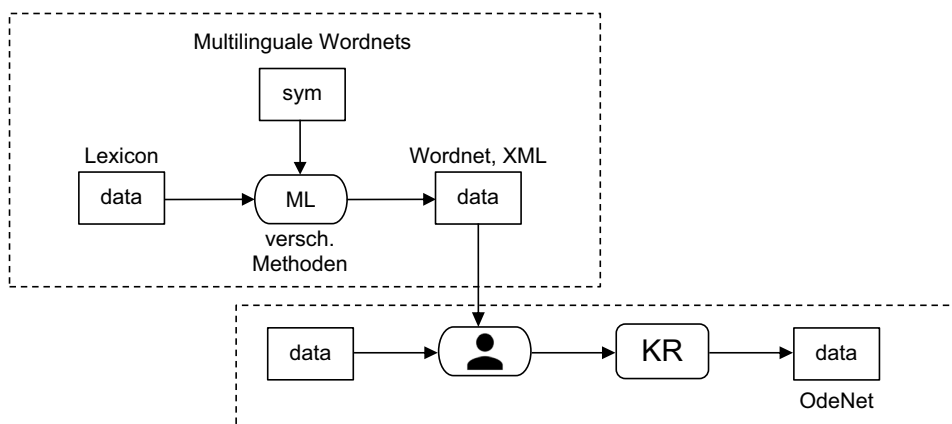


Abb. 5.2 Boxology-Modell des hybriden KI-Systems

Im nächsten Abschnitt stellen wir den Stand der Forschung dar und motivieren die Entwicklung eines deutschen Wordnet. Anschließend folgt die Beschreibung des Problems, das gelöst werden muss. Der erste Entwicklungszyklus beinhaltet die Abbildung des OpenThesaurus auf OdeNet. Darauf folgen einige Optimierungszyklen. Die Anwendungen für OdeNet werden anschließend beschrieben.

5.2 Stand der Forschung

Ein Wordnet aufzubauen oder aus einer anderen Ressource abzuleiten, erfordert erheblichen Aufwand, der nur zum Teil automatisiert werden kann. Nach dem Release der ersten Version von PWN entstanden viele Wordnets für verschiedene Sprachen wie Finnisch (Linden und Carlson 2010), Polnisch (Derwojedowa et al. 2008), Norwegisch (Fjeld und Nygaard 2009), Dänisch (Pedersen et al. 2009), Lettisch (Paikens et al. 2023), Türkisch (Bakay et al. 2021) und Französisch (Sagot und Fišer 2008). Im Rahmen der Initiative Open Multilingual Wordnet (OMW, Bond und Paik 2012; Bond et al. 2015) wurden Wordnets für mehrere Sprachen entwickelt, miteinander verknüpft und unter einer Open-Source-Lizenz veröffentlicht.

Eine manuell gut gestaltete Wordnet-Ressource für Deutsch ist GermaNet (Hamp und Feldweg 1997). GermaNet wurde seit über 20 Jahren entwickelt und ist sehr stabil und präzise. Das Problem ist, dass es nicht unter einer Open-Source-Lizenz steht und daher in Sprachtechnologieanwendungen nicht weitverbreitet ist. Außerdem ist es aufgrund der eingeschränkten Lizenz nicht möglich, GermaNet in die OMW-Initiative einzubinden. Aus diesem Grund haben wir uns entschlossen, eine neue Ressource zu erstellen.

Vossen (1998, S. 11) beschreibt zwei grundlegende Ansätze zur Entwicklung neuer Wordnet-Ressourcen: Im ersten Fall – „Expand“ – werden bestehende PWN-Synsets genommen und lexikalische Einträge für die spezifische Sprache hinzugefügt. Im zweiten Fall – „Merge“ – werden sprachspezifische Ressourcen erstellt und dann mit dem PWN verknüpft.

Ein Beispiel für *Expand* ist das japanische Wordnet (Isahara et al. 2008). Es basiert auf Übersetzungen von PWN ins Japanische. Das japanische Wordnet wird nicht vollständig automatisch erstellt, alle Übersetzungen werden manuell und sorgfältig überprüft. Die Autoren stellten fest, dass es Unterschiede zwischen den Begriffsstrukturen im Englischen und Japanischen gibt, sodass mehrere Synsets nicht übersetzt werden konnten. Weitere Beispiele für *Expand* sind das finnische (Linden und Carlson 2010) und das französische (Sagot und Fišer 2008) Wordnet, ebenso wie das Wordnet für Guarani (Chiruzzo et al. 2023).

Das russische Wordnet (Alexeyevsky und Temchenko 2016) ist ein Beispiel für *Merge*. Es basiert auf einem einsprachigen Wörterbuch und den darin enthaltenen Wortdefinitionen. Die Idee ist, dass Definitionen Hyperonyme (Oberbegriffe) der definierten Wörter enthalten, oft in der Form von WORD:HYPERNYM ..., und dass diese Infor-

mationen genutzt werden können, um hierarchische Strukturen im Wordnet aufzubauen. Weitere Beispiele für *Merge* mit zum Teil anderen Ideen sind das polnische Wordnet (Derwojedowa et al. 2008), das norwegische Wordnet (Fjeld und Nygaard 2009), das dänische Wordnet (Pedersen et al. 2009) und das türkische Wordnet (Bakay et al. 2021). Der Ansatz der OdeNet-Initiative ist ebenfalls *Merge*. Wir verwenden ein bestehendes Synonymwörterbuch und versuchen, die Synsets mit dem PWN zu verknüpfen.

Braslavski et al. (2016) beschreiben die Erstellung eines großen Thesaurus für Russisch mittels Crowdsourcing. Die Daten werden direkt im Wordnet-Stil gesammelt, aber die Synsets sind nicht mit dem OMW verknüpft. Die Basisdaten für OdeNet wurden ebenfalls im Crowdsourcing-Stil im OpenThesaurus-Projekt generiert. Das OpenThesaurus-Projekt (Naber 2004) ist eine Crowd-Initiative zum Aufbau eines deutschen Synonymlexikons. Es enthielt zum Start des Projekts im April 2017 rund 120.000 lexikalische Einträge in etwa 36.000 Synsets.

5.3 Entwicklung eines Wordnet für deutsche Sprache: Problembeschreibung

5.3.1 Repräsentation der lexikalischen Bedeutung in Wordnet

Wordnet¹ ist ein maschinenlesbares Lexikon auf psycholinguistischer Grundlage. Dabei wird der Schwerpunkt auf lexikalische Einheiten gelegt, anders als zum Beispiel bei Ontologien, die den Schwerpunkt auf Konzepte legen. Das lexikalische Wissen wird in Gruppen von Synonymen – sogenannten Synsets – organisiert und gruppiert. Die grundlegenden Relationen in Wordnet sind Synonymie, Hyponymie, Antonymie und Meronymie (Abb. 5.3), wobei im Laufe der Zeit zahlreiche weitere Relationen wie *mero_member*, *has_feminine* oder auch *diminutive* hinzugekommen sind (McCrae et al. 2021).

Die Datenstruktur der Wordnets in OMW ist eine XML-Struktur (die sich in ein JSON-Format umwandeln lässt). Ein Lexem hat einen „LexEntry“ mit einer eindeutigen ID, der Information über die geschriebene Form, der syntaktischen Kategorie und den Bedeutungen, mit den Links zu den dazugehörigen Konzepten. Hier ist ein Beispiel für das Lexem „Hand“:

```
<LexicalEntry id="w48976">
  <Lemma writtenForm="Hand" partOfSpeech="n"/>
  <Sense id="w48976_12493-n" synset="odenet-12.493-n"/>
  <Sense id="w48976_34710-n" synset="odenet-34.710-n"/>
</LexicalEntry>
```

¹Grundlegende Informationen zu Wordnet finden sich unter anderem in Fellbaum (1998) und Kunze und Lemnitzer (2010).

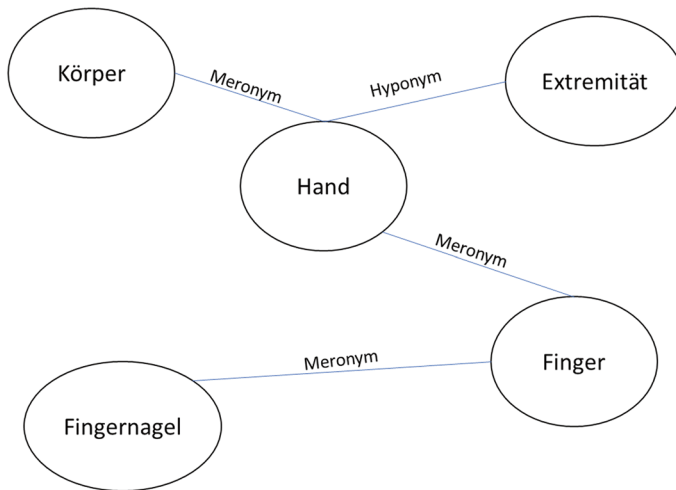


Abb. 5.3 Ausschnitt aus OdeNet-Relationen für das Wort „Hand“

Das „Synset“ repräsentiert das Konzept mit einer eindeutigen ID, dem interlingualen Index, der syntaktischen Kategorie, der Definition, seinen Relationen zu anderen Synsets und einem Beispiel. Hier ist ein Beispiel für das Synset mit der ID „odenet-12.493-n“:

```

<Synset id="odenet-12.493-n" ili="i66153" partOfSpeech="n">
  <Definition>Greiforgan bei Menschen und Affen</Definition>
  <SynsetRelation target="odenet-7450-n" relType="hyponym"/>
  <SynsetRelation target="odenet-33.274-n" relType="hyponym"/>
  <SynsetRelation target="odenet-21.145-n" relType="mero_member"/>
  <SynsetRelation target='odenet-9362-n' relType='hypernym'/>
  <SynsetRelation target='odenet-2024-n' relType='holo_member'/>
  <Example>Sie gibt ihr zur Begrüßung die Hand.</Example>
</Synset>

```

In OMW werden die Konzepte der verschiedenen Sprachen über den „Interlingual Index ILI“ (Fellbaum und Vossen 2008), in einer späteren Version „Collaborative Interlingual Index CILI“ (Bond et al. 2016), miteinander verknüpft. Diese Verknüpfung ist wesentlich, denn sie erlaubt multilinguale Anwendungen von Wordnet. So hat das Synset im englischen Wordnet, in dem „hand“ steht, ebenfalls die ili „i66153“.

5.3.2 Besonderheiten der deutschen Sprache für die Wordnet-Entwicklung

Die Erstellung eines Wordnet für die lexikalischen Informationen einer Sprache erfordert eine eingehende Analyse von Phänomenen wie lexikalische Ambiguität und Mehrwortlexeme. Im Deutschen ist zusätzlich das Problem der Komposita ein Thema.

Es gibt viele Beispiele für lexikalische Ambiguitäten im Deutschen, wie z. B. Mutter (Elternteil oder Teil der Schraube) oder umfahren (herumfahren oder umstoßen). Diese sind in vielen Fällen nicht parallel zu den lexikalischen Ambiguitäten anderer Sprachen, was die Übersetzung zum Zwecke der Verknüpfung der multilingualen Konzepte in OMW erschwert. In den meisten Fällen bleiben Ambiguitäten im Deutschen innerhalb einer syntaktischen Kategorie, denn die Großschreibung der deutschen Substantive verhindert Ambiguitäten zwischen Substantiven und anderen syntaktischen Kategorien, wie es im Englischen oft der Fall ist (z. B. *Change*, Geld, wechseln). Morphosyntaktische Ambiguitäten, die im Deutschen häufig vorkommen, sind für Wordnet nicht relevant, da hier nur Lemmata (Grundformen) aufgenommen werden. Es gibt einige Wörter, die sowohl als Verben als auch als Adjektive verwendet werden können, wie z. B. verlegen (platzieren, an eine andere Stelle legen, veröffentlichen oder peinlich).

Andere Ambiguitäten der syntaktischen Kategorie sind für das aktuelle Vorhaben nicht relevant, weil sie sich auf feinere Verteilungen der syntaktischen Kategorie beziehen, als wir sie im Moment bereitstellen können (Partikel – Präposition, Demonstrativpronomen – Artikel).

Im Bereich der Mehrwortlexeme geht es um komplexe Verbkonstruktionen, wie z. B. „Abschied nehmen“ oder „in Rechnung stellen“. Außerdem gibt es Redewendungen wie „das geht auf keine Kuhhaut“ und „es spottet jeder Beschreibung“. Besonders bei Redewendungen ist es schwierig, die syntaktische Kategorie automatisch zu bestimmen.

Komplexe Substantive werden jedoch nicht wie im Englischen durch Mehrwortausdrücke realisiert, sondern mit Komposita. Nominale Komposita sind im Deutschen sehr produktiv und daher komplex. Sie können sehr lang sein, wie das bekannte Beispiel „Donaudampfschiffahrtskapitänsmütze“. Es ist nicht möglich, eine endliche Liste deutscher Komposita zu erstellen, da immer wieder neue Kombinationen möglich sind. Eine automatische Extraktion mit Analyse aus Textdaten ist komplex, weil es auch hier Ambiguitäten gibt.

Bei regulären deutschen Komposita gibt es eine Hyponymiebeziehung zwischen dem Kopf und dem Kompositum. Zum Beispiel ist Wassereis (Kompositum) ein Eis (Kopf), das aus Wasser besteht, während Eiswasser (Kompositum) Wasser (Kopf) ist, das eiskalt ist.

Zum Modifikator können unterschiedliche Beziehungen bestehen. Die Regelmäßigkeit der Hyponymiebeziehung zum Kopf deutscher Komposita kann genutzt werden, um Relationen zu OdeNet hinzuzufügen.

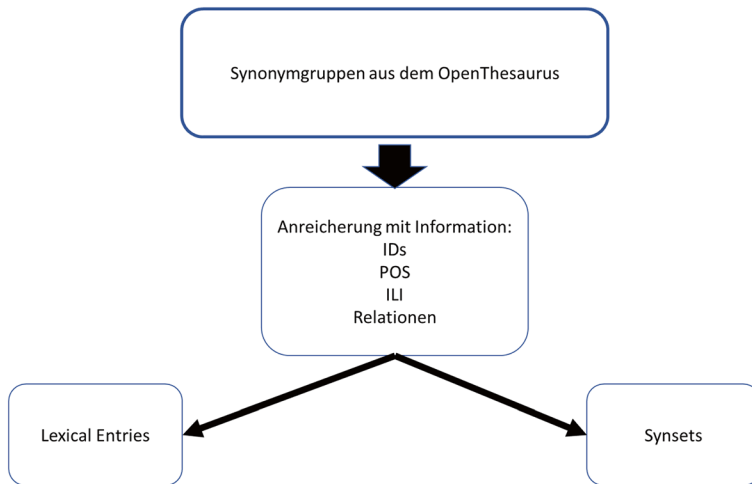


Abb. 5.4 Erster Entwicklungszyklus

5.4 Erster Entwicklungszyklus: Automatische Erstellung von OdeNet aus anderen Ressourcen²

5.4.1 Abbildung von Synonymgruppen des OpenThesaurus in Wordnet Synsets

Die Basis für den ersten Entwicklungszyklus sind die Synonymgruppen aus dem OpenThesaurus (Naber 2004).³ Dieser Thesaurus ist als Crowd-Initiative entwickelt und steht unter einer Open-Source-Lizenz. Die erste Version von OdeNet wurde vollständig automatisch durch Kompilierung aus OpenThesaurus erstellt (Abb. 5.4).

Die Textdatei enthält ein Synset pro Zeile, wobei die lexikalischen Elemente in jedem Synset durch Semikolon getrennt sind, z. B.:

Mobilität; Unabhängigkeit; Beweglichkeit.

Das Ziel des Transferprozesses für diesen Eintrag ist es, drei lexikalische Einträge und einen Synset-Eintrag zu haben. Das Format ist in Bond et al. (2016) beschrieben. Wir beginnen mit dem Synset:

```

<Synset id="odenet-9784-n" ili="i62097" partOfSpeech="n" dc:de-
scription="the quality of moving freely">
  <SynsetRelation target='odenet-23.172-n' relType='hypernym' />
</Synset>
  
```

²Diese Arbeiten wurden in Siegel und Bond (2021) vorgestellt.

³<https://www.openthesaurus.de/about/download>

Das Synset hat eine eindeutige Synset-ID innerhalb des Wordnet, einen Link zum Collaborative Interlingual Index in „ili“, eine syntaktische Kategorie in „partOfSpeech“ (POS), eine Definition und Relationen zu anderen Synsets. Eine eindeutige Synset-ID lässt sich einfach durch Hochzählen vergeben.

Die erste Aufgabe besteht nun darin, Informationen über die syntaktische Kategorie (POS) zu finden. Wir haben die Python-Bibliothek TextBlob⁴ für die Erkennung der syntaktischen Kategorie verwendet. OdeNet verwendet nur „n“ (Nomen), „v“ (Verb) und „a“ (Adjektiv, Adverb) als POS-Tags, sodass wir die präziseren POS-Tags der Penn Treebank, die TextBlob liefert, auf diese abbilden. Im Falle von Mehrwortausdrücken, wie z. B. „moralische Werte“, nehmen wir den POS-Wert des letzten Wortes im Ausdruck, das in den meisten Fällen die syntaktische Kategorie des Gesamtausdrucks steuert. Das führt allerdings durchaus zu Fehlern, wie wir sehen werden.

Die nächste Aufgabe besteht darin, ein paralleles englisches Synset zu finden, sodass wir den Collaborative Interlingual Index in „ili“ aus EWN übernehmen können. Wir haben die Wörter des Synset im ersten Schritt mit Google-Translate⁵ ins Englische übersetzt. Die Verwendung eines statistischen maschinellen Übersetzungssystems anstelle eines Wörterbuchs hat den Vorteil, dass die Übersetzung auf dem Kontext basiert. Im Falle mehrdeutiger Wörter erfolgt die Entscheidung damit kontextbasiert, wobei der Kontext durch die anderen Wörter im Synset gebildet wird. Unter Verwendung der NLTK-Wordnet-API haben wir dann nach Synsets mit diesen englischen Wörtern gesucht und hatten dadurch Zugriff auf die EWN-Synset-ID und die ILI.

Auf diese Weise haben wir 19.845 OdeNet-Synsets einen ILI zuweisen können, ungefähr der Hälfte der Synsets. Synsets, die nicht verlinkt werden konnten, waren oft Mehrwortlexeme und metaphorische Ausdrücke wie „es kann Gott weiß was passieren“, „für nichts garantieren können“ oder „mit allem rechnen müssen“. OpenThesaurus ist besonders reich an umgangssprachlichen und metaphorischen Ausdrücken. Durch die Übersetzung hatten wir Zugriff auf die Definitionen in EWN und konnten diese unter „dc:description“ übernehmen. Wir hatten ebenfalls Zugriff auf die Relationen in EWN und konnten diese ebenfalls übernehmen, falls die Ziele der Relationen in OdeNet existieren und einen ILI bekommen haben.

Wir konnten auf diese Weise rund 36.000 Synsets in OdeNet erzeugen. Circa 20.000 dieser Synsets konnte ein ILI zugewiesen werden. Die Qualität dieser Übersetzungen wurde in den Optimierungszyklen untersucht.

⁴<https://textblob.readthedocs.io/en/dev/>

⁵<https://translate.google.de/>

5.4.2 Lexikalische Einträge und Bedeutungen

Dies sind die lexikalischen Einträge für die Wörter im Synset „odenet-9784-n“:

```
<LexicalEntry id="w39185">
  <Lemma writtenForm="Mobilität" partOfSpeech="n"/>
  <Sense id="w39185_9784-n" synset="odenet-9784-n"/>
</LexicalEntry>
<LexicalEntry id="w33556">
  <Lemma writtenForm="Beweglichkeit" partOfSpeech="n"/>
  <Sense id="w33556_8203-n" synset="odenet-8203-n"/>
  <Sense id="w33556_9784-n" synset="odenet-9784-n"/>
  <Sense id="w33556_11420-n" synset="odenet-11.420-n"/>
  <Sense id="w33556_19087-n" synset="odenet-19.087-n"/>
</LexicalEntry>
<LexicalEntry id="w35624">
  <Lemma writtenForm="Unabhängigkeit" partOfSpeech="n"/>
  <Sense id="w35624_8795-n" synset="odenet-8795-n"/>
  <Sense id="w35624_9784-n" synset="odenet-9784-n"/>
  <Sense id="w35624_28976-n" synset="odenet-28.976-n"/>
</LexicalEntry>
```

Die lexikalischen Einträge, die zu einem Synset gehören, haben eine Bedeutung (Sense), die auf das Synset verweist. Alle drei lexikalischen Einträge verweisen auf das Synset „odenet-9784-n“. „Beweglichkeit“ und „Unabhängigkeit“ haben weitere Bedeutungen, die auf andere Synsets verweisen. Jeder lexikalische Eintrag hat eine innerhalb von OdeNet eindeutige ID, ein Lemma und eine syntaktische Kategorie. Auch die Bedeutungen haben eine eindeutige ID, die sich aus der ID des lexikalischen Eintrags und der ID des Synset zusammensetzt.

Im ersten Entwicklungszyklus sind ca. 120.000 lexikalische Einträge in OdeNet entstanden.

5.5 Optimierungszyklen⁶

5.5.1 Erster Zyklus – Korrektur der syntaktischen Kategorien

In einer ersten stichprobenartigen Evaluation haben wir herausgefunden, dass lediglich 77 % der syntaktischen Kategorien unter „partOfSpeech“ (POS) korrekt waren. Ein wichtiger Grund dafür ist, dass unsere Datenbasis, der OpenThesaurus, sehr viele Mehr-

⁶Die hier beschriebenen Verfahren wurden auch in (Siegel und Bond 2021) beschrieben.

wortlexeme und auch Umgangssprache enthält. Das führt dazu, dass Standardverfahren für die Zuweisung von syntaktischen Kategorien nicht zuverlässig genug sind. Die Idee, bei Mehrwortlexemen die syntaktische Kategorie des letzten Wortes im Ausdruck zu nehmen, war nicht zuverlässig genug. So wurde z. B. dem Lexem „nach dem Tod“ die Kategorie „n“ zugewiesen, wobei „a“ korrekt wäre.

Ein erster Hinweis auf nicht korrekte syntaktische Kategorien waren Synsets mit Lexemen darin, die unterschiedliche syntaktische Kategorien zugewiesen haben. Diese extrahierten wir aus OdeNet und korrigierten sie manuell. Eine erneute Evaluation zeigte einen Anstieg der korrekten POS-Werte auf 90 %.

Die nächste Idee war, mit Wortendungen zu arbeiten. So sind im Deutschen Wörter, die auf „-ung“, „-heit“ oder „-keit“ enden, immer Nomen, während Wörter, die auf „-lich“ enden, Adjektive sind. Außerdem kann die Großschreibung von Nomen mit in Betracht gezogen werden. Mit diesen Informationen wurden weitere POS-Werte automatisch korrigiert. Die Evaluation zeigte einen Anstieg der korrekten POS-Werte auf 93,3 %.

5.5.2 Nutzung einer Kompositaanalyse für Relationen

Regelmäßige deutsche Komposita haben eine Hyponymiebeziehung zwischen dem Kompositum und seinem letzten Element,⁷ dem Kopf. So ist das Hyperonym von „Jägerschnitzel“ das „Schnitzel“, das Hyperonym von „Eiswasser“ das „Wasser“ und von „Tannenbaum“ der „Baum“. Die Relationen der Komposita zu den anderen Elementen sind vielfältig und werden nicht berücksichtigt.

Diese Regelmäßigkeit kann nun genutzt werden, um OdeNet-Hyperonymierelationen automatisch hinzuzufügen. Zu diesem Zweck haben wir ein einfaches Tool implementiert, das deutsche Komposita erkennt und den Kopf ausgibt. Das Tool basiert auf einer Liste von Nomen aus der TIGER-Baumbank (Brants et al. 2004) und dem pyphen-Modul,⁸ das Wörter in Silben segmentiert. Wenn ein Wort aus zwei Silben besteht, wird in der Nomenliste nachgesehen, ob beide schon ein Nomen bilden, wie bei „Haustür“. Wenn das nicht der Fall ist, wird versucht, vom Ende des ersten Elements eine Fuge abzuziehen, wie bei „Wirtshaus“ (Wirt+s+Haus). In einigen Fällen sind die Wörter mit Fuge als Genitivvariante oder als Pluralvariante in der Nomenliste enthalten, sodass dieser Schritt entfallen kann. Wenn das Wort aus mehr als zwei Silben besteht, werden verschiedene Silbenkombinationen getestet, bis eine Analyse gefunden ist. Hier ist der Prozess mit den Zwischenergebnissen für das Wort „Herstellungskosten“:

⁷Unregelmäßige deutsche Komposita sind solche, die nicht-kompositional interpretiert werden, wie „Dummkopf“.

⁸<https://pyphen.org/>

("Herstellungskosten")

SILBEN:

['Her', 'stel', 'lungs', 'kos', 'ten']

SILBEN-KOMBINATIONEN:

['Herstel', 'Stellungs', 'Lungskos', 'Kosten', 'Herstellungs', 'Stellungs', 'Lungskosten', 'Herstellungskosten', 'Stellungskosten']

KOMPONENTEN DES KOMPOSITUMS:

['Herstellungs', 'Kosten']

Mit diesem Tool haben wir alle lexikalischen Einträge in OdeNet, die keine Mehrwortlexeme sind, analysiert und 19.115 Komposita extrahiert. In den Fällen, in denen der Kopf in OdeNet nicht ambig war (also nur einem Synset zugewiesen war), haben wir automatisch eine Hyperonymrelation zu diesem Synset und eine Hyponymrelation zurück zum Synset des Wortes in OdeNet eingefügt. Da diese Relationen zwischen Synsets und nicht zwischen lexikalischen Einträgen bestehen, profitieren auch die Synonyme des Kompositums von der zusätzlichen Relation. Ein Beispiel: Das Wort „Butterbrot“ kann als Kompositum aus „Butter“ und „Brot“ analysiert werden. Daher wurde eine Hyponymrelation zwischen den Synsets `odenet-11.770-n` ['Brotlaib', 'Wecken', 'Brot'] und `odenet-10.073-n` ['Knifite', 'belegtes Brot', 'Scheibe', 'Butterbrot', 'Schnitte', 'Bemme', 'Stulle'] eingefügt.

Es gab einige interessante Ausnahmen für die Hyperonymrelation von Kompositum und Kopf. In einigen Fällen ist das Kompositum synonym zu seinem Kopf, wie im Fall von „Fachterminus“ und „Terminus“. In diesen Fällen treten diese im gleichen Synset auf und konnten daher automatisch ausgenommen werden.

Komplexer sind die Fälle mit Negationen in Komposita. Ein „Nichtraucher“ ist kein Hyponym zum „Raucher“, sondern ein Antonym. Andererseits ist „Nichteisenmetall“ eine Art „Metall“. Daher mussten wir alle Komposita mit Negationen von Hand überprüfen.

Ein anderes Problem bilden die Komposita mit „Pseudo-“ oder „Schein-“. Ist eine „Scheinschwangerschaft“ eine „Schwangerschaft“? Ist eine „Pseudodokumentation“ eine „Dokumentation“? Wir haben entschieden, diese nicht als Hyponyme zu betrachten und von der automatischen Relationszuweisung auszunehmen.

Von den Köpfen der 19.115 Komposita, die in OdeNet gefunden wurden, hatten 12.132 mehrere Bedeutungen und bekamen daher (zunächst) keine zusätzlichen Relationen. In weiteren 1810 Fällen gab es keinen Eintrag für den Kopf in OdeNet, sodass auch diese ignoriert wurden. Zu allen Hyperonymrelationen, die wir hinzugefügt haben, haben wir auch die dazugehörige Hyponymrelation hinzugefügt.

Insgesamt konnten 10.346 Relationen zu OdeNet-Synsets hinzugefügt werden. OdeNet enthält etwa 35.000 Synsets, sodass wir zu 29 % aller Synsets Informationen hinzufügen konnten.

Eine Evaluierung der Kompositaanalyse zeigte eine hohe Präzision von 99 % und einen moderaten Recall von 83 %. Da Präzision bei der automatischen Eintragung von Relationen in OdeNet wichtiger ist als Recall, ist dies ein hervorragendes Ergebnis. Auch die Präzision der neu eingetragenen Relationen liegt auf einem hohen Wert von 98 %.

5.5.3 Nutzung einer Antonymliste für neue Antonymrelationen

Auf der Konferenz „GWC 2023“ stellten Kühn et al. (2023) Listen mit Antonymen vor, die aus Wiktionary automatisch extrahiert wurden. Wir bekamen Zugriff auf die deutsche Liste und analysierten, wie OdeNet davon profitieren könnte, um Antonymrelationen einzufügen.

Die deutschsprachige Liste enthält 45.500 Lemmata mit ihren Antonymen, also eine reichhaltige Informationsquelle.

In der ersten Version waren alle Wörter kleingeschrieben, weil die Prozesse für die englische Sprache konzipiert waren, bei der die Groß- und Kleinschreibung nicht zur Disambiguierung beiträgt. Die Groß- und Kleinschreibung trägt im Deutschen jedoch zur Disambiguierung bei, sodass durch die Vorgehensweise zusätzliche Ambiguität zugefügt wurde:

```
'arm': {'vielfältig', 'begütert', 'privilegiert', 'oft', 'reichlich',
'bein', 'beneidenswert', 'reich', 'betucht', 'situiert', 'wohlhabend',
'steinreich', 'schwerreich', 'voll'}
```

Dies wurde von der Arbeitsgruppe jedoch behoben und diese Art der Ambiguität aufgelöst:

```
'arm': {'privilegiert', 'voll', 'steinreich', 'oft', 'betucht', 'vielfältig',
'reich', 'reichlich', 'schwerreich', 'begütert', 'wohlhabend', 'situiert',
'beneidenswert'}
```

```
'Arm': {'Bein'}
```

Das Konzept „Antonym“ ist in der Arbeit von Kühn et al. (2023) weit gefasst, es werden auch Heteronyme aufgeführt, wie:

```
'Dezember': {'November', 'September', 'Januar', 'April', 'Juni', 'Februar',
'Feber', 'März', 'Oktober', 'Jänner', 'August', 'Mai', 'Juli'}
```

Die Nomen sind dabei oft ambig, wie:

```
'Note': {'Münze', 'Pause'}
```

Für OdeNet haben wir entschieden, Heteronymie und Antonymie zu unterscheiden, auch wenn das bei Nomen konzeptuell oft schwierig ist. Daher haben wir uns im ersten Schritt auf die Adjektive beschränkt.

Die Wordnet-DTD erlaubt es, Antonymierelationen auf der Ebene der Lexeme oder der Synsets einzutragen. Die Entscheidung fiel für die Synset-Ebene, denn mit einem Synset sind oft mehrere Lexeme betroffen, die von der Antonymierelation profitieren können.

Dennoch gehen wir von den Lexemen aus und extrahieren aus OdeNet alle Adjektive, die nicht mehrdeutig sind, also nur zu einem Synset zugehörig. Wenn diese Adjektive einen Eintrag in der Liste von Kühn et al. (2023) haben, werden die Synset-IDs der dazugehörigen Antonyme in OdeNet gesucht. Die Antonymrelation wird eingetragen, wenn das Antonym in OdeNet enthalten ist und auch nur Teil eines einzigen Synsets ist, also auch nicht mehrdeutig. Diese Vorgehensweise vermeidet Ambiguitäten, führt aber dazu, dass nur ein kleiner Teil der Information in den Listen genutzt werden konnte. Es konnten ca. 670 neue Antonymrelationen für Adjektive eingetragen werden, sodass OdeNet aktuell 2740 Antonymrelationen enthält.

Im nächsten Schritt werden wir uns die Nomen noch einmal näher ansehen und starten mit Nomen, die mit „Nicht-“ beginnen, wie „Nichtprimzahl“ oder mit „Un-“ wie „Unfähigkeit“.

5.5.4 Verbesserung der ILI-Verknüpfungen⁹

Ein wichtiges Problem bei der Nutzung von automatischer Übersetzung für die Verknüpfung von äquivalenten Synsets in verschiedenen Sprachen sind Homografen und Polyseme. Das ist besonders problematisch, wenn ein Wort aus der Quellsprache in der Zielsprache mehrdeutig ist.

Ein Beispiel dafür ist das Wort „Unterlegscheibe“ aus OdeNet. Die englische Übersetzung dafür ist „washer“. In EWN finden wir drei Synsets, zu denen „washer“ gehört:

- Name: washer
EWN ID: ewn-10.788.571-n
ILI: i94042
Definition: someone who washes things for a living
- Name: washer
EWN ID: ewn-04.562.157-n
ILI: i60971
Definition: seal consisting of a flat disk placed to prevent leakage

⁹Die hier beschriebenen Verfahren sind auch in Bergh und Siegel (2023) beschrieben.

- Name: washer
EWN ID: ewn-04.561.970-n
ILI: i60970
Definition: a home appliance for washing clothes and linens automatically

Aus dieser Problematik heraus sind im ersten Entwicklungszyklus einige falsche Zuordnungen via ILI entstanden.

Es wird eine Methode benötigt, mit der das korrekte Synset in EWN ausgewählt werden kann, das mit dem Synset von „Unterlegscheibe“ verknüpft werden kann. Mit Sprachkenntnissen im Englischen und Deutschen ist klar, dass das EWN-Synset mit der ID ewn-04.562.157-n und dem ILI i60971 das richtige ist, sodass dieser ILI auch zum Synset von „Unterlegscheibe“ hinzugefügt werden kann. Aber ohne den dazugehörigen Kontext kann diese Auswahl nicht automatisch erfolgen.

Ein weiteres Problem in OdeNet, das daraus entstanden ist, war, dass in vielen Fällen unterschiedliche Synsets im ersten Entwicklungszyklus denselben ILI bekommen haben. Zum Beispiel hatten die Synsets odenet-4330-n ['Anzahl' , 'Zahl'] und odenet-688-n ['Summe' , 'Gesamtmenge'] beide den ILI i35594, der in EWN zum Synset mit den Wörtern „measure“, „amount“ und „quantity“ gehört.

Es wird also eine Methode benötigt, mit der eine einzige korrekte Verknüpfung via ILI zugewiesen werden kann. Die grundlegende Idee hierfür ist, für die Übersetzung mehr Kontext zu nutzen. Dieser Kontext steht in EWN mit den Definitionen für die Synsets zur Verfügung. Wir gehen also bei der Übersetzung von Englisch als der Quellsprache aus und nutzen die Definitionen als Kontext.

Zuerst kombinieren wir die Wörter im EWN-Synset mit ihren Definitionen und übersetzen diese dann automatisch mit DeepL.¹⁰ Dann werden die übersetzten Wörter aus den Ergebnissen extrahiert und es wird nach einem korrespondierenden Wort in OdeNet gesucht. Hier sind die Ergebnisse für das „washer“-Beispiel:

- EWN ID: ewn-10.788.571-n
ILI: i94042
Wort-Definition-Kombination: washer: someone who washes things for a living
Automatische Übersetzung: Wäscher: jemand, der beruflich Dinge wäscht
- EWN ID: ewn-04.562.157-n
ILI: i60971
Wort-Definition-Kombination: washer: seal consisting of a flat disk placed to prevent leakage
Automatische Übersetzung: Unterlegscheibe: Dichtung, die aus einer flachen Scheibe besteht, um ein Auslaufen zu verhindern

¹⁰<https://www.deepl.com>

- EWN ID: ewn-04.561.970-n
 ILI: i60970
 Wort-Definition-Kombination: washer: a home appliance for washing clothes and linens automatically
 Automatische Übersetzung: Waschmaschine: ein Haushaltsgerät zum automatischen Waschen von Kleidung und Wäsche

Es wird deutlich, dass die automatische Übersetzung nun die korrekte Auswahl von i60971 für das Synset von „Unterlegscheibe“ erlaubt.

In dem Fall jedoch, in dem die Zielsprache Deutsch mehrere Bedeutungen für ein Wort aus der Ausgangssprache Englisch enthält, wird eine andere Lösung benötigt. Ein Beispiel hierfür ist das EWN-Synset mit dem ILI i66412:

- EWN ID: ewn-05.621.745-n
 ILI: i66412
 Wort-Definition-Kombination: depth: the intellectual ability to penetrate deeply into ideas
 Automatische Übersetzung: Tiefe: die intellektuelle Fähigkeit, tief in Ideen einzudringen

Für das Wort „Tiefe“ finden sich in OdeNet drei verschiedene Synsets:

- odenet-847-n: ['Tiefe', 'Tiefsinn']
- odenet-6615-n: ['Abgrund', 'Tiefe', 'Schlund', 'Hölle']
- odenet-16.328-n: ['Tiefe', 'Teufe']

Welches OdeNet-Synset soll dem ILI zugeordnet werden? Natürlich sollte dies odenet-847-n sein, aber das kann nicht automatisch entschieden werden.

Mehr als ein EWN-Synset kann außerdem einem einzigen OdeNet-Synset entsprechen. Hier sind zum Beispiel die Wort-Definition-Kombinationen und Übersetzungen der EWN-Synsets mit den ILI i6124 und i68929:

- ILI: i6124
 Wort-Definition-Kombination: ethic: the principles of right and wrong that are accepted by an individual or a social group
 Automatische Übersetzung: Ethik: die Grundsätze des Richtigen und Falschen, die von einem Individuum oder einer sozialen Gruppe akzeptiert werden
- ILI: i68929
 Wort-Definition-Kombination: ethics: the philosophical study of moral values and rules
 Automatische Übersetzung: Ethik: das philosophische Studium der moralischen Werte und Regeln

Für beide Lemmata in den jeweiligen EWN-Synsets ist das übersetzte Lemma im Deutschen „Ethik“, das in den OdeNet-Synsets odenet-10-n [‘Sittlichkeit’, ‘Wertvorstellungen’, ‘Wertmaßstäbe’, ‘Wertesystem’, ‘Moral’, ‘Moralvorstellungen’, ‘Ethik’, ‘sittliche Werte’, ‘moralische Werte’] und odenet-4879-n [‘Ethik’, ‘Morallehre’, ‘Sittenlehre’, ‘Tugendlehre’] steht. Welchem ILI der EWN-Synsets ordnet man welches der OdeNet-Synsets zu?

Da es in den OdeNet-Synsets mehrere Kandidaten für ILI in den EWN-Synsets geben kann, muss eine Klassifizierungsfunktion geschrieben werden, um jedem der Kandidaten eine Gewichtung zuzuweisen, sodass die optimale Zuordnung vorgenommen werden kann. Glücklicherweise ist OdeNet sehr reich an Synonymen (viel mehr als andere Wordnets) und wir können diese Synonyme in Kombination mit einem deutschen Word2Vec-Modell von spaCy¹¹ für die Klassifizierung verwenden.

Zunächst extrahieren wir den Definitionsteil der übersetzten Lemma- und Definitionsübersetzung. Die Inhaltswörter in dieser Übersetzung werden zu einem Vektor (v1) hinzugefügt. Es werden nur Inhaltswörter wie Adjektive, Adverbien, Nomen und Verben berücksichtigt. In gleicher Weise werden alle Synonyme (Lemmata im Kandidaten-Synset) zu einem zweiten Vektor (v2) hinzugefügt. Für jeden Wert in v1 und v2 wird mit der „similarity“-Funktion von spacy ein Ähnlichkeitswert berechnet. Die Ähnlichkeitswerte werden summiert und auf einen Wert zwischen 0 und 1 normalisiert. Dieser gewichtete Wert wird den Synset-Kandidaten in OdeNet zugewiesen, die um den ILI in einem bestimmten EWN-Synset konkurrieren.

Ein anderes Problem ist, dass es im Englischen viele Nomen und Verben gibt, die gleich geschrieben werden, so wie das Wort „search“ (Suche, suchen). Eine Idee, damit umzugehen, ist, Vorverarbeitung zu nutzen, um bessere Ergebnisse der maschinellen Übersetzung zu erzielen. Experimente mit der maschinellen Übersetzung haben gezeigt, dass sich die Übersetzungsergebnisse für Verben vom Englischen ins Deutsche verbessern, wenn man „to“ vor das Verb setzt. In Fällen, in denen wir englische Substantive und Verben mit der gleichen Schreibweise haben, hilft dies der maschinellen Übersetzung auch, die syntaktische Kategorie korrekt zu unterscheiden. Zum Beispiel funktioniert die maschinelle Übersetzung des Verbs „search“ besser, wenn das Verb in seine Infinitivform „to search“ gebracht wird. Dann wird es auch in der Zielsprache in ein Verb übersetzt. Das EWN-Synset mit ILI i28263 bezieht sich auf das Verb „search“. Die Wort-Definitions-Kombination, angereichert mit „to“, ist:

„to search: try to locate or discover, or try to establish the existence of“.

Für den maschinell übersetzten deutschen Text waren in einigen Fällen dann Nachbearbeitungen erforderlich. So fügte die maschinelle Übersetzung in einigen Fällen das

¹¹ <https://spacy.io/>

Tab. 5.1 Korrekturen an OdeNet

	EWN	OdeNet vorher	OdeNet nachher
Synsets	120.053	36.159	36.159
Synsets mit eindeutigem ILI	117.480	13.818	19.547
Synsets ohne ILI	2573	16.376	16.612
Synsets mit doppelten ILI	0	5965	0
Doppelte ILI	0	3703	0

Wort „zu“ vor dem Verb hinzu, was auf die Hinzufügung von „to“ vor den englischen Verben zurückzuführen ist. Daher haben wir das Wort „zu“ aus dem übersetzten Text entfernt, wenn es sich bei der syntaktischen Kategorie um ein Verb handelt.

Tab. 5.1 zeigt den Zustand von OdeNet, vor und nachdem der Algorithmus angewandt wurde. Es ist zu erkennen, dass es 13.818 Synsets mit eindeutigem ILI gab. Außerdem gab es 5965 Synsets mit doppelten ILI; das bedeutet, dass ein ILI mehr als einem Synset in OdeNet zugeordnet war. Die Gesamtzahl der eindeutigen doppelten ILI betrug 3703, was bedeutet, dass im Durchschnitt ein doppelter ILI 1,61 Synsets zugeordnet war.

Der auffälligste Unterschied nach Anwendung des Algorithmus ist die vollständige Eliminierung der doppelten ILI. Die Anzahl der Synsets mit eindeutigem ILI hat sich auf 19.547 erhöht und alle doppelten ILI wurden entfernt.

5.5.5 Zweiter Zyklus – Korrektur der syntaktischen Kategorien

Da die Präzision der syntaktischen Kategorien in OdeNet bei 93,3 % lag (mit Fehlern vor allem bei Mehrwortlexemen), gab es hier noch Potenzial für Verbesserungen. Mit den Übersetzungstabellen lagen nun weitere Informationen vor, mit denen die syntaktischen Kategorien korrigiert werden konnten.

Für jedes Synset in OdeNet wird das erste Lemma extrahiert. Dann werden alle Datensätze in der Übersetzungstabelle aufgerufen, bei denen das erste Lemma des Synset mit dem übersetzten Ziellemma übereinstimmt. Wenn die syntaktische Kategorie des Synset des Lemmas nicht mit einem der syntaktischen Kategorien der entsprechenden Datensätze in der Tabelle übereinstimmt, dann könnte eine Fehlklassifizierung im OdeNet-Synset vorliegen, denn es wäre vernünftig anzunehmen, dass die syntaktischen Kategorien des EWN-Synset auch die gleichen syntaktischen Kategorien in der Zielsprache haben sollten.

Dieser Algorithmus identifizierte 361 Synsets mit möglichen Fehlern in den syntaktischen Kategorien. Nach einer manuellen Auswertung hatten 325 dieser Synsets tatsächlich eine falsche syntaktische Kategorie. Dies bedeutet eine erfolgreiche Identifizierung von 90 % der Synsets mit falschen syntaktischen Kategorien. Von den 36 falsch-positiven Ergebnissen schlugen die meisten ein Adjektiv für ein Nomen oder ein Verb vor

und in vielen Fällen handelte es sich um Umgangssprache, wie zum Beispiel im Fall von odenet-19.938-n ['Tüftelei', 'Getüftel'].

Ein Test von zufällig ausgewählten Wörtern und ihren syntaktischen Kategorien zeigt, dass nun 97 % der lexikalischen Einträge korrekte syntaktische Kategorien haben. Idiome wie „ich hätte da mal 'ne Frage“ werden noch nicht alle korrekt klassifiziert.

5.5.6 Umgang mit Homonymie

Homonymie wird nach Löbner (2015, S. 53) folgendermaßen definiert:

„Zwei Lexeme sind totale Homonyme, wenn zwischen ihren Bedeutungen kein Zusammenhang besteht, sie aber sonst in allen konstitutiven Lexemeigenschaften übereinstimmen. Zwei Lexeme sind partielle Homonyme, wenn zwischen ihren Bedeutungen kein Zusammenhang besteht, sie aber in mindestens einer grammatischen Form übereinstimmen.“

Partielle Homonymie findet sich im Deutschen z. B. dann, wenn die unterschiedlichen Bedeutungen eines Wortes sich in den Pluralformen unterscheiden. Ein wohlbekanntes Beispiel dafür ist das Wort „Bank“ mit den Pluralformen „Banken“ und „Bänke“.

Unterschiedliche Bedeutungen eines Wortes können auch mit der unterschiedlichen Aussprache zusammenhängen, es kann sich um Homographe handeln. Tatsächlich finden sich in OdeNet einige Beispiele für Homographe, wie „Boot“ (in IPA¹² Notation [bu:t] oder [bo:t]). Eine andere Möglichkeit für partielle Homonymie sind Wörter mit unterschiedlichem Geschlecht, wie „die See“ oder „der See“.

Schließlich kommen auch in der deutschen Sprache Homonyme vor, die unterschiedliche syntaktische Kategorien haben, wie „erfahren“ (Verb und Adjektiv).

Im Fall von Homonymen handelt es sich nicht um einen Lexikoneintrag mit unterschiedlichen Bedeutungen (wie bei Polysemen), sondern um unterschiedliche Lexikoneinträge, die sich in der Schriftform zwar nicht unterscheiden, jedoch in anderen lexikalischen Eigenschaften.

In den Arbeiten zu Declerck et al. (2020) haben wir uns zunächst mit den Homographe beschäftigt. Wir haben Homographe mit ihrer phonetischen Information aus dem deutschen Teil von Wiktionary extrahiert, z. B.:

Boot

['bu:t'] ['über die Knöchel reichende Lederstiefel']

['bo:t'] [„kleines Wasserfahrzeug“, "Militär: seegehende Einheiten einer bestimmten Größenordnung bei der Marine"]

¹²IPA steht für „International Phonetic Alphabet“. Siehe auch <https://www.internationalphoneticassociation.org/content/ipa-chart>.

Diese haben wir dann geprüft und in OdeNet eingetragen:

```
<LexicalEntry id="w18234" confidenceScore="1.0">
  <Lemma writtenForm="Boot" partOfSpeech="n">
    <Pronunciation>bo:t</Pronunciation>
  </Lemma>
  <Sense id="w18234_4118-n" synset="odenet-4118-n"/>
</LexicalEntry>

<LexicalEntry id="w1202866" confidenceScore="1.0">
  <Lemma writtenForm="Boot" partOfSpeech="n">
    <Pronunciation>bu:t</Pronunciation>
  </Lemma>
  <Sense id="w1202866_9964-n" synset="odenet-9964-n"/>
</LexicalEntry>
```

Ebenfalls aus Wiktionary wurden die partiellen Homonyme mit unterschiedlichen Pluralformen, Geschlecht und syntaktischen Kategorien extrahiert. Eine sorgfältige und aufwendige manuelle Validierung der Einträge ist notwendig, um die Einträge in OdeNet aufzunehmen. Aktuell wurden 18 Lemmata doppelt als Homonyme eingetragen und mit Ausspracheinformation ergänzt. Davon haben 6 Homonyme unterschiedliche syntaktische Kategorien. Ein Problem gibt es mit dem grammatischen Geschlecht, denn dieses ist in der aktuellen DTD von Wordnet nicht vorgesehen. Aktuell müssen wir daher Einträge wie „Rasen“ (Maskulinum) und „Rasen“ (Neutrum) ohne weitere Information verdoppeln. Dasselbe gilt für Homonyme mit unterschiedlichen Pluralformen.

5.6 Anwendungen von OdeNet

Die Information, die OdeNet zur Verfügung stellt, lässt sich in vielfältige Anwendungen einbinden, die die deutsche Sprache benötigen. Hier stellen wir einige Beispiele vor.

Die Information über Synonymie lässt sich nutzen, um Menschen, die die deutsche Schriftsprache (noch) nicht gut verstehen, mit Informationen zu unterstützen. Wir haben in Wordnet Wörter aus dem Grundwortschatz der deutschen Sprache (die häufigsten Wörter des Deutschen)¹³ markiert. Anschließend haben wir ein Browser-Plug-in implementiert, das, wenn ein Synonym zu einem der Wörter aus dem Grundwortschatz auftritt, Information über das leichtere Synonym gibt.

Die Tatsache, dass OdeNet sehr viel Umgangssprache enthält, macht die Ressource wertvoll für Social-Media-Analysen. So kann man z. B. für die Sentiment-Analyse (siehe Siegel und Alexa 2020) in OdeNet 79 Synonyme zu „gut“ finden:

¹³<https://wortschatz.uni-leipzig.de/de>

'schon überredet', 'es sei', 'O. K.', 'warum nicht', 'das ist ein Wort', 'dein Wille geschehe', 'einverstanden', 'geht in Ordnung', 'alles klar', 'ist recht', 'abgemacht', 'ich nehme dich beim Wort', "so machen wir's", 'o. k.', 'geht klar', 'schön', 'so sei es', "d'accord", 'es geschehe nach deinen Worten', 'da sage ich nicht nein', 'genehmigt', 'machen wir', 'okay', 'gemacht', 'roger', 'akzeptiert', 'okey-dokey', 'gebongt', 'alles paletti', 'trefflich', 'gewinnbringend', 'lohnend', 'wertvoll', 'förderlich', 'oportun', 'von Vorteil', 'nutzbringend', 'fruchtbar', 'von Nutzen', 'sachdienlich', 'vorteilhaft', 'nützlich', 'nutzwertig', 'gedeihlich', 'günstig', 'dienlich', 'zutraglich', 'hilfreich', 'ersprießlich', 'schön', 'eins a', 'sympathisch', 'angenehm', 'vorteilhaft', 'günstig', 'ersprießlich', 'tadellos', 'positiv', 'charmant', 'ergötzlich', 'glücklich', 'fein', 'manierlich', 'herzerfrischend', 'wohltuend', 'gefremt', 'erbaulich', 'herzerquicklich', 'erhebend', 'erfreulich', 'pläsierlich', 'erquicklich', 'la', 'von guter Qualität', 'gütig', 'gutmütig', 'herzensgut', 'menschlich', 'Seele von Mensch'.

Dadurch, dass die Konzepte über mehrere Sprachen hinweg verlinkt sind, kann man dasselbe Konzept in Wordnets anderer Sprachen suchen und dann die Synonyme dieser Wordnets automatisch ins Deutsche übersetzen, um weitere Synonyme zu bekommen.

Die Meronymierelationen in OdeNet können genutzt werden, um – ebenfalls in der Sentiment-Analyse – Aspekte zu Entitäten zu erkennen. So will man z. B. nicht nur wissen, ob ein Nutzer ein Handy gut oder schlecht findet, sondern auch, ob es die Kamera oder der Lautsprecher ist, die/der gut oder schlecht funktioniert. Die Information über Hyponymie kann genutzt werden, um Unter- und Oberbegriffe zu finden (siehe auch Siegel 2018a).

Auch die Klassifikation von Hassmeldungen (Demus et al. 2022) kann von der reichhaltigen Information über Synonymie und der Umgangssprache in OdeNet profitieren. Zum Beispiel enthält OdeNet Synonyme zum Wort „erschießen“: 'wegpusten', 'abschießen', 'umlegen', 'abknallen', 'abballern', 'totschießen', 'umnieten', 'niederschließen', 'über den Haufen schießen', 'wegknallen'.

Der Aufbau von Terminologie für eine Domäne kann für die technische Dokumentation relevant sein (Siegel und Drewer 2012) oder auch für Informationsextraktion aus Texten (Siegel 2018b). OdeNet kann dazu dienen, Synonyme zu Benennungen zu finden und damit die Terminologie zu bereichern und die Informationsextraktion mächtiger zu machen.

Den Nutzen von Wordnet für medizinische Terminologie in Sprachen, die nicht Englisch sind, beschreiben Zotova et al. (2023). Die Konzeptverlinkung mit ILI über meh-

rere Sprachen kann dazu dienen, für die englische Sprache entwickelte Ontologien auch auf andere Sprachen anzuwenden.

5.7 Verfügbarkeit von OdeNet

OdeNet liegt – wie auch die anderen Wordnets in OMW – im XML-Format vor. Die Wordnets in OMW teilen sich eine gemeinsame DTD.¹⁴ Dadurch lassen sich u. a. mit Python recht einfach Zugriffe auf die Informationen definieren, die auf alle Wordnets übertragbar sind.¹⁵ Teil des OdeNet-Pakets sind Skripte mit Zugriffsfunktionen. Darüber hinaus gibt es ein Jupyter-Notebook, in dem Funktionen definiert sind, mit denen man Änderungen an OdeNet durchführen kann.

OdeNet wird unter einer Open-Source-Lizenz¹⁶ verteilt und ist über GitHub¹⁷ verfügbar.

Durch das konsequente Open-Source-Prinzip tragen regelmäßig Nutzer zu OdeNet bei. Es wird für Experimente in der Global Wordnet Community genutzt (wie bei Kühn et al. (2023) oder bei Lohk et al. (2023)).

5.8 Empfehlungen

Beim Aufbau von OdeNet haben wir wichtige Erfahrungen gemacht, die es uns nun erleichtern, wenn wir neue Wordnets für neue Sprache entwickeln wollen, wie aktuell das ukrainische Wordnet Ukrajinet. Die folgenden Empfehlungen sollen die Entwicklung von Wordnets unterstützen:

1. Für sprachtechnologische Anwendungen stehen Wordnets unter einer Open-Source-Lizenz im multilingualen Kontext zur Verfügung.
2. Es ist möglich, für eine neue Sprache mit semiautomatischen Methoden ein Wordnet zu entwickeln, indem Methoden der Sprachtechnologie verwendet werden.
3. Ein neu aufgebautes Wordnet muss (noch) nicht perfekt sein, um in Anwendungen einsetzbar zu sein.
4. Ein konsequenter Open-Source-Ansatz führt dazu, dass interessierte Personen relevante Arbeit leisten und Wordnets erweitern helfen.

¹⁴<https://globalwordnet.github.io/schemas/WN-LMF-1.1.dtd>

¹⁵Ein Beispiel für so einen Zugriff ist <https://dict.passivlingo.com>.

¹⁶<https://creativecommons.org/licenses/by-sa/4.0/>

¹⁷<https://github.com/hdaSprachtechnologie/odenet>

5. Die Einbindung der Arbeiten in einen multilingualen Forschungskontext erlaubt es, Information aus Wordnets anderer Sprachen zu nutzen und zu übertragen.
6. Gleichzeitig macht die multilinguale Zugriffsmöglichkeit auf Konzepte interessante Forschungsergebnisse und Anwendungen möglich.

5.9 Fazit

In diesem Kapitel wurde OdeNet vorgestellt, das offene deutsche Wordnet. OdeNet ist mit dem Merge Approach automatisch aus dem OpenThesaurus abgeleitet und mithilfe automatischer Übersetzung im Open Multilingual Wordnet integriert. Nach der automatischen Erstellung wurden Korrekturen, Anpassungen und Erweiterungen durchgeführt:

- Die Zuweisung der syntaktischen Kategorien wurde korrigiert, zum Teil vollautomatisch mit Information über Wortendungen und zum Teil halbautomatisch auf Basis automatischer Analysen von OdeNet.
- Relationen wurden automatisch hinzugefügt, basierend auf Kompositaanalysen und auf Antonymlisten.
- Die Verknüpfung von OdeNet zum Open Multilingual Wordnet wurde mit neuen Ideen zur automatischen Übersetzung verbessert.
- Manuell wurden Informationen zu Homonymen eingetragen, die aus automatischen Analysen abgeleitet wurden.

Die Arbeiten an OdeNet haben gezeigt, dass es möglich ist, mit Verfahren der Sprachtechnologie, wenigen Ressourcen und in kurzer Zeit ein Wordnet aufzubauen. Dieses kann kontinuierlich verbessert werden, ist aber schon von Anfang an eine verwendbare Ressource. Kühn et al. (2023) beschreiben zum Beispiel, wie sie die Antonymrelationen in OdeNet verwenden.

Der Open-Source-Ansatz trägt dazu bei, dass auch andere Personen aus der Wissenschaft OdeNet verwenden und Fehlerkorrekturen und Erweiterungen beitragen. Die Einbindung in den multilingualen Entwicklungskontext erlaubt Übertragung von Informationen aus den Wordnets anderer Sprachen (wie bei Lohk et al. (2023)).

OdeNet wird kontinuierlich weiterentwickelt. Die Ergebnisse der Arbeiten von Kühn et al. (2023) und Lohk et al. (2023) werden zu neuen Relationen (Hyponymie und Antonymie) führen.

Mit den Erfahrungen aus der Entwicklung von OdeNet bauen wir aktuell ein Wordnet für die ukrainische Sprache auf. Erste Ergebnisse dazu sind in Siegel et al. (2023) beschrieben.

Literatur

- Alexeyevsky, D. & Temchenko, A. V. (2016), WSD in monolingual dictionaries for Russian WordNet, in 'Proceedings of the Eighth Global WordNet Conference', Bucharest, Romania.
- Bakay, Ö., Ergelen, Ö., Sarmış , E., Yıldırım, S., Arıcan, B. N., Kocabalcıoğlu, A., Özçelik, M., Sanyar, E., Kuyrukçu, O., Avar, B., Yıldız, O. T. (2021), Turkish wordnet KeNet, in 'Proceedings of the 11th global wordnet conference', pp. 166–174.
- Bergh, J. & Siegel, M. (2023), Connecting multilingual wordnets: Strategies for improving ILI classification in OdeNet, in 'Proceedings of the Global Wordnet Conference', Donostia, Spain.
- Bond, F. & Foster, R. (2013), Linking and extending an open multilingual wordnet, in 'Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics', Sofia, pp. 1352–1362. URL: <http://aclweb.org/anthology/P13-1133>.
- Bond, F. & Paik, K. (2012), 'A survey of wordnets and their licenses', *Small* 8(4), 5.
- Bond, F., Da Costa, L. M. & Le, T. A. (2015), 'Imi—a multilingual semantic annotation environment', *Proceedings of ACL-IJCNLP 2015 System Demonstrations* pp. 7–12.
- Bond, F., Vossen, P., McCrae, J. P. & Fellbaum, C. (2016), CILI: The Collaborative Interlingual Index, in 'Proceedings of the Global WordNet Conference', Vol. 2016.
- Brants, S., Dipper, S., Eisenberg, P., Hansen-Schirra, S., K'önig, E., Lezius, W., Rohrer, C., Smith, G. & Uszkoreit, H. (2004), 'TIGER: Linguistic interpretation of a German corpus', *Research on language and computation* 2(4), 597–620.
- Braslavski, P., Ustalov, D., Mukhin, M. & Kiselev, Y. (2016), Yarn: Spinninginprogress, in 'Proceedings of the 8th Global WordNet Conference, GWC 2016', pp. 58–65.
- Chiruzzo, L., **Agüero-Torales**, M., Alvarez, A. & **Rodríguez**, Y. (2023), Initial experiments for building a Guarani wordnet, in '12th International Global Wordnet Conference'.
- Declerck, T., Bajcetic, L. & Siegel, M. (2020), Adding pronunciation information to wordnets, in 'Proceedings of the LREC 2020 Workshop on Multimodal Wordnets (MMW2020)', pp. 39–44.
- Demus, C., Pitz, J., Schütz, M., Probol, N., Siegel, M. & Labudde, D. (2022), Detox: A comprehensive dataset for German offensive language and conversation analysis, in 'Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)', Seattle, Washington.
- Derwojedowa, M., Piasecki, M., Szpakowicz, S., Zawislawska, M. & Broda, B. (2008), 'Words, concepts and relations in the construction of Polish wordnet', *Proceedings of GWC 2008* pp. 162–177.
- Fellbaum, C., ed. (1998), *WordNet: An Electronic Lexical Database*, MIT Press, Cambridge, MA.
- Fellbaum, C. & Vossen, P. (2008), Challenges for a global Wordnet, in 'Online Proceedings of the First International Workshop on Global Interoperability for Language Resources', pp. 75–82.
- Fjeld, R. V. & Nygaard, L. (2009), NorNet—a monolingual Wordnet of modern Norwegian, in 'NO-DALIDA 2009 workshop: WordNets and other Lexical Semantic Resources-between Lexical Semantics, Lexicography, Terminology and Formal Ontologies', Vol. 7, pp. 13–16.
- Goodman, M. W. & Bond, F. (2021), Intrinsically interlingual: The Wn Python library for wordnets, in '11th International Global Wordnet Conference (GWC2021)'.
- Hamp, B. & Feldweg, H. et al. (1997), GermaNet—a lexical-semantic net for German, in 'Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications', pp. 9–15.
- Isahara, H. Bond, F., Uchimoto, K., Utiyama, M., & Kanzaki, K. (2008), Development of the japanese wordnet., in 'LREC'.
- Kühn, R., Mitrovic, J. & Granitzer, M. (2023), Hidden in plain sight: Can German Wiktionary and wordnets facilitate the detection of antithesis?, in '12th International Global Wordnet Conference'.

- Kunze, C. & Lemnitzer, L. (2010), 'Lexical-semantic and conceptual relations in GermaNet', *Lexical-semantic relations: Theoretical and practical perspectives* pp. 163–183.
- Lindén, K. & Carlson, L. (2010), 'Finnwordnet – Finnish wordnet by translation', *LexicoNordica – Nordic Journal of Lexicography* 17, 119–140.
- Löbner, S. (2015), *Semantik*, de Gruyter.
- Lohk, A., Rebane, M. & Orav, H. (2023), An experiment: Finding parents for parentless synsets by means of CILI, in '12th International Global Wordnet Conference'.
- McCrae, J. P., Rademaker, A., Bond, F., Rudnicka, E. & Fellbaum, C. (2019), English wordnet 2019—an open-source wordnet for english, in 'Proceedings of the 10th Global WordNet Conference', pp. 245–252.
- McCrae, J. P., Rademaker, A., Rudnicka, E. & Bond, F. (2020), English wordnet 2020: Improving and extending a wordnet for english using an open-source methodology, in 'proceedings of the LREC 2020 workshop on multimodal WordNets (MMW2020)', pp. 14–19.
- McCrae, J. P., Goodman, M. W., Bond, F., Rademaker, A., Rudnicka, E. & Da Costa, L. M. (2021), The GlobalWordNet formats: Updates for 2020, in 'Proceedings of the 11th Global Wordnet Conference', pp. 91–99.
- Naber, D. (2004), 'Openthesaurus: Building a thesaurus with a web community', Retrieved January 3, 2005. URL: <https://www.openthesaurus.de/download/openthesaurus.pdf>.
- Paikens, P., Klints, A., Lokmane, I., **Pretkalniņa**, L., Rituma, L., **Stāde**, M. & Strankale, L. (2023), Latvian wordnet, in '12th International Global Wordnet Conference'.
- Pedersen, B. S., Nimb, S., Asmussen, J., Sørensen, N. H., Trap-Jensen, L. & Lorentzen, H. (2009), 'DanNet: the challenge of compiling a wordnet for Danish by reusing a monolingual dictionary', *Language resources and evaluation*, 43, 269–299.
- Sagot, B. & Fišer, D. (2008), Building a free French wordnet from multilingual resources, in 'On-toLex'.
- Siegel, M. (2018a), The role of ontologies in sentiment analysis, in A. Reibold, T. Hoppe & B. Humm, ed., 'Semantic Applications. Methodology, Technology, Corporate Use', Springer-Verlag, Berlin, Heidelberg.
- Siegel, M. (2018b), Text mining in economics, in A. Reibold, T. Hoppe & B. Humm, ed., 'Semantic Applications. Methodology, Technology, Corporate Use', Springer-Verlag, Berlin, Heidelberg.
- Siegel, M. & Alexa, M. (2020), *Sentiment-Analyse deutschsprachiger Meinungsäußerungen*, Springer, Wiesbaden.
- Siegel, M. & Bond, F. (2021), Compiling a German wordnet from other resources, in '11th International Global Wordnet Conference (GWC2021)'.
- Siegel, M. & Drewer, P. (2012), Terminologieextraktion – multilingual, semantisch und mehrfach verwendbar., in 'Tagungsband der TEKOM-Frühjahrstagung'.
- Siegel, M., Vakulenko, M. & Baum, J. (2023), Towards Ukrainian wordnet: Incorporation of an existing thesaurus in the domain of physics, in 'Proceedings of KONVENS', Ingolstadt. to appear.
- Van Harmelen, F. & Ten Teije, A. (2019), 'A boxology of design patterns for hybrid learning and reasoning systems', *Journal of Web Engineering* 18(1–3), 97–123.
- Vossen, P., ed. (1998), *Euro WordNet*, Kluwer.
- Witschel, H. F., Pande, C., Martin, A., Laurenzi, E. & Hinkelmann, K. (2020), Visualization of patterns for hybrid learning and reasoning with human involvement, in 'New Trends in Business Information Systems and Technology: Digital Innovation and Digital Business Transformation', Springer, pp. 193–204.
- Zotova, E., Cuadros, M. & Rigau, G. (2023), Towards the integration of WordNet into ClinIDMap, in '12th International Global Wordnet Conference'.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.




Teil II

Wissensbasierte KI für Machine Learning



Integration von terminologischen Wissen in eine Word-Embedding-basierende Semantische Suche

6

Thomas Hoppe 

Kernaussagen

1. Word-Embeddings stellen eine aus Dokumenttexten berechnete, numerische Vektor-Repräsentation von Wortbedeutungen dar.
2. Prinzipiell kann eine semantische Suche mithilfe von Word-Embeddings zwar realisiert werden, mit ihnen kann jedoch ohne eine weitere Aufbereitung der Dokumentensuchraum nicht sinnvoll eingeschränkt werden.
3. Word-Embeddings können durch Clustering-Verfahren zu sogenannten SimSets zusammengefasst werden, mit denen – ähnlich wie mit Synsets oder Synonymen – die Suchergebnisse eingeschränkt werden können.
4. Je nach eingesetztem Clustering-Verfahren ist eine distanz-gewichtete Kosinus-Ähnlichkeit für Word-Embeddings vorzuziehen, da eine einfache Kosinus-Ähnlichkeit lediglich die Richtungs-Ähnlichkeit, nicht aber die Distanz-Ähnlichkeit berücksichtigt.
5. Die Integration von Hintergrundwissen in Word-Embeddings kann durch die Ableitung von künstlichen Sätzen, die den Kontext von Konzepten beschreiben, erreicht werden.
6. Der Vergleich von Synonymen eines Thesaurus mit aus dem Thesaurus abgeleiteten SimSets zeigt, dass ein Großteil der Synonyme exakt oder näherungsweise rekonstruiert werden kann.
7. Dieses Rekonstruktionsexperiment zeigt, dass die aus Word-Embeddings durch Clustering abgeleiteten SimSets in der Tat einen Teil der Semantik von Begriffen erfassen.

T. Hoppe (✉)
Fraunhofer-Institut Fokus, Berlin, Deutschland
E-Mail: thomas.hoppe@fokus.fraunhofer.de

6.1 Einleitung

Suchverfahren gehören zu den Verfahren des Information Retrievals. Konventionelle semantische Suchverfahren basieren auf der Anwendung von symbolisch-repräsentiertem, terminologischen Wissen zur Verbesserung der Suchergebnisse (Hoppe 2020). Hierzu werden in der Regel Oberbegriffe, Unterbegriffe, Synonyme oder verwandte Begriffe herangezogen, entweder um die Suchanfragen zu erweitern oder um die Dokumente mit Metadaten zu annotieren. Dieses terminologische Wissen wird in der Regel in Synonymlisten, Taxonomien, Thesauri, Wortnetze (Wordnet), Ontologien oder Wissensgraphen (Knowledge Graph) repräsentiert. Die Qualität der Suchergebnisse hängt bei diesen Verfahren von der Qualität dieses Wissens und von der Fähigkeit des Verfahrens ab, mehrdeutige Begriffe zu disambiguieren.

Ein wesentlicher Nachteil dieser Klasse semantischer Suchverfahren stellt jedoch die Verfügbarkeit des terminologischen Wissens dar.

- Für allgemeine Suchanwendungen, wie z. B. Suche von Dokumenten im Internet oder aktuelle Nachrichten, reicht es in der Regel aus, Quellen mit allgemeinem Weltwissen, wie die Wissensgraphen von DBpedia oder Wikidata, zu nutzen, die viele Themenbereiche erfassen, aber nur punktuell in der Tiefe abdecken.
- Für einige Fachgebiete, die von großem wirtschaftlichen oder gesellschaftlichen Interesse sind, wie Medizin, Chemie, Finanzwesen, Jura usw., existiert hingegen umfangreiches, anwendungsbereichs-spezifisches Wissen, in Form von Taxonomien, Thesauri und Ontologien, welches auf den jeweiligen Themenbereich eingeschränkt ist und diesen in der Tiefe sehr detailliert abdeckt.
- Spezialisierte Themenbereiche, die zwar eine tiefe Abdeckung erfordern, die aber nur für bestimmte, mitunter kleine Gruppen von Anwendenden von Interesse sind, stellen eine dritte Klasse dar. Für diese Themenbereiche wurde das erforderliche terminologische Wissen in der Regel noch nicht modelliert, da der Aufwand hierfür zu groß erschien.

Zur letzten Klasse zählen z. B. Stellen- und Jobsuche oder die berufliche Bildung und Weiterbildung. Für diese Bereiche existieren zwar einige Klassifikationsschemata, wie z. B. *International Standard Classification of Occupations (ISCO)*¹, *European Skills, Competences, Qualifications and Occupations (ESCO)*² oder die *Klassifikation der Berufe (KldB)*³, die Kompetenzen, Fähigkeiten und Berufsbezeichnungen standardisieren.

¹ <https://warwick.ac.uk/fac/soc/ier/researchthemesoverview/researchprojects/classification/isco88/german> (letzter Zugriff 7.9.2023)

² <https://esco.ec.europa.eu/de/node/4> (letzter Zugriff 7.9.2023)

³ <https://statistik.arbeitsagentur.de/DE/Navigation/Grundlagen/Klassifikationen/Klassifikation-der-Berufe/KldB2010-Fassung2020/KldB2010-Fassung2020-Nav.html> (letzter Zugriff 7.9.2023)

Viele Anbieter von Stellen- und Weiterbildungsangeboten halten sich aber nicht unbedingt an diese Standards und verwenden für die Beschreibung ihrer Angebote ihren eigenen Sprachgebrauch.

Die alleinige Fokussierung auf existierende Klassifikationssysteme ist für diese Anwendungsbereiche nicht ausreichend, vielmehr muss die Verwendung der Begriffe im jeweiligen Kontext berücksichtigt werden.

Diese Bereiche sind durch eine hohe Dynamik gekennzeichnet: alte Bezeichnungen verschwinden in den Angeboten, z. B. in der DDR verwendete Berufsbezeichnungen, neue kommen durch die sich weiterentwickelnde Berufswelt hinzu, beispielsweise Web Designer, Data Scientists, Home Office, Work-Life-Balance, Transformermodelle, LLM, Quantencomputer, etc. Darüber hinaus werden unterschiedliche Bezeichnungen quasi synonym gebraucht, wie z. B. Sales Director, Vertriebsleiter, Direktor Verkauf, Verkaufsleiter, etc. (Hoppe 2015).

Die Fragestellung: „Wie kann für einen spezialisierten Anwendungsbereich, für den noch kein terminologisches Wissen verfügbar ist, allein anhand der Dokumentinhalte eine intelligentere Suchfunktion realisiert werden, die ohne explizit formalisiertes Wissen auskommt?“, führte zur Entwicklung eines vom Fraunhofer-Institut FOKUS zum Patent angemeldeten Verfahrens⁴ des Autors.

Dieses Kapitel (siehe Einordnung in Abb. 6.1) stellt eine neue Form semantischer Suchverfahren am Beispiel von Stellenbörsen und Weiterbildungsangeboten vor, deren grundlegendes Prinzip auf dem Clustering von Word-Embeddings beruht. Die Cluster dienen zur Gruppierung von ähnlichen Begriffen allein auf der Basis der Dokumenttexte. Sie können zur Umsetzung semantischer Suchverfahren, die initial ohne Modellierungen der Begriffswelt des Anwendungsbereichs auskommen, genutzt werden. Darüberhinaus

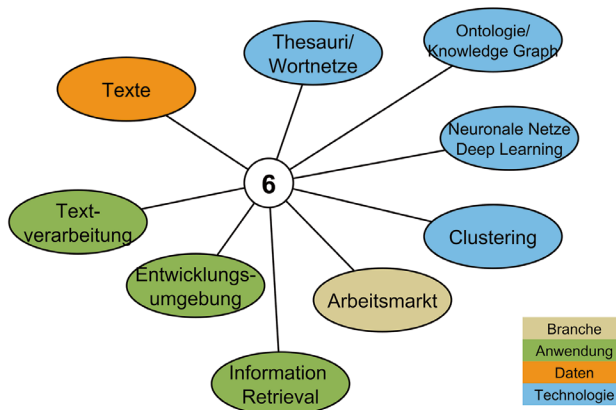


Abb. 6.1 Thematische Einordnung des Kap. 6

⁴DE102019212421, EP2020073304, CA000003151834, US020220292123, WO002021032824

können sie Wissensingenieure bei der manuellen Modellierung von Wissensgebieten unterstützen. Es wird weiter gezeigt, wie Word-Embeddings durch formalisiertes Wissen ergänzt werden können.

6.1.1 Anwendungsfall

Im Bereich der Suche in Jobportalen und Weiterbildungsdatenbanken wurde zwischen 2010 und 2015 ein Ansatz zur symbolischen, wissensbasierten semantischen Suche durch das Unternehmen Ontonym entwickelt⁵. Auch wenn sich der Aufwand zur Modellierung des benötigten terminologischen Wissens – entgegen der allgemeinen Auffassung – bei einer pragmatischen Vorgehensweise in Grenzen hält, so muss doch vor Beginn der Entwicklung eine Hürde überwunden werden: Um den Nutzen einer intelligenten Suchfunktion erfahrbar zu machen, muss ein initialer Aufwand für die Modellierung des terminologischen Wissens in Kauf genommen werden.

Im Rahmen des vom *Bundesministeriums für Bildung und Forschung* und des *Bundesinstituts für Berufsbildung* geförderten Projekts „Expand+er WB³⁺⁶“ wird eine neue hybride semantische Suchfunktion für berufliche Weiterbildungsangebote auf der Basis des zum Patent angemeldeten Verfahrens prototypisch entwickelt. Dieses Verfahren setzt auf dem in den Weiterbildungsangeboten verwendeten Sprachgebrauch auf und ermittelt mit Hilfe subsymbolischer Word-Embeddings eine symbolische Repräsentation von verwandten Begriffen, die für Anfrageerweiterungen, zur Indexierung oder zur Annotation genutzt werden können.

Für die Entwicklung im Rahmen des Expand+er WB³ Projekts werden rd. 46.000 Weiterbildungsangebote der Weiterbildungsdatenbank Berlin-Brandenburg genutzt, die größtenteils deutschsprachig sind, aber auch Weiterbildungsangebote auf Spanisch, Italienisch, Französisch und Schwedisch umfassen.⁷

6.1.2 Sprachgebrauch

Für die Interpretation von Suchanfragen und deren Verarbeitung mittels Word-Embeddings ist der Sprachgebrauch, i.E. die Verwendung von Worten im Kontext, eine wesentliche Grundlage.

⁵Teilweise dokumentiert in (Hoppe und Junghans 2015), (Hoppe und Tolksdorf 2018) und (Hoppe 2020).

⁶https://www.fokus.fraunhofer.de/en/projekt/viscom/expander_2021-12, gefördert durch das BMBF und BIBB unter Förderkennzeichen 21INVI3106.

⁷Aus rechtlichen Gründen erfolgt die Darstellung dieses Verfahrens in diesem Kapitel auf einem anderen Korpus von rd. 75.000 Stellenanzeigen für Ingenieure.

Ludwig Wittgenstein hat in seinen Philosophischen Untersuchung (Wittgenstein 1977) den Begriff des *Sprachspiels* eingeführt, um den von Lebensformen⁸ gelebten Gebrauch normaler Sprache zu beschreiben⁹. Er weist dabei darauf hin, dass die Bedeutung von Worten nicht vorgegeben ist, sondern erst durch deren Gebrauch festgelegt wird. Im Bereich der Linguistik und der geschriebenen Sprache drückte dies John Rupert Firth durch „You Shall Know a Word by the Company It Keeps“ aus (Firth 1957).

Konsequent angewandt auf die Entwicklung intelligenter Systeme bedeutet dies, dass sowohl der gelebte Sprachgebrauch der Nutzenden und der Publizierenden, ebenso wie der in normativen Standards definierte Sprachgebrauch zu berücksichtigen sind.

6.1.3 Übersicht über das Word-Embedding-basierende semantische Suchverfahren

Mit Boxology haben van Harmelen und ten Teije (2019) und Witschel et al. (2021), wie im Kap. 1 dargestellt, einen Ansatz zur vereinheitlichten Beschreibung von maschinellen Lernverfahren und wissensbasierten Systemen vorgestellt. Die neue semantische Suche kann als Boxology-Grafik, wie in Abb. 6.2. dargestellt, skizziert werden.

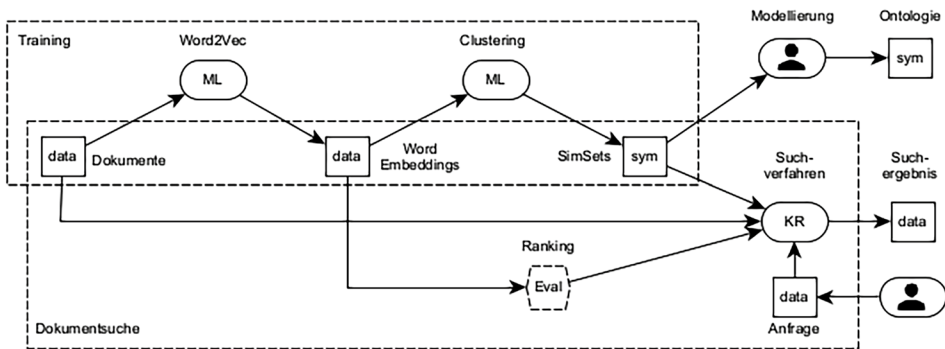


Abb. 6.2 Boxology-Modell des hybriden KI-Systems

⁸Wittgenstein verwendet diesen Begriff mehrdeutig und bezeichnet damit sowohl unterschiedliche Lebensformen im engeren Sinn (ggf. mit ihren eigenen Sprachen), Gruppen von Handelnden, als auch deren Handlungskontexte.

⁹Dirschel & Harz (Kap. 15) verweisen darauf, „dass die Verwendung von Wörtern im juristischen Bereich sich teilweise von derselben Verwendung der Wörter im alltäglichen Leben unterscheiden kann“. Im Sinne von (Wittgenstein 1977) benutzt die „Lebensform Jurist“ ein anderes – spezifischeres – „Sprachspiel“ als wir es in unserem alltäglichen Leben mit dem „Sprachspiel Umgangssprache“. Im Kontext digitaler Anwendungen nimmt die Vermittlung zwischen diesen unterschiedlichen Sprachwelten eine wichtige Rolle ein, um Fehlinterpretationen zu vermeiden.

Innerhalb der Trainingsphase werden Word-Embeddings aus Dokumenten berechnet. Mithilfe eines Clustering-Verfahrens werden *SimSets*, als Mengen von abgrenzbaren Begriffen mit ähnlicher Bedeutung, berechnet. Diese Simsets dienen der eigentlichen Suchfunktion als Entscheidungskriterium, um aus einem invertierten Index die Dokumente zu ermitteln, die mindestens einen der SimSet-Begriffe enthalten. Unter der Maßgabe, dass es sich hierbei um die abgrenzbar, ähnlichsten Begriffe handelt, die für die Suche als relevant betrachtet werden, kann die Menge aller Dokumente auf die relevantesten begrenzt werden. Diese Dokumente können wiederum mithilfe der Word-Embeddings in eine Rangfolge sortiert werden, die die Ähnlichkeit zur Suchanfrage widerspiegelt. Darüber hinaus können die SimSets als Ausgangspunkt und Hilfsmittel für eine spätere manuelle/intellektuelle Modellierung des jeweiligen Anwendungsbereichs genutzt werden.

6.2 Technologischer Hintergrund

Konventionelle und semantische Volltext-Suchmaschinen basieren auf der sogenannten Bag-of-Words-Repräsentation von Dokumenten. Dokumente werden hierbei als eine ungeordnete Menge von unabhängigen Termen betrachtet, die in einem Dokument vorkommen.

6.2.1 Bag-of-Words

Eine Bag-of-Words-Repräsentation kann man sich als einen extrem großen Vektor vorstellen. Für jeden Term des in den Dokumenten verwendeten Vokabulars umfasst dieser Vektor eine Komponente. Die Vektoren haben somit eine Größe, die dem Umfang des Vokabulars entspricht. Oder anders ausgedrückt, jeder Term repräsentiert eine Dimension eines sehr hochdimensionalen Vektorraums, dessen Dimensionalität der Anzahl der Terme des Vokabulars entspricht.

Mit dieser Modellvorstellung wird jedes Dokument als Punkt in einem Vektorraum repräsentiert. Vektoren, die die Dokumente repräsentieren, sind nur sehr dünn („sparse“) besetzt, da es kein einziges Dokument geben wird, welches einen Großteil aller oder alle Terme beinhaltet. Bei einer ungünstigen Repräsentation würde so enorm viel Speicherplatz verschwendet werden. Dieser Nachteil lässt sich zwar durch effizientere Vektor-Repräsentationen beheben, problematischer ist aber, dass die einzelnen Vektor-Komponenten als voneinander unabhängig angenommen werden. Zudem werden Begriffe, die inhaltlich verwandt sind, auf unterschiedliche, unabhängige Dimensionen abgebildet (Hoppe 2020).

Solche hochdimensionalen Räume kann man sich weder vorstellen noch veranschaulichen. Die einzelnen Terme werden in diesen Räumen durch orthogonale Dimensionen repräsentiert. Dokumente bestehen aus Linearkombinationen der die Terme repräsen-

tierenden Orthogonalbasis und liegen in dem Vektorraum nur auf den Hyper-Flächen, -Kanten und -Ecken.

Die Häufigkeit des Vorkommens der einzelnen Terme in den Dokumenten kann hierbei nur positive Werte aus dem Bereich der natürlichen Zahlen annehmen. Hieraus folgt, dass alle Dokumente nur in den nicht-negativen Orthanten¹⁰ liegen können.

6.2.2 Word-Embeddings

Eine kompaktere Repräsentation für Terme bilden sogenannte Word-Embeddings. Hierbei werden Terme in einen niedrig-dimensionalen, reell-wertigen Vektorraum eingebettet, der in der Regel lediglich aus mehreren hundert Dimensionen gebildet wird. In diesem Raum sind die Vektoren dicht besetzt, d. h. alle Komponenten werden genutzt und die Terme werden daher in diesen Vektorräumen auch in die Hyper-Räume verteilt. Da für Word-Embeddings reellwertige Vektoren verwendet werden, bevölkern die Vektoren alle Orthanten.

Word-Embedding-Verfahren können unterschieden werden in single-sense und multi-sense Word-Embedding-Verfahren. Zu den single-sense Word-Embedding-Verfahren zählen Word2Vec (Mikolov et al. 2013), GloVe (Pennington et al. 2014), FastText (Bojanowski et al. 2016), Gauss2Vec (Vilnis und MacCallum 2014), die Homonyme auf einen einzigen dicht besetzten Vektor abbilden.¹¹

Kontextualisierte Word-Embedding-Verfahren, wie Bayesian SkipGram (Bražinskas et al. 2018)¹², ElMo-Embeddings (Peters et al. 2018) oder Flair-Embeddings (Akbik et al. 2018)¹³ können die unterschiedlichen Bedeutungen von Homonymen auf mehrere Word-Embeddings abbilden. In diesem Sinn handelt es sich bei ihnen um multi-sense Word-Embeddings.

Das Interessante an dieser Repräsentationsform ist es, dass die Lage der einzelnen Term-Vektoren durch ein unüberwachtes Lernverfahren mithilfe eines künstlichen neuronalen Netzes (KNN) mit einer Autoencoder-, Variational-Autoencoder- oder LSTM-Architektur ermittelt wird. Hierzu werden die einzelnen Terme zunächst zufällig im

¹⁰Orthanten <https://www.spektrum.de/lexikon/mathematik/orthant/7578> (Letzter Zugriff 5.9.2023) bezeichnen die n-dimensionale Verallgemeinerung der 2D-Quadranten. Sie werden auch als d-dimensionaler Hyperoctant bezeichnet (Weisstein, Eric W. „Hyperoctant.“ From *MathWorld*-A Wolfram Web Resource., <https://mathworld.wolfram.com/Hyperoctant.html> (Letzter Zugriff 15.7.2023).

¹¹Das Large Language Model BERT (Devlin et al. 2018) kombiniert solche Word-Embeddings mit Sentence-Embeddings, um zusätzlich noch unterschiedliche Verwendungen des gleichen Terms repräsentieren zu können.

¹²Die Referenz-Implementierung basiert auf einem Variational Auto-Encoder, ist jedoch nur unter Python 2 verfügbar. Eine Portierung nach Python 3 ist nur mit veralteten Bibliotheken lauffähig.

¹³https://github.com/flairNLP/flair/blob/master/resources/docs/embeddings/FLAIR_EMBEDDINGS.md (Letzter Zugriff 15.7.2023).

Vektorraum verteilt. Durch das Training des neuronalen Netzes werden die Positionen der Term-Vektoren anhand von Kontextinformationen der Terme justiert, sodass im Laufe des Trainings Terme, die in gleichen oder sehr ähnlichen Kontexten verwendet werden, in enge räumliche Nähe rücken.¹⁴

Der Kontext wird hierbei aus dem Satzzusammenhang gebildet, in dem die Terme in den Dokumenten auftreten. Hierzu wird ein gleitendes Fenster (sliding window) über die in den Dokumenten auftretenden Sätze geschoben. Jede Verschiebung des Fensters erzeugt ein positives Trainingsbeispiel. Die Fenster haben hierbei in der Regel eine ungerade Länge (z. B. 5, 7 oder 9 Worte). Je nach Ausgestaltung bildet das zentrale Wort das Wort, welches aus den anderen Worten, den Kontextworten, vorherzusagen ist (die sogenannte CBOW-Variante¹⁵), bzw. für das die Kontextworte vorherzusagen sind (die sogenannte SkipGram-Variante).

Aus den Word-Embeddings der Terme eines Dokuments können Dokumente repräsentierende Document-Embeddings als Linearkombination der Word-Embeddings berechnet werden. Diese Document-Embeddings können zwar ebenfalls den Vektorraum der Word-Embeddings bevölkern, genau genommen aber bevölkern sie einen eigenen Vektorraum, der in den Word-Embedding-Raum abbildbar ist.

6.2.3 Wortähnlichkeiten

Die Ähnlichkeit von Vektoren wird in der Regel mithilfe der Kosinus-Ähnlichkeit über den Winkel zwischen den Vektoren ermittelt, bei der die Vektoren auf Einheitslänge normiert werden. Kosinus-Ähnlichkeiten nahe der 1 bedeuten eine große Ähnlichkeit (ein kleiner Winkel), Werte nahe der 0 deuten auf die Unabhängigkeit hin (rechter Winkel), während Werte nahe der -1 diametral entgegengerichtete Vektoren anzeigen (Winkel von 180°).¹⁶

Die Kosinus-Ähnlichkeit ermittelt die Ähnlichkeit der Richtung beider Vektoren, berücksichtigt aber nicht die räumliche (euklidische) Distanz der Vektorendpunkte. So können zwei Vektoren zwar in nahezu der gleichen Richtung ausgerichtet sein, aber dennoch unterschiedliche Längen besitzen. Dies bedeutet, dass allein aus der

¹⁴Dies erlaubt manchmal sehr spektakuläre Rechenoperationen mit den Vektoren auf den Worten, wie z. B.: König – Mann + Frau = Königin, die jedoch in der Regel nur punktuell in Abhängigkeit von den zum Training verwendeten Texten möglich werden.

¹⁵CBOW steht für ‚continuous bag of words‘.

¹⁶Für die Bag-Of-Words Repräsentation kann die Kosinus-Ähnlichkeit nur Werte zwischen 0 und 1 annehmen, da sich alle Vektoren in nicht-negativen Orthanten befinden. Für reellwertige Vektoren der Word-Embeddings kann die Kosinus-Ähnlichkeit auch negative Werte bis zu -1 annehmen.

Richtungs-Ähnlichkeit noch nicht darauf geschlossen werden kann, dass die Vektoren auch Punkte in räumlicher Nähe repräsentieren.

6.2.4 Einfache Suchfunktion mit Word-Embeddings

ElasticSearch bietet seit Version 7.3 die Möglichkeit, Einbettungen (Embeddings) zu verwenden, um Dokumente nach ihrer Ähnlichkeit zur Anfrage mithilfe der Kosinus-Ähnlichkeit zu sortieren. Hierbei werden bei der Indexierung der Dokumente und der Anfrage Texteinbettungen durch ein Satzeinbettungsverfahren berechnet. Hierbei wird die Vektor-Ähnlichkeit aber lediglich für das Ranking genutzt und nicht zur Ermittlung der passenden Dokumente.¹⁷

Für ElasticSearch in der Version 8.0 werden für Suchanfragen zwar auch Sentence-Embeddings berechnet, diese werden jedoch nur für eine effiziente approximative K-Nearest-Neighbour-Suche verwendet, bei der lediglich nach einer vorgegebenen Anzahl von Dokumenten gesucht wird.¹⁸ Letzteres ist eine pragmatische Einschränkung, um nicht alle Dokumente mit der Suchanfrage vergleichen zu müssen. Diese Einschränkung hat jedoch zwei prinzipielle Nachteile:

1. Die Anzahl der Ergebnisse wird willkürlich auf K – entsprechend der Kosinus-Ähnlichkeit – ähnliche Dokumente beschränkt. Weitere potenziell relevante Dokumente werden u. U. nicht gefunden.
2. Unter Umständen werden auch irrelevante Dokumente, deren Ähnlichkeit geringer ist, zurückgeliefert, da kein Kriterium verwendet wird, mit dem die Ähnlichsten von den nicht so ähnlichen Dokumenten abgegrenzt werden können.

Mit dieser pragmatischen Lösung kann vermieden werden, dass für jedes Anfrage-Embedding immer die Kosinus-Ähnlichkeit zu jedem Document-Embedding berechnet werden müsste, da jede Anfrage jeweils zu allen anderen Dokumenten ähnlich ist, jedoch zu unterschiedlichen Graden.¹⁹

Einer solchen naiven Verwendung der Kosinus-Ähnlichkeit für Embeddings liegt das Problem zugrunde, dass es solchen Verfahren an einem effektiven Filterkriterium mangelt, um die wirklich relevanten von den vermeintlich relevanten Dokumente

¹⁷<https://www.elastic.co/de/blog/text-similarity-search-with-vectors-in-elasticsearch> (letzter Abruf 2.5.2023)

¹⁸<https://www.elastic.co/de/blog/how-to-deploy-nlp-text-embeddings-and-vector-search> (letzter Abruf 2.5.2023)

¹⁹Natürlich können alle Dokumente mit Kosinus-Ähnlichkeiten kleiner gleich 0 ignoriert werden, dennoch müssten viele Dokumente berücksichtigt werden, auch solche mit einer sehr kleinen positiven Kosinus-Ähnlichkeit.

abzugrenzen. Eine Volltextsuche verwendet hierzu das boolesche Entscheidungskriterium des Enthaltenseins von gesuchten Begriffen in den Dokumenten.

Hieraus folgt, dass die Kosinus-Ähnlichkeit lediglich ein Kriterium zur Sortierung passender Dokumente darstellt und die Entscheidung, welche Dokumente zu betrachten sind, auf einem anderen Kriterium basieren muss. Dieses Kriterium wird im Folgenden durch das Konzept des Enthaltenseins der Suchbegriffe in sogenannten *SimSets* dargestellt. Als *SimSets* werden im Folgenden Mengen von abgrenzbaren Begriffen mit ähnlicher, über die Ähnlichkeit ihrer Word-Embeddings ermittelter Bedeutung bezeichnet.

6.3 Semantische Suche mit Word-Embeddings

Anforderungen an eine semantische Suche mit Word-Embeddings in einer unbekanntem Domäne sind:

- Ranking potenziell passender Dokumente mittels Vektor-Ähnlichkeit
- Ermittlung potenziell passender Dokumente anhand von Suchbegriffen und Begriffen, die zu diesen ähnlich sind
- Berücksichtigte ähnliche Begriffe sollten so ähnlich wie möglich zu den Anfragebegriffen sein
- Ähnliche und unähnlichere Begriffe sollten durch eine diskrete Grenze abgrenzbar sein, um ein boolesches Entscheidungskriterium zu bilden mit dem die Treffermenge potenziell passender Dokumente möglichst klein gehalten werden kann
- Geringe, für Suchmaschinen übliche Latenzzeit zum Anfragezeitpunkt

Die wichtigste Eigenschaft aber besteht darin, dass diese Suche auch in neuen Anwendungsgebieten, für die noch kein formales, anwendungsbereichs-spezifisches, terminologisches Wissen existiert, bessere Suchergebnisse – im Sinne eines höheren Recalls – liefert als eine reine Volltextsuche.

Das Problem, das es zu lösen gilt, stellt sich somit wie folgt dar:

Zur Auswahl von relevanten Dokumenten benötigen wir für Anfrageterme eine Menge von synonymen oder hinreichend ähnlichen Termen. Mit dieser Begriffsmenge werden zunächst alle Dokumente ermittelt, in denen mindestens einer der Begriffe vorkommt, um in einem nachfolgenden Schritt anhand der Kosinus-Ähnlichkeit des Anfrage-Embeddings zu den Document-Embeddings das Ranking der Dokumente zu ermitteln.

Um diese Begriffsmengen zu ermitteln, benötigen wir ein Verfahren, um sehr ähnliche Word-Embeddings, die von anderen Word-Embeddings klar abgrenzbar sind, zu identifizieren und die von ihnen repräsentierten Begriffe zu einem *SimSet*²⁰

²⁰In Analogie zu den Synsets in Wortnetzen, bezeichnen wir diese Begriffsmengen als *SimSets*.

zusammenzufassen. Die SimSets können dann analog wie Synonyme bei einer konventionellen semantischen Suche verwendet werden, um die Menge der potenziellen passenden Dokumente mithilfe einer booleschen Suche über einen invertierten Index zu ermitteln.

6.3.1 Ermittlung von SimSets

Wie können solche SimSets ermittelt werden? Word-Embeddings oder allgemeiner Document-Embeddings stellen eine Repräsentation von Begriffen in einem mehrere hunderte von Dimensionen umfassenden Vektorraum dar. Um Gruppen von ähnlichen Begriffen in diesem numerischen Raum zu ermitteln, können Verfahren des unüberwachten Lernens verwendet werden, um Cluster von Embeddings zu identifizieren. Die Menge der durch ein Word-Embedding-Cluster repräsentierten Begriffe bildet dann ein SimSet.

Offensichtlich kann für einen unbekanntem Anwendungsbereich nicht unbedingt angegeben werden, wie viele solcher Begriffs-Cluster in dem Vektorraum existieren bzw. existieren könnten. Parametrisierte Clusteringverfahren, die die Vorgabe der Clusteranzahl voraussetzen, sind somit ungeeignet.

Zwei Klassen von Verfahren zur Ermittlung disjunkter Cluster können daher genutzt werden²¹:

- Community-basierte, graphentheoretische Verfahren, die die Modularität von Graphen ermitteln und für Graphknoten die Community ermitteln, zu denen sie gehören. Die Louvain-Methode (Blondel et al. 2008), die eine „gierige“ (greedy) Optimierungsmethode mit geringer Laufzeitkomplexität, aber potenziellen Fehlzuordnungen ist, kann hierzu herangezogen werden.²²
- Dichtebasierte Verfahren, die die Cluster anhand hoher intra-cluster und geringer inter-cluster Dichte bestimmen. Hierzu zählt die Familie der DBSCAN-Verfahren, i.E. HDBSCAN, OPTICS, etc.²³

6.3.2 Community-basiertes Graph-Clustering

Zum Clustern von Word-Embeddings durch ein graph-basiertes Clustering-Verfahren wie Louvain oder Markov-Clustering ist es notwendig, einen Ähnlichkeitsgraphen der

²¹ Alternativ könnten Gaußsche Multivariate Mixture Modelle in Betracht gezogen werden, um die Mehrfachzuordnung von Vektoren zu unterschiedlichen Clustern zu ermitteln. Diese Verfahren setzen jedoch voraus, dass die Anzahl der Cluster vorgegeben wird.

²² Ein weiteres alternatives community-basiertes Graph-Clustering stellt Markov-Clustering (MCL) (van Dongen 2008) dar, mit dem jedoch noch keine Erfahrungen gewonnen werden konnten.

²³ Weitere Hinweise auf verwandte Clustering-Verfahren finden sich z. B. bei <https://de.wikipedia.org/wiki/DBSCAN> (letzter Zugriff 20.6.2023).

Word-Embeddings zu konstruieren. Hierbei handelt es sich um einen ungerichteten Graphen, der die Kosinus-Ähnlichkeit zwischen Paaren von Word-Embeddings als gewichtete Kanten repräsentiert. Ein solcher Ähnlichkeitsgraph mit Begriffsähnlichkeiten über $\cos(60^\circ) = 0,5$ ist in Abb. 6.3 dargestellt.

Dieser Graph wurde mit Louvain geclustert, die unterschiedlichen Farben stellen die unterschiedlichen Cluster dar. Diese Cluster fassen inhaltlich verwandte Begriffe zusammen, wie z. B. Softskills (pink, links oben), Incentives und Vergünstigungen (orange), Ortsbezeichnungen (rechts unten, ockerbraun), Bewerbungsinformationen (dunkelbraun, links unten), Ausbildung und Studium (blau, Mitte rechts), Aufgaben (grün rechts oben), Anwendungsbereiche (türkis, linke Seite des großen Clusters rechts oben).

Die Cluster in Abb. 6.3 zeigen, dass eine inhaltliche Gruppierung der Begriffe über Word-Embeddings möglich ist. Allerdings sind diese Cluster noch zu umfangreich, um sinnvoll für ein Suchverfahren genutzt werden zu können.

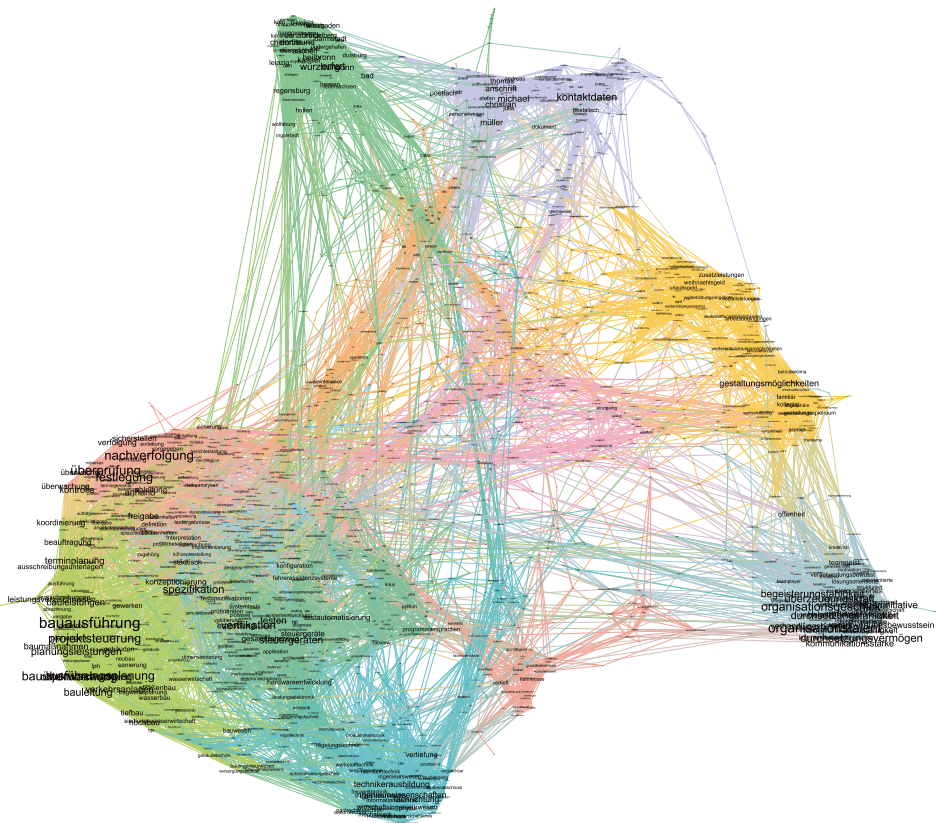


Abb. 6.3 Geclusterter Graph der informativsten Begriffe aus 75.565 Stellenanzeigen für Ingenieure mit einer Kosinus-Ähnlichkeit von 0,5

Wie im Abschnitt 6.2.3 beschrieben, macht es wenig Sinn, alle Paare von Word-Embeddings zu verwenden. Um nicht alle Vektorpaare berücksichtigen zu müssen, ist es sinnvoll, nicht alle Terme und alle Paare zu verwenden, sondern nur diejenigen, die über bestimmten Schwellenwerten liegen.

1. Für den Einsatz in einem Suchverfahren ist es sinnvoll, für die SimSets nur Begriffe in Betracht zu ziehen, die einen hohen Informationsgehalt besitzen, indem nur die – im Sinne der TFIDF – informativsten Begriffe, die oberhalb eines Grenzwerts liegen, betrachtet werden. Begriffe, die häufig in einem Korpus vorkommen und einen geringen Informationsgehalt besitzen, würden die Treffermenge einer Suchfunktion nur vergrößern und wären für die Nutzenden von geringem Wert. Hingegen ist es für eine Suche mit Begriffen, die selten im Korpus auftreten, sinnvoll, auch ähnliche Begriffe in die Suche nach relevanten Dokumenten einzubeziehen, da Nutzende wahrscheinlich eher wenig Wissen über diese Begriffsähnlichkeiten besitzen.
2. Ebenfalls für den Einsatz in einem Suchverfahren, welches neben der eigentlichen Anfrage noch zusätzlich verwandte Begriffe berücksichtigt, ist darauf zu achten, dass die Menge der zusätzlich betrachteten Begriffe nicht zu groß wird, damit Nutzende nicht mit einer Unmenge von Dokumenten überflutet werden. Von daher ist es zweckmäßig für die SimSets nur Begriffspaare mit einer hohen Ähnlichkeit zu berücksichtigen.

Für den Schwellwert 1 ist es aus pragmatischen Gründen sinnvoll, der Pareto-Regel zu folgen und lediglich 80 % oder 90 % der informativsten Begriffe zu nutzen.

Für den Schwellwert 2 haben sich Kosinus-Ähnlichkeiten zwischen 0,85 und 0,95 als zweckmäßig erwiesen. Ein zu kleiner Schwellwert führt zu wenigen sehr großen SimSets (siehe Abb. 6.3), ein zu großer Wert zu wenigen kleinen SimSets.²⁴

Abb. 6.4 illustriert die mit einer Kosinus-Ähnlichkeit von 0,9 aus den 5.640 informativsten Begriffen, die 90 % aller Terme ausmachen, gewonnenen SimSets. Wie ersichtlich wird, besteht der Großteil der SimSets aus wenigen Begriffen. Auffällig sind jedoch einige große SimSets, die inhaltlich aus dem in Stellenanzeigen verwendeten Vokabular herausfallen:

- Die große orange-farbene und die daran links oben anschließende grüne Komponente umfassen englischsprachige Begriffe. Erstere sind allgemeine Begriffe, letztere sind für Stellenangebote spezifische Begriffe. Dass diese Begriffe eine große zusammenhängende Komponente bilden, erklärt sich dadurch, dass sie zum größten Teil aus rein englischsprachigen Sätzen und Stellenanzeigen stammen und nur wenige im Kontext deutschsprachiger Begriffe auftraten.

²⁴Die Untersuchung von (Çelik 2023) deutet darauf hin, dass sich die durchschnittliche Größe der SimSets unabhängig vom Korpus ab einem Kosinus-Schwellwert von 0,65 kaum ändert, die Anzahl der SimSets jedoch abhängig vom Vokabularumfang des Korpus ist.

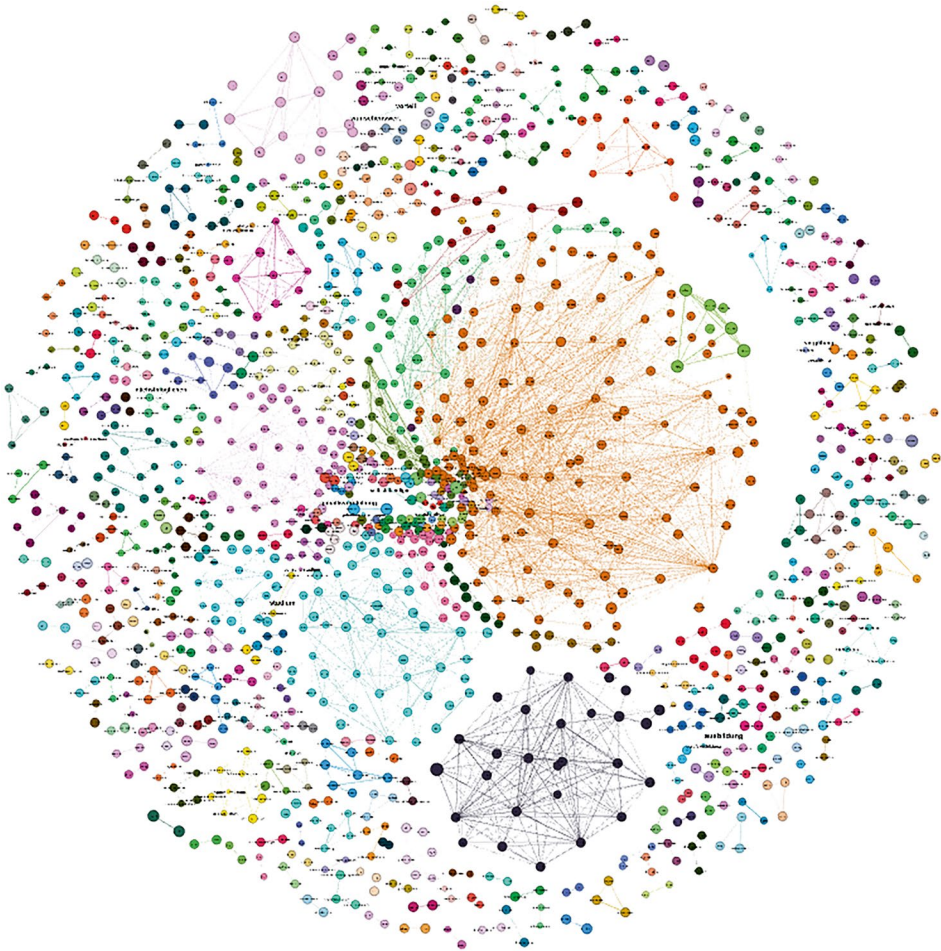


Abb. 6.4 SimSet für die informativsten 5.640 Begriffe aus Stellenanzeigen mit einer Kosinusähnlichkeit von 0,9

- Das Gleiche gilt für die schwarze Komponente darunter, die französischen Worten stammt.
- Die türkisfarbene Komponente links von beiden umfasst Begriffe aus der Softwareentwicklung und ist für Stellenanzeigen von Ingenieuren auch eher untypisch.
- Die pinkfarbene Komponente und darunter die kleine hellblaue umfassen Vornamen von Frauen resp. Männern²⁵. Auch diese Begriffe stehen in der Regel in separaten Kontexten mit Angaben über Kontaktpersonen von Stellenanzeigen.

²⁵Die Farbzuordnung erfolgte rein zufällig und stellt kein geschlechtsspezifisches Vorurteil dar.

6.3.3 Kosinus-Ähnlichkeit revisited

Für den Vergleich von Word-Embeddings wird sehr oft die Kosinus-Ähnlichkeit angewandt.

$$\text{sim}_{cs}(a, b) = \frac{a \cdot b}{\|a\|_2 \|b\|_2}$$

Für die Konstruktion des Ähnlichkeitsgraphen besitzt die Kosinus-Ähnlichkeit jedoch einen entscheidenden Nachteil, sie bildet lediglich die Richtungs-Ähnlichkeit der Vektoren ab. D. h. zwei Embedding-Vektoren können hinsichtlich ihrer Richtung sehr ähnlich sein, jedoch können die damit beschriebenen Raumpunkte in großer räumlicher Entfernung zueinander liegen.

Dies führte auch in der ersten Implementierung zu dem beobachteten Effekt, dass mitunter Begriffe in SimSets auftraten, die inhaltlich keinen Bezug zu den restlichen Begriffen hatten. Zwar kann es immer mal dazu kommen, dass Worte mit stark unterschiedlichen Bedeutungen in gleichen Kontexten verwendet werden. Bei einem Korpus, der sich aber ausschließlich aus einer bestimmten Textkategorie zusammensetzt, erscheint dies eher unwahrscheinlich. Wahrscheinlicher ist es, dass diese inhaltlich abweichenden Begriffe zwar richtungsähnlich, aber räumlich in einer anderen Region des Vektorraums verortet sind, d. h. die Koordinaten vom Ursprung des Koordinatensystems aus betrachtet räumlich vor- oder hinter anderen SimSet-Begriffen liegen.

Um diesen Effekt auszuschließen, ist es zweckmäßiger, die Kosinus-Ähnlichkeit mit einer Distanzgewichtung zu kombinieren.

$$\text{sim}_{cids}(a, b) = w \frac{a \cdot b}{\|a\|_2 \|b\|_2}$$

wobei w eine Gewichtung der euklidischen Distanz der Vektoren wie z. B.

$$w = \frac{1}{1 + \|a - b\|_2}$$

oder

$$w = \frac{1}{e^{\|a - b\|_2}}$$

ist, sodass geringe Abstände höher gewichtet werden als größere.

6.3.4 Dichte-basiertes Clustering

Einfacher ist das Clustering mit einem dichte-basierten Verfahren. Während DBSCAN selber die Vorgabe eines Radius-Parameters ϵ , der die maximale Distanz zwischen Datenpunkten innerhalb eines Clusters angibt, erfordert und dieser nicht ohne weiteres festleg-

bar ist, kommt OPTICS ohne diesen Parameter aus. Beide Verfahren benötigen jedoch einen Parameter minPts , der die minimale Anzahl von Datenpunkten in einem Cluster definiert. Dieser Parameter ist für die SimSets, die ja die Suche nach alternativen Begriffen unterstützen sollen, einfach als zwei definierbar.

Entgegen einem ersten Versuch der Nutzung von OPTICS, der aus Effizienzgründen verworfen wurde und zum Einsatz von Louvain führte, ist eine aktuelle Implementierung von OPTICS mittlerweile dermaßen effizient geworden, sodass dichte-basiertes Clustering aus Performanzgründen vorzuziehen ist.

Zudem ergeben sich zwei weitere Vorteile:

1. Auf die Konstruktion des Ähnlichkeitsgraphen kann verzichtet werden.
2. Anstelle der im Abschnitt Kosinus-Ähnlichkeit revisited beschriebenen distanzgewichteten Kosinus-Ähnlichkeit kann bei der Nutzung von OPTICS direkt die standardmäßig verwendete euklidische Distanz verwendet werden.

Darüber hinaus zeigt sich, dass dichte-basiertes Clustering nicht nur zu zusätzlichen SimSets, sondern auch zu kleineren SimSets führt. Diese Faktoren sind vermutlich durch eine Erhöhung der Dimensionalität der Word-Embeddings nochmals vergrößerbar.

6.3.5 Beispiele von SimSets

Eine große Zahl der automatisch ermittelbaren SimSets umfassen Synonyme und Quasi-Synonyme (siehe Abb. 6.5).²⁶

Bei den Word-Embedding- und Clustering-Verfahren handelt es sich um unüberwachte, heuristische Verfahren, die lediglich auf den Kontexten von Worten resp. der räumlichen Nähe von Vektoren basieren. Die Ergebnisse, die sie produzieren, sind daher stark vom Dokumentenkörper und dem darin verwendeten Sprachgebrauch, der die Satz-Kontexte bestimmt, abhängig. Es kann daher passieren, dass ab und zu auch nicht so passende Worte in die SimSets gelangen.

Abb. 6.6 zeigt einige der Beispiele, in denen durch den Verwendungskontext bestimmte Ausreißer in die SimSets einfließen oder sinnverwandte Begriffe zusammengefasst werden.

²⁶Bei der Verarbeitung wurde auf Stemming oder Lemmatisierung (Hoppe 2020) verzichtet, sodass auch Pluralformen und gebeugte Formen der Wörter in den SimSets auftreten.

```
{
  "0": ["kunden", "großkunden"],
  "1062": ["kenntnisse", "grundkenntnisse", "fachkenntnisse",
    "kenntnis", "grundlagenkenntnisse"],
  "1950": ["weltweit", "international", "global"],
  "1374": ["berufserfahrung", "berufserfahrungen", "praxiserfahrung",
    "projekterfahrung", "berufspraxis", "praxiserfahrungen",
    "industrieerfahrung"],
  "1285": ["arbeitsweise", "denkweise", "vorgehensweise",
    "herangehensweise", "handlungsweise"],
  "1170": ["studium", "hochschulstudium"],
  "1066": ["vergütung", "bezahlung", "entlohnung",
    "verdienstmöglichkeiten", "dotierung"],
  ... }

```

Abb. 6.5 Exemplarische SimSets, die Synonyme resp. Quasi-Synonyme umfassen

```
{
  "1208": ["entwicklungsingenieur", "systemingenieur", "versuchingenieur",
    "testingenieur", "applikationsingenieur", "serviceingenieur",
    "entwicklungsingenieure", "prüfstandsingenieur"],
  "837": ["sitz", "hauptsitz", "nahe", "stammsitz", "firmensitz",
    "hauptstandort"],
  "861": ["gesundheitsmanagement", "betriebliches"],
  "1620": ["arbeiten", "handeln"],
  "1001": ["transportation", "bombardier"],
  "262": ["ios", "android"],
  ... }

```

Abb. 6.6 Exemplarische SimSets mit Ausreißern und sinnverwandten Begriffen

- SimSet 1208 fasst unterschiedliche Spezialisierung zusammen und kann im Kontext einer semantischen Suche nach Stellen für Ingenieure durchaus sinnvoll sein.
- Der Begriff „nahe“ in SimSet 837 scheint aus dem Kontext „Nahe des Stammsitzes“ etc. zu stammen.
- Die Begriffe in Simset 861 gehören zur Nominalphrase „betriebliches Gesundheitsmanagement“ und treten oft im gleichen Kontext auf. Durch eine vorausgegangene Nominalphrasen-Erkennung und -zusammenfassung könnten solche SimSets vermieden werden.
- Da Arbeiten als eine Form des Handelns (im Sinne von Tätig werden) aufgefasst werden kann oder Handeln (im Sinn von Verkaufen) als eine Form von Arbeit, stellt Simset 1620 eher eine Ober-/Unterbegriffsbeziehung dar. Ein multi-sense Word-Embedding-Verfahren könnte hier zu einer besseren Unterscheidung beider Bedeutungen beitragen.
- Die SimSets 1001 und 262 stellen inhaltlich sehr ähnliche Begriffe zusammen, die im Suchkontext durchaus hilfreich sind.

```
{ "1806": ["weltraum", "außergewöhnliches", "erdboden"],
  "604": ["rohrleitungen", "apparaten", "aufstellungsplanung"],
  "964": ["pdf", "download", "herunterladen", "kb"],
  "326": ["hauptstätter", "topstep", "jservice", "crimmitschauer"],
  "510": ["wismar", "stralsund"],
  "371": ["heidi", "jans"],
  "509": ["goerdelerstr", "neubauer"],
  "153": ["neustadt", "donau"],
  ... }
```

Abb. 6.7 SimSets mit kaum erklärbaren Zusammenhängen, starkem Textsortenbezug und benannten Entitäten

Natürlich erzeugen heuristische Verfahren auch sehr ungewöhnliche Ergebnisse, wie Abb. 6.7. zeigt.

SimSets 1806 und 604 sind kaum inhaltlich zu erklären, SimSets 964 und 326 hängen offensichtlich von der Textsorte „Stellenanzeigen“ ab, während SimSets 510, 1371, 509 und teilweise auch 326, benannte Entitäten, i.E. Ortsangaben und Personennamen enthalten. Letztere können durch eine vorausgehende Namend-Entity-Recognition (NER) identifiziert und ausgefiltert werden, da im Kontext einer Stellensuche eine Suche nach benannten Entitäten kaum erfolgt.²⁷

6.3.6 Nutzung der Begriffe in SimSets

SimSets können als eine automatisch aus den Dokumenten abgeleitete Form von domänen-spezifischem, symbolischem, terminologischem Wissen betrachtet werden. Insbesondere für spezialisierte Suchanwendungen, wie z. B. für unternehmensinterne Suchfunktionen, für Suchfunktionen in sehr stark spezialisierten Anwendungsbereichen, oder Suchfunktionen, die einen schnellen Machbarkeitsnachweis erfordern, kann mit Ihnen der Aufwand zur Modellierung, beispielsweise im Rahmen eines Proof-Of-Concept, vermieden werden.

Dieses symbolische Wissen kann im Rahmen einer semantischen Suche dazu verwendet werden, die Treffermenge durch eine boole'sche Suche stark einzuschränken. Hierzu werden die SimSets dazu verwendet, um zusätzliche, die Anfrage ergänzende Begriffe aus den SimSets zu ermitteln. SimSets können daher für eine semantische Suche zur

²⁷Für die Suche nach Stellen an bestimmten Orten wird in der Regel keine semantische Suche im Volltext benötigt, da auf die Stelle bezogene Ortsinformation zweckmäßigerweise besser als Meta-Datum suchbar sein sollten.

- Anfrageerweiterung,
- Annotation der Dokumente, oder
- Erweiterung des invertierten Index

eingesetzt werden.

SimSets stellen darüber hinaus eine hilfreiche Informationsquelle für Wissensingenieure dar, wenn es darum geht, eine qualitativ hochwertige, intellektuelle Modellierung eines Anwendungsgebietes zu erstellen. Im Gegensatz zu Term-Häufigkeitslisten (Hoppe und Tolksdorf 2018), stellen sie eine nach Verwendungsaspekten der Terme erzeugte Gruppierung informativer Begriffe dar. Erfahrungen mit den SimSets zeigen, dass es sich hierbei oft um Gruppierungen von Synonymen, Unterbegriffen, Antonymen oder Nominalphrasen handelt. Ein Wissensingenieur kann diese Beziehungen sehr leicht den SimSets entnehmen und für eine formale Modellierung eines Thesaurus oder einer Ontologie nutzen.

6.4 Anreicherung von Word-Embeddings mit terminologischem Wissen

Bis hier haben wir lediglich gesehen, dass SimSets aus Dokumenten abgeleitet werden können. Solange es noch kein modelliertes symbolisches Wissen über einen Anwendungsbereich in Form einer Taxonomie, eines Thesaurus, eines Wortnetzes, einer Ontologie oder eines Knowledge Graphen gibt, stellen SimSets ein Hilfsmittel dar, um allein aus dem Sprachgebrauch der Dokumente eine semantische Suche zu konstruieren.

Es stellt sich jedoch die Frage: Wie können die aus Dokumenten gewonnenen SimSets, um terminologisches Wissen aus existierenden Quellen, wie Klassifikationshierarchien, Thesauri oder Ontologien erweitert werden? Wie können symbolische Terminologien mit den subsymbolischen Word-Embeddings verknüpft werden?

Die Grundidee des mit (Hoppe 2023) zum Patent angemeldeten Verfahrens besteht hierbei darin, für die Bezeichnungen der in den symbolischen Formalismen repräsentierten Begriffen einen Verwendungskontext zu konstruieren, der der Verwendung des Begriffs in einem Satz entspricht. Da die Bedeutung der Begriffe in diesen Formalismen teilweise durch deren Ober-/Unterbegriffsbeziehungen definiert wird, drängt sich die Idee auf, aus den Pfaden von den Wurzelknoten bis zu den Blattknoten dieser gerichteten azyklischen Graphen (DAG) eine Sequenz von Bezeichnern abzuleiten. Jede dieser Sequenzen kann als ein künstlicher Satz (im Folgenden auch als Pseudo-Satz bezeichnet) betrachtet werden, der die Ober-/Unterbegriffs-Beziehungen der Begriffe beschreibt.

Werden bei der Ableitung der Pseudo-Sätze Synonyme, alternative und versteckte Bezeichnungen, Abkürzungen und fremdsprachliche Übersetzungen berücksichtigt²⁸,

²⁸Dieser Konstruktionsprozess ist vorsichtig zu konzipieren, da die Berücksichtigung alternativer Bezeichnungen zu einem exponentiellen Wachstum der Anzahl der Pseudo-Sätze führen kann.

```

["Fähigkeiten", "Arbeiten mit Computern", "Nutzung digitaler Tools für die
Zusammenarbeit, Inhaltserstellung und Problemlösung", "Nutzung digitaler
Tools für Zusammenarbeit und Produktivität", "Erstellung digitaler
Inhalte", "Io", "Markup-Programmierung", "Live script"]
["Fähigkeiten", "Arbeiten mit Computern", "Nutzung digitaler Tools für die
Zusammenarbeit, Inhaltserstellung und Problemlösung", "Nutzung digitaler
Tools für Zusammenarbeit und Produktivität", "Erstellung digitaler
Inhalte", "Io", "Markup-Programmierung", "Java script"]
["Fähigkeiten", "Arbeiten mit Computern", "Nutzung digitaler Tools für die
Zusammenarbeit, Inhaltserstellung und Problemlösung", "Nutzung digitaler
Tools für Zusammenarbeit und Produktivität", "Erstellung digitaler
Inhalte", "Io", "Markup-Programmierung", "JavaScript"]
["Fähigkeiten", "Unterstützung und Pflege", "Schutz und Durchsetzung",
"Kontamination vermeiden"]
["Fähigkeiten", "Unterstützung und Pflege", "Schutz und Durchsetzung",
"Verunreinigung vermeiden"]
["Fähigkeiten", "Unterstützung und Pflege", "Schutz und Durchsetzung",
"Drogen- und Alkoholmissbrauch bemerken", "Medikamenten- und
Alkoholabhängigkeit von Patienten/Patientinnen beurteilen"]
["Fähigkeiten", "Unterstützung und Pflege", "Schutz und
Durchsetzung", "Medikamentenmissbrauch aufdecken", "Medikamenten- und
Alkoholabhängigkeit von Patienten/Patientinnen beurteilen"]

```

Abb. 6.8 Pseudo-Sätze mit Synonymen abgeleitet aus dem ESCO-Thesaurus

dann treten diese in gleichen Kontexten auf. Im Vektorraum der Word-Embeddings sollten diese in enger räumlicher Nähe angeordnet und durch Clustering mit ähnlichen Dokumentbegriffen zu SimSets zusammengefasst werden.

Abb. 6.8 zeigt einige der, aus dem ESCO-Thesaurus²⁹ gewonnenen, 351.879 Pseudo-Sätze. Hierbei wurden synonyme Bezeichnungen, die in gleichen Kontexten modelliert wurden, durch Fettdruck hervorgehoben. Die Zeichenketten werden hier als atomar betrachtet und können als Linearkombination der in ihnen verwendeten Word-Embeddings bestimmt und so mit diesen einzelnen Begriffen in Ähnlichkeitsbeziehungen gesetzt werden.

Werden diese Pseudo-Sätze mit einem Word-Embedding-Verfahren inkrementell verarbeitet, welches mit den aus den Dokumenten gewonnenen Word-Embeddings initialisiert wurde, so können die Begriffe aus den Dokumenten mit den Begriffszusammenhängen aus dem terminologischen Wissen kombiniert und in die Word-Embeddings integriert werden.

²⁹ <https://esco.ec.europa.eu/de> (letzter Zugriff 11.7.2023)

6.5 Erfassen SimSets die Begriffssemantik?

Word-Embedding-Verfahren nutzen Kontextinformation, um die räumliche Anordnung von Wortvektoren in einem Vektorraum zu ermitteln. Wörter, die in ähnlichen Kontexten verwendet werden, werden in diesem Vektorraum in räumlicher Nähe angeordnet. Das Clustering der Wortvektoren erzeugt SimSets von räumlich nahen und von anderen Clustern abgrenzbaren Wortvektoren. Es stellen sich die Fragen:

- Erfassen SimSets überhaupt Semantik und repräsentieren sie Bedeutung?
- Wie kann dies überprüft bzw. nachgewiesen werden?

Ein Nachweis mit Hilfe eines Dokumentenkorporus ist schwierig, da es keinen Goldstandard gibt, an dem überprüft werden kann, ob die SimSets inhaltlich Sinn machen. Aus Dokumenten abgeleitete SimSets könnten natürlich einigen Fachexperten oder einer großen Menge von Nutzenden vorgelegt werden und diese nach ihrer Meinung zur Qualität der SimSets befragt werden. Dieses Vorgehen ist jedoch mit Unsicherheiten behaftet, müsste daher mit einer großen Zahl von Probanden durchgeführt werden und würde somit großen Aufwand nach sich ziehen.

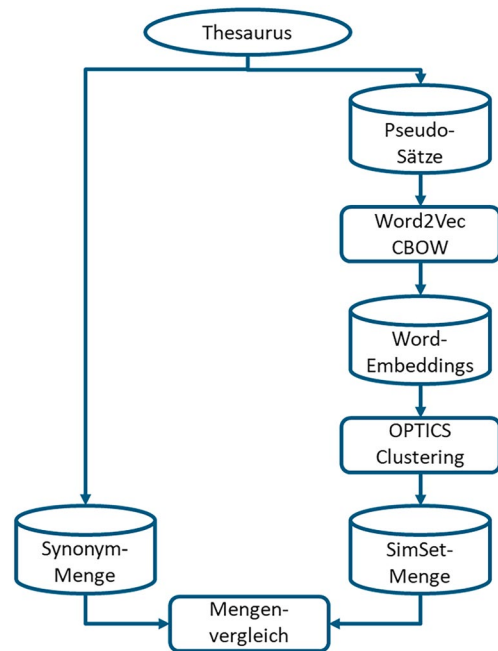
6.5.1 Rekonstruktion von Synonymen

Ein gangbarer Nachweis besteht in einem Rekonstruktionsexperiment. Bei diesem Experiment werden aus einer von Experten erzeugten kontrollierten Terminologie, in Form eines Thesaurus oder einer Ontologie, Pseudo-Sätze abgeleitet und überprüft, ob aus diesen ein Teil der Terminologie rekonstruiert werden kann. Als zu rekonstruierender Teil erweisen sich Synonyme als zweckmäßig. Einerseits, da die CBOV-Variante von Word-Embeddings aus gleichen Verwendungskontexten ähnliche Begriffe ermittelt und andererseits, das Clustering von Word-Embeddings auf die Abgrenzung der ähnlichsten Begriffe abzielt.

Zur Rekonstruktion werden aus der Terminologie Pseudo-Sätze abgeleitet, wobei auch mit Synonymen gebildete Varianten berücksichtigt werden. Für die Bezeichnungen in den Pseudo-Sätzen werden Word-Embeddings errechnet und diese zu SimSets geclustert. Entsprechen diese SimSets nun einem Teil der Terminologie, i.E. den Synonymen des Thesaurus oder der Ontologie, dann erfassen sie die von Experten kodierte Bedeutung dieser Teile. Woraus folgt, dass mit den SimSets ein Teil der Bedeutung von Begriffen erfasst werden kann.

Abb. 6.9 visualisiert das Rekonstruktionsexperiment, das mit den deutschsprachigen Begriffen und deren Synonymen des ESCO-Thesaurus durchgeführt wurde. Hierfür wurden als Vergleichsmenge zunächst alle Mengen von synonymen Bezeichnungen aus dem Thesaurus extrahiert. Aus den Ober- und Unterbegriffsbeziehungen wurden, unter Be-

Abb. 6.9 Struktur des Rekonstruktionsexperiments



rücksichtigung aller synonymen Bezeichnungen, alle möglichen Pseudo-Sätze generiert (wie in Abb. 6.8 dargestellt) und alle SimSet-Mengen berechnet.

6.5.2 Ergebnisse des Rekonstruktionsexperiments

Zum Vergleich der SimSet-Mengen mit den Synonym-Mengen ist jeweils zu ermitteln, welche Mengen miteinander zu vergleichen sind. Hierbei gibt es vier unterschiedliche Vergleichsmöglichkeiten:

- Gleichheit von Synonym-Menge und SimSet
- SimSet ist echte Teilmenge einer Synonym-Menge
- Synonym-Menge ist echte Teilmenge eines SimSets
- Synonym-Menge und SimSet besitzen eine nicht leere Schnittmenge

Prinzipiell müssten alle Synonym-Mengen mit allen SimSets verglichen werden, damit würden sowohl unnötige als auch unsinnige Vergleiche gemacht werden. Sinnvoller ist es daher, die Reihenfolge der Vergleiche entsprechend der obigen Vergleichsreihenfolge durchzuführen und die jeweils genutzten Begriffsmengen von den nachfolgenden Ver-

gleichen auszuschließen. Diese Vorgehensweise ist korrekt, da alle obigen Vergleich disjunkt sind.

Im Rekonstruktionsexperiment der 9324 Synonym-Mengen des ESCO-Thesaurus konnten 6953 SimSets ermittelt werden. Tab. 6.1 fasst die Ergebnisse des Vergleichs zusammen.

Insgesamt konnten für 8031 der 9324 Synonym-Mengen (86,1 %) als SimSets rekonstruiert werden. 32,1 % der SimSets rekonstruierten die Synonyme korrekt, für 36,7 % der Synonyme wurden SimSets als Teilmengen identifiziert und 3,6 % der SimSets waren Obermengen von Synonymen. 13,7 % der SimSets hatten nichtleere Schnittmengen mit Synonymen.

Dieses Experiment zeigt, dass durch SimSets ein Großteil der von Experten modellierten Synonym-Mengen zumindest teilweise rekonstruiert werden kann. Hieraus folgt, dass der Konstruktionsprozess der SimSets einen Teil der Bedeutung der modellierten Begriffe erfassen kann. Wird der Konstruktionsprozess der SimSets somit allein auf Dokumenttexten angewandt, dann werden mit den abgeleiteten SimSets auch Teile der Bedeutung der in den Dokumenten verwendeten Begriffe ermittelt.

6.6 Technologien und Implementierung

Die Umsetzung der Implementierung erfolgte in Python unter Verwendung der Standardbibliotheken NLP, Spacy, Gensim, Louvain und Scikit-Learn.

Der aufwendigste Teil der Implementierung ist hierbei die Tokenisierung der Dokumenttexte, bei der Stoppwörter durch ein spezielles Token „UNK“ ersetzt werden, um die Abfolge der Wörter nicht zu verändern. Hierbei werden gleichzeitig Daten über die Termfrequenz (TF) und inverse Dokumentenfrequenz (IDF) erhoben. Prinzipiell könnte hierbei gleichzeitig eine Lemmatisierung der Worte erfolgen, um diese auf einen einheitlichen Wortstamm abzubilden. Es hat sich jedoch gezeigt, dass hierdurch Fehler eingeführt werden, sodass der Nutzen kaum gerechtfertigt ist.

Je nach Textsorte und Anwendungsgebiet ist es zweckmäßig, auch benannte Entitäten wie Orts- oder Personenbezeichnungen zu ignorieren, indem sie ebenfalls durch „UNK“ ersetzt werden, nachdem sie durch ein Named-Entity-Recognition Verfahren erkannt wurden.

Für die Berechnung der Word-Embeddings wird die Word2Vec Implementierung von Gensim in der CBOW-Variante verwendet. Für die Ermittlung ähnlicher Begriffe erweist sich diese als geeigneter als die SkipGram-Variante, mit der eher Begriffe, die zueinander in Beziehung stehen, aus dem Satzkontext ermittelt werden. Prinzipiell kann für die Berechnung der Word-Embeddings von sehr großen Korpora auch auf GPU-basierte Implementierungen zurückgegriffen werden.³⁰ Für die bisher betrachteten Korpora mit rd. 46.000 bzw. 75.000 Dokumenten war dies bisher noch nicht notwendig.

³⁰Wie z. B. <https://cusim.readthedocs.io/en/latest/>

Tab. 6.1 Ergebnisse des Vergleichsexperiments

Vergleich	Synonym-Mengen	SimSets	Anteil
=	2995	2995	32,1 %
\supset	3418		36,7 %
\subset		340	3,6 %
$\cap \neq \emptyset$	1278	1278	13,7 %

Die Word2Vec Implementierung von Gensim ist darüber hinaus inkrementell einsetzbar. Zusätzliche Dokumente können die Bedeutung von Begriffen verändern, sodass deren Word-Embeddings zu aktualisieren sind. Insbesondere in Anwendungsfällen, in denen neue Dokumente hinzukommen oder sich der Sprachgebrauch kontinuierlich verändert, wie bei Stellen- oder Weiterbildungsangeboten, ist dies wichtig. Für die richtige Nutzung dieser inkrementellen Aktualisierung von Word-Embeddings muss jedoch darauf geachtet werden, dass zunächst das von Word2Vec genutzte Vokabular aktualisiert werden muss – um auch neue Begriffe als neue Wortvektoren in die Berechnung einzubeziehen – bevor die Aktualisierung der Word-Embeddings erfolgen kann. Mit dieser inkrementellen Aktualisierung lässt sich zu einem späteren Zeitpunkt auch existierendes terminologisches Wissen, wie oben beschrieben, in die Word-Embeddings integrieren.

Aufwendig erweist sich die Ermittlung des Ähnlichkeitsgraph, da hierfür prinzipiell $n * (n - 1)$ Kombinationen von Termen betrachtet werden müssen. Zweckmäßig ist es daher

- sich auf die 80–90 % der informativsten Terme zu beschränken und
- in den Ähnlichkeitsgraphen nur Kanten zu den Termen, die eine über einem Kosinus-Schwellenwert liegende Ähnlichkeit besitzen, in den Graph aufzunehmen.

Kosinus-Schwellenwerte unterhalb von 0,85 erzeugen in der Regel zu große SimSets (siehe Abb. 6.3). Diese haben in der Regel zur Folge, dass zu viele relevante Dokumente identifiziert werden. Größere Kosinus-Schwellenwerte, insbesondere oberhalb von 0,95, führen dazu, dass sich die Anzahl der SimSets stark verringert und sich somit ihr Nutzen für eine semantische Suche verringert.

Die Erfahrung zeigt, dass ein dichte-basiertes Clustering, z. B. durch OPTICS, zusätzliche und insgesamt kleinere SimSets ermittelt als Graph-basiertes Clustering mit Louvain.

6.7 Empfehlungen

Zusammenfassend können folgenden Empfehlungen ausgesprochen werden:

1. Eine naive Nutzung von Word-Embeddings in einer semantischen Suche, die allein auf der Nutzung der Kosinus-Ähnlichkeit basiert, reicht nicht aus, um die Treffermenge sinnvoll einzuschränken.

2. Die Nutzung der Kosinus-Ähnlichkeit von Word-Embeddings berücksichtigt nur die Richtungs-Ähnlichkeit, aber nicht die Distanz-Ähnlichkeit, sodass auch Begriffe, die im Vektorraum der Word-Embeddings weit voneinander entfernt lokalisiert sind, als ähnlich betrachtet werden.
3. Für Anwendungsgebiete, in denen modelliertes Hintergrundwissen noch nicht verfügbar ist, stellt der beschriebene Ansatz eine Lösung dar, um eine über eine einfache Volltextsuche hinausgehende einfache semantische Suche schnell zu realisieren.
4. Die beschriebene Vorgehensweise erlaubt es darüber hinaus zusätzliches terminologisches Wissen, in Form von Synonymen in Thesauri, Ontologien oder Knowledge Graphen, oder SynSets von Wortnetzen zu einem späteren Zeitpunkt zu integrieren.
5. SimSets bilden für Wissensingenieure ein hilfreiches Werkzeug zur Vereinfachung des Modellierungsprozesses, da sie semantisch ähnliche Begriffe zweckmäßig gruppieren.
6. Je nach Anwendungsgebiet ist die Suche nach Orts- oder Personenbezeichnungen nicht sinnvoll, beispielsweise für die Suche nach Stellen- oder Weiterbildungsangeboten. Diese sollten mithilfe einer Named-Entity-Recognition von der Verarbeitung ausgeschlossen werden.

6.8 Fazit

Der in diesem Kapitel beschriebene Ansatz zu einer semantischen Suche basierend auf Word-Embeddings füllt eine Lücke zwischen einfacher Volltextsuche, deren naiver Erweiterung um subsymbolische Word-Embeddings und deren Vergleichen mithilfe der Kosinus-Ähnlichkeit, einerseits, und wissensintensiven, auf symbolischen Wissensrepräsentationen basierenden semantischen Suchverfahren, andererseits.

Die Kombination von unüberwachten Verfahren zur Berechnung von subsymbolischen Word-Embeddings und deren Clustering ermöglicht es, eine symbolische Repräsentation von SimSets abzuleiten, die Synonyme, Quasi-Synonyme und sehr ähnliche Begriffe voneinander abgrenzen.

Mit dieser symbolischen Repräsentation kann nicht nur ein darauf aufbauendes, effizientes semantisches Suchverfahren implementiert werden, sondern auch die damit ermittelten Treffermengen gegenüber einer naiven Verwendung von Word-Embeddings eingeschränkt werden. Die Qualität dieser neuen Form semantischer Suche reicht an eine konventionelle semantische Suche heran (Celik 2023), da sie aber auf der heuristischen Komponente der SimSet-Berechnung basiert, kann aber nicht jede Suchanfrage verbessert werden.

Durch die Erweiterung des Berechnungsverfahrens der SimSets zur Integration symbolischen Wissens in die Word-Embeddings wird nicht nur die Verwendung von Anwendungsbereichs-spezifischen Terminologien möglich, mit ihr konnte auch nachgewiesen werden, dass SimSets in der Tat die Semantik von Begriffen erfassen.

SimSets bilden darüber hinaus ein für Wissensingenieure hilfreiches Werkzeug, um den Modellierungsprozess durch die Vorverarbeitung von Begriffen zu eng

zusammengehörigen Mengen von Bezeichnungen zu unterstützen. Eine gezielte Entwicklung von Verfahren zur Identifikation von Begriffsbeziehungen, die über reine Synonym- und Quasi-Synonym-Beziehungen hinausgehen und z. B. Ober-/Unterbegriffsbeziehungen umfassen, stellt sowohl ein lohnendes Ziel für eine weitergehende Unterstützung von Wissensingenieuren, als auch zur Verbesserung dieser neuen Form semantischer Suche dar.

Literatur

- (Hoppe, 2020) Semantische Suche – Grundlagen und Methoden semantischer Suche von Textdokumente, Springer Vieweg, Wiesbaden, 2020. ISBN 978–3–658–30426-3 (mit Beiträgen von Bernhard Humm).
- (Hoppe, 2015) Modellierung des Sprachraums von Unternehmen, in: „Corporate Semantic Web – Wie semantische Anwendungen in Unternehmen Nutzen stiften“, B. Humm, A. Reibold (Hrsg.), Springer Verlag, 2015, <http://www.springer.com/springer+vieweg/it+%26+informatik/anwendungen/book/978-3-642-54885-7>
- (Hoppe & Junghans, 2015) Messung des Nutzens semantischer Suche, mit H. Junghans, in: „Corporate Semantic Web – Wie semantische Anwendungen in Unternehmen Nutzen stiften“, B. Ege, B. Humm, A. Reibold (Hrsg.), Springer Verlag, 2015, <http://www.springer.com/springer+vieweg/it+%26+informatik/anwendungen/book/978-3-642-54885-7>
- (Hoppe & Tolksdorf, 2018) Guide for Pragmatical Modelling of Ontologies in Corporate Settings, mit R. Tolksdorf, in: „Semantic Applications - Methodology, Technology, Corporate Use“, T.Hoppe, B. Humm, A. Reibold, (Hrsg.), Springer Verlag, 2018.
- (Wittgenstein, 1977) Philosophische Untersuchungen, Ludwig Wittgenstein, suhrkamp taschenbuch wissenschaft, 203, 1977, Übersetzung von Philosophical Investigations, Basil Blackwell, 1958
- (Firth, 1957) Papers in Linguistics (1934–1951). Oxford University Press, 1957.
- (van Harmelen & ten Teije, 2019) “A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems”, Frank van Harmelen, Annette ten Teije, Journal of Web Engineering, Vol. 18 1–3, 97–124. <https://doi.org/10.13052/jwe1540-9589.18133>
- (Witschel et al., 2021) “Visualization of patterns for hybrid learning and reasoning with human involvement”, Hans Friedrich Witschel, Charuta Pande, Andreas Martin, Emanuele Laurenzi, Knut Hinkelmann, In: Dornberger, R. (eds) New Trends in Business Information Systems and Technology. Studies in Systems, Decision and Control, vol 294. Springer, Cham. https://doi.org/10.1007/978-3-030-48332-6_13
- (Mikolov et al., 2013) Word2Vec: Efficient Estimation of Word Representations in Vector Space, Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, <https://arxiv.org/abs/1301.3781> (letzter Zugriff 6.6.2023).
- (Pennington, Socher, Manning, 2014) GloVe: Global Vectors for Word Representation, Jeffrey Pennington, Richard Socher, Christopher D. Manning, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, October 25–29, 2014, Doha, Qatar.
- (Bojanowski et al., 2016) Enriching Word Vectors with Subword Information, Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov, [1607.04606] Enriching Word Vectors with Subword Information, arxiv.org, 2016 (letzter Zugriff 20.6.2023)

- (Vilnis & MacCallum, 2014) Word Representations via Gaussian Embedding, Luke Vilnis, Andrew MacCallum, [1412.6623] Word Representations via Gaussian Embedding arxiv.org, 2014 (letzter Zugriff 20.6.2023)
- (Devlin et al., 2018) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, <https://arxiv.org/abs/1810.04805v2> arxiv.org, 2018, (letzter Zugriff 9.7.2023)
- (Bražinskas, Havrylov, Titov, 2018) Embedding Words as Distributions with a Bayesian Skip-gram Model, in Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018 Santa Fe, New Mexico, USA, 2018. Embedding Words as Distributions with a Bayesian Skip-gram Model – ACL Anthology (letzter Zugriff 20.6.2023)
- (Peters et al., 2018) Deep contextualized word representations, Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer, [1802.05365] Deep contextualized word representations, arxiv.org, 2014 (letzter Zugriff 20.6.2023)
- (Akbik, Blythe, Vollgraf, 2018) Contextual String Embeddings for Sequence Labeling, in Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018 Santa Fe, New Mexico, USA, 2018. Contextual String Embeddings for Sequence Labeling – ACL Anthology (letzter Zugriff 20.6.2023)
- (Blondel et al., 2008) Fast unfolding of communities in large networks, Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Etienne Lefebvre, Journal of Statistical Mechanics: Theory and Experiment. 2008 (10): P10008. <https://arxiv.org/abs/0803.0476> (letzter Zugriff 20.6.2023)
- (vanDongen, 2008) Graph clustering via a discrete uncoupling process, Stijn van Dongen, Siam Journal on Matrix Analysis and Applications 30–1, p.121–141, 2008, <https://doi.org/10.1137/040608635>, (letzter Zugriff 9.7.2023)
- (Celik, 2023) Evaluation eines Word-Embedding-basierten Information-Retrieval Verfahrens, Hüseyin Çelik, Masterarbeit, Technische Hochschule Brandenburg, Fachbereich Informatik und Medien, Mai 2023.
- (Hoppe, 2023) Verfahren und Vorrichtung zum Trainieren eines Word-Embedding Verfahren, Thomas Hoppe, Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V., Patentanmeldung 10 2023 116 650.6

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Vorhersage von landwirtschaftlichen Erträgen und Wachstum

7

Alexander Münzberg und Miro Miranda Lorenz

Kernaussagen

1. Hybride KI-Modelle sind für das Modellieren von pflanzlichem Wachstum und Erträgen geeignet.
2. Erdbeobachtungsdaten werden in der Praxis vielseitig eingesetzt, um landwirtschaftliche Erträge zielgenau vorherzusagen.
3. Die Kopplung von maschinellem Lernen mit prozessbasierten Simulationssystemen als Lieferanten von symbolisch aufbereiteten Daten findet bereits Anwendung in der Praxis.
4. Autoencoder werden u. a. angewendet im Bereich der Datenkomprimierung und der Unterdrückung von Rauschen, haben sich aber auch für die Interpolation landwirtschaftlicher Daten als reif erwiesen.
5. Convolutional Neural Networks (CNN) eignen sich neben der Bildanalyse auch für die Zeitreihenprognose, insbesondere für hochdimensionale Zeitreihendatensätze.
6. Mit Wissensgraphen können Daten so aufbereitet werden, dass deren Semantik, Beziehungen und Klassen strukturiert und maschinenlesbar dargestellt werden können.

A. Münzberg · M. Miranda Lorenz (✉)
Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Kaiserlautern, Deutschland
E-Mail: miro.miranda_lorenz@dfki.de

A. Münzberg
E-Mail: alexander.muenzberg@dfki.de

7.1 Einleitung

Künstliche Intelligenz (KI) und Landwirtschaft können inzwischen problemlos in einem Atemzug genannt werden. Spätestens seit dem Aufkommen des *Digital Farming*, also einer Landwirtschaft, die immer digitalisierter wird, finden auch Methoden Einzug, die in hohem Maße Daten verarbeiten können. Dabei spielt das datengetriebene Modellieren von pflanzlichen Prozessen eine wichtige Rolle. Das Modellieren von Pflanzenverhalten, Nährstoffbedarf und Umweltfaktoren sind hier eine wichtige Aufgabe. Es unterstützt Landwirte und Agrarunternehmen bei der Entscheidungsfindung, der Anpassung an sich verändernde Klimabedingungen, Schädlings- und Unkrautbekämpfung sowie Dünge- und Erntezeitpunktbestimmung. Mithilfe von Sensoren, Datenanalysen und KI-Systemen können Landwirte, Agrarunternehmen und auch Agrarforschungseinrichtungen ihre Ressourcen effizienter nutzen, Erträge maximieren und Umweltauswirkungen minimieren. Die Anwendungsgebiete von KI in der Landwirtschaft sind so vielfältig wie die Landwirtschaft selbst. Auch die Gründe für den Einsatz von KI sind vielfältig. Heute steht die Landwirtschaft vor globalen Herausforderungen. Eine steigende Weltbevölkerung und ein größer werdender Bedarf nach Umweltschutz erfordern einen effizienteren Ressourceneinsatz bei weiter steigenden Erträgen. In der Vergangenheit konnte der landwirtschaftliche Ertrag mit dem Bevölkerungswachstum Schritt halten. Dies ist vor allem auf eine zunehmende Technisierung und einen erhöhten Ressourceneinsatz zurückzuführen, welcher mit hohen externen Effekten verknüpft ist. So hat die Landwirtschaft heute einen großen Anteil am Ausstoß klimawirksamer Treibhausgase, aber auch mit regionalen Problemen zu kämpfen, wie Bodenverdichtung, Wasserknappheit, Grundwasserunreinigungen und Resistenzentwicklungen. Um den Balanceakt zwischen Ertragssteigerungen und effizientem Ressourceneinsatz zu meistern, bedarf es neuer Technologien. Der Einsatz von KI hat in vielen Bereichen bereits große Erfolge gezeigt. Damit Machine Learning (ML) einen Beitrag leisten kann, müssen genügend Daten vorhanden sein. Dies ist oftmals ein limitierender Faktor. Auch ist die Erklärbarkeit von KI ein limitierender Faktor für deren Einsatz. Eine mögliche Lösung kann die Integration von zusätzlichem Wissen sein. Sogenannte hybride Methoden sind vielversprechend, da sie auf weniger Trainingsdaten angewiesen sind und oftmals eine höhere Erklärbarkeit aufweisen. Auch wenn diese Ansätze noch neu sind, zeichnet sich schon heute ein großes Potenzial ab für die Ertrags- und Wachstumsvorhersage zur Entscheidungsunterstützung in der Landwirtschaft.

Abb. 7.1 ordnet dieses Kapitel thematisch ein bezüglich der Branche, Anwendungen, Daten sowie Technologien.

Abb. 7.2 zeigt das Boxology-Modell¹ des hybriden KI-Systems. Hierbei werden die Input-/Output-Fähigkeiten von landwirtschaftlichen Simulationssystemen ausgenutzt,

¹ Für eine Einführung in die Boxology siehe Kap. 1.

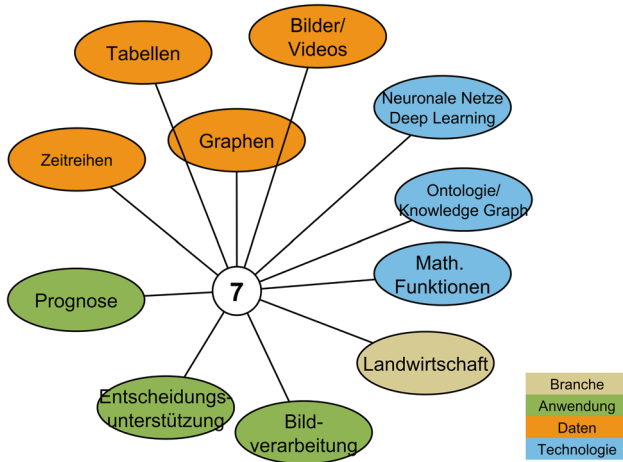


Abb. 7.1 Thematische Einordnung des Kap. 7

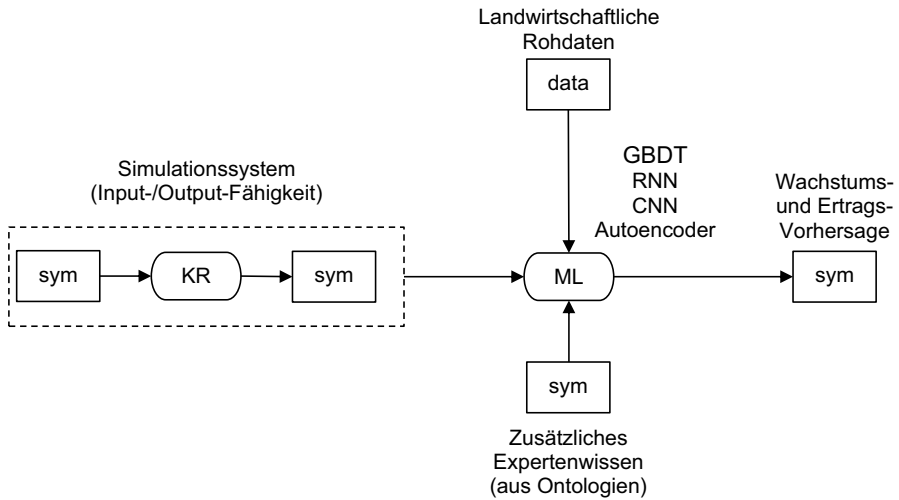


Abb. 7.2 Boxology-Modell des hybriden KI-Systems

welche größtenteils auf Expertenwissen basieren, um zusammen mit zusätzlichem Wissen aus Ontologien und weiteren aufbereiteten Rohdaten ML-Verfahren zu trainieren.

7.2 Simulationssysteme

Vor dem Aufkommen von datengetriebenen Methoden wurden biologische Zusammenhänge und Interaktionen experimentell beschrieben und in prozessbasierte Simulationsmodelle übersetzt. Solche mechanistischen dynamischen Modelle simulieren das tägliche Pflanzenwachstum auf Grundlage der zugrunde liegenden Prozesse wie Photosynthese und Respiration und wie diese Prozesse durch die Umweltbedingungen beeinflusst werden. Solche Modelle sind im Gegensatz zu ML-Verfahren auf weniger Daten angewiesen. In der Anwendung kommen prozessbasierte Modelle jedoch schnell an ihre Grenzen, aufgrund eines hohen Kalibrierungsaufwands, regionaler Unterschiede und der Tatsache, dass diese nur eine Vereinfachung der Realität darstellen. Resultierend sind Modellierungen oft ungenau. Derzeit versuchen Wissenschaftler und Wissenschaftlerinnen, die Brücke zwischen ML-Verfahren und Expertensystemen zu bauen. Ziel sind hybride Simulationsmodelle, welche die Vorteile aus beiden Welten vereinen, also das Lernen in datenarmen Regimen verbunden mit einer erhöhten Erklärbarkeit. Wissensbasierte KI baut auf verschiedenen Wissensquellen auf. Diese können verschiedene Repräsentationen haben, die in die folgenden Kategorien unterteilt werden können (von Rueden et al. 2021):

- Regelsysteme,
- Wissensgraphen,
- Gleichungssysteme,
- probabilistische Zusammenhänge,
- Simulationen.

In der Landwirtschaft spielen Simulationssysteme und Wissensgraphen eine wichtige Rolle und sind Gegenstand der Entwicklung von Hybriden-KI-Anwendungen. So finden sich derzeit hauptsächlich Ansätze, die den Datenraum, also Messwerte, Beobachtungen, Sensordaten, historische Aufzeichnungen und andere relevante Daten, durch Simulationsergebnisse und Wissensgraphen erweitern (Shahhosseini et al. 2021; Jeong et al. 2022; Shuai und Basso 2022; Muenzberg et al. 2023). Ferner spielt das Imitieren von Simulationsmodellen eine entscheidende Rolle (Muenzberg et al. 2023).

Im Nachfolgenden werden zwei Anwendungsgebiete der Ertrags- und Wachstumsvorhersagen genauer erläutert. Zunächst wird auf die Ertragsvorhersage mittels Fernerkundung eingegangen und anschließend wird die Zeitreihenvorhersage landwirtschaftlicher Daten mittels Transfer Learning näher beschrieben.

7.3 Ertragsvorhersagen durch Fernerkundung

Der Einsatz von Fernerkundung, einem Verfahren, mit dem der Zustand der Erdoberfläche erfasst wird, hat im Bereich der Landwirtschaft großes Potenzial, besonders in Bezug auf Ertragsvorhersagen. Diese können einen vielseitigen Beitrag zu einer zukunftsfähigen Landwirtschaft leisten, etwa zur Erhöhung der Ressourceneffizienz, Nahrungsmittelsicherheit, für Risikomanagement oder Marktplanung. Fernerkundungstechnologien haben besonders in der Skalierbarkeit einen entscheidenden Vorteil. So können große Flächen global und kostengünstig abgedeckt werden. Dies ermöglicht die Entwicklung von global skalierbaren Modellen, ein entscheidender Vorteil gegenüber traditionellen Methoden. Satellitenbilder sind besonders geeignet, um die Wachstumsperiode zu überwachen und Rückschlüsse auf den potenziellen Ertrag zu erzielen. Zusätzliche Datenquellen wie Wetterdaten, Bodeninformationen, Managementdaten und Höhenprofile (DEM) liefern weitere Informationen, welche nicht in Bilddaten enthalten sind. Diese helfen, wichtige Zusammenhänge zu lernen, etwa den Zusammenhang zwischen Wetterereignissen und Ertrag oder den Einfluss von Staunässe, verursacht durch unterschiedliche Höhenprofile innerhalb eines Feldes. Obwohl im Bereich der Fernerkundung kein Mangel an Trainingsdaten besteht, herrscht oftmals ein großer Mangel an Ground-Truth-Daten, die es erlauben, die Qualität von Modellen zu überprüfen. Besonders bei Erträgen auf Feldebene und innerhalb eines Feldes besteht großer Mangel an Trainingsdaten. Um der geringen Datenverfügbarkeit entgegenzuwirken, werden hybride Modelle entwickelt. Dabei spielen prozessbasierte Simulationen eine wichtige Rolle. Mittels Simulationsdaten können ML-Modelle effizienter trainiert werden. Tägliche Informationen über Biomasse, Wasserstress oder Wachstumsstadium können in den Lernprozess integriert werden und ergänzen Trainingsdaten. Abhängig von der Region erweisen sich Bodenfeuchte, Wasserstress und Grundwasserspiegel als wichtige Input-Features (Shahhosseini et al. 2021). Das Anreichern von Trainingsdaten durch Simulationsmodelle hat in einigen Studien bereits großes Potenzial gezeigt (Shahhosseini et al. 2021; Jeong et al. 2022; Shuai und Basso 2022). Jedoch ist die Auswahl eines geeigneten Simulationsmodells von grundlegender Bedeutung. Ebenso müssen Anwender darauf achten, dass Simulationsmodelle einen hohen Kalibrierungsaufwand haben. Folglich sind Simulationen oft ungenau. Daher kann ein unzureichend kalibriertes und schlecht angepasstes Modell einen erfolgreichen Lernprozess verhindern.

7.4 Ertragsvorhersage durch Transfer Learning

Im vom Bundesministerium für Bildung und Forschung geförderten Projekt SimLearn adressiert ein Konsortium aus verschiedenen Institutionen (Deutsches Forschungsinstitut für Künstliche Intelligenz (DFKI), Universität Hohenheim und Kuratorium für Technik und Bauwesen in der Landwirtschaft (KTBL)) unter anderem das Problem der zu ge-

ringen Mengen landwirtschaftlicher Realdaten in der notwendigen Qualität, die für das Training von ML-Modellen essenziell sind. Hierbei wird versucht, die fehlende Menge an Realdaten durch Hinzunahme synthetisch erzeugter Daten im Bereich der landwirtschaftlichen Ertrags- und Wachstumsprognose auszugleichen. Mit agentenbasierten Simulationssystemen, in denen über Jahre gesammeltes Expertenwissen landwirtschaftlicher Prozesse integriert ist, lassen sich Daten mit Bezug zur Pflanzen-/Bodenentwicklung (Boden-/Pflanzencharakterisierung, Feldarbeitsprozesse, Zustand des Pflanzenwachstums und der biochemischen Vorgänge im Boden, Wetter und Klima) simulieren. Zwar sind diese Systeme nur bedingt für Simulationen der Prozesse realer Betriebe tauglich und aufgrund der Komplexität und des Aufwandes solcher Systeme an sich ständig ändernde reale Begebenheiten anzupassen, dennoch sind deren synthetisch generierte Daten wertvoll. Ein durch diese Daten trainiertes tiefes neuronales Netz ist in der Lage, die Fähigkeiten des Simulationssystems zu imitieren, wenn man die internen Berechnungsvorgänge außen vor lässt. Beide Systeme sind hier als Blackbox zu betrachten, welche aus gegebener Eingabe eine dazugehörige Ausgabe liefern. Der Vorteil bei der Verwendung von neuronalen Netzen liegt darin, dass es sich durch Transfer Learning, also das Anpassens mithilfe einiger wenigen Realdaten, an reale Bedingungen einfacher anpassen lässt. Im Projekt wurde zunächst die Eignung eines tiefen neuronalen Netzes erfolgreich getestet, und zwar in der Hinsicht, ob dieses Verfahren die Eingabe-/Ausgabefähigkeiten eines Simulationssystems (dem Multiagentensystem MPMAS_XN der Universität Hohenheim) imitieren kann (Muenzberg et al. 2022). Anschließend wurde evaluiert, ob es sich in spezifischen Bereichen durch Retraining anpassen lässt (Muenzberg et al. 2023). Als Technologie wurde ein Faltungsnetz, ein sogenanntes Convolutional Neural Network (CNN), eingesetzt, genauer ein 1D-Multistep-Ahead-CNN, das in der Lage ist, zeitliche Verläufe der Simulation zu erlernen und zu prognostizieren. Verwendet wird diese Technologie in einem hybriden Ansatz zusammen mit vorgeschalteten Techniken wie Wissensgraphen und Autoencodern, um Inkonsistenzen oder Lücken in den Realdaten zu korrigieren bzw. durch Interpolation zu schließen. Diese Techniken verbessern die Vorhersagequalität des CNN, da sie Expertenwissen in die Vorhersage einfließen lassen und notwendige fehlende Parameter einbringen.

7.5 Softwarearchitektur und Technologien

7.5.1 Fernerkundung

Ertragsvorhersagen innerhalb eines Feldes sind in der Praxis besonders interessant, innovativ und herausfordernd. So benötigt man Trainingsdaten in hoher räumlicher Auflösung. Ertragsdaten von Mähdreschern auf Teilfeldebene können hier verwendet werden (Shuai und Basso 2022; Pathak et al. 2023). Während der Ernte fährt der Mähdrescher mit Ertragssensoren durch das Feld und sammelt äquidistante und georeferenzierte Datenpunkte in hoher räumlicher Auflösung. Alle Datenpunkte sind durch verschiedene

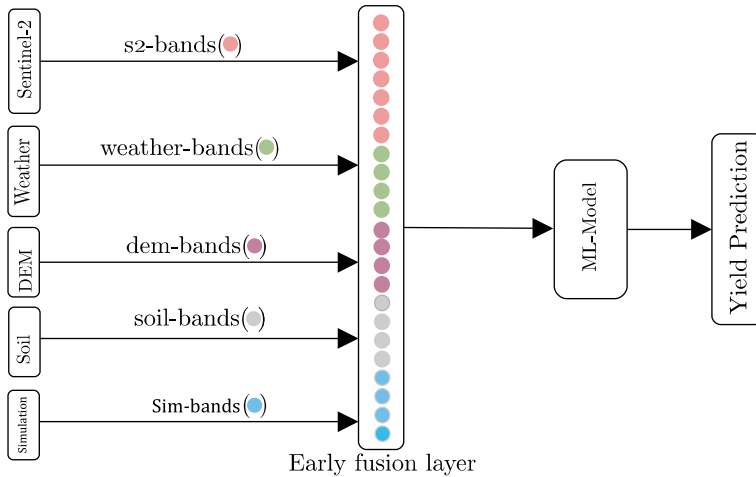


Abb. 7.3 Multimodale Ertragsprognose² Grafik basierend auf Pathak et al. (2023)

Merkmale charakterisiert, wie z. B. die geografische Koordinate und der gemessene Ertrag. Daraus resultierende Ertragskarten sind oft inhomogen, unterscheiden sich stark von Hersteller zu Hersteller und weisen oftmals eine Vielzahl von Messfehlern auf. Eine fachgerechte Vorverarbeitung ist daher besonders wichtig (Sanchez et al. 2023). Als Inputdaten für Ertragsmodelle haben sich Sentinel-2-(S2-)Satellitenbilder (L2A), gekoppelt mit Wetter, Boden und DEM-Daten, als praxisreif erwiesen. Da die verwendeten Daten oftmals unterschiedliche räumliche und zeitliche Auflösungen haben, müssen diese zunächst homogenisiert werden. In Pathak et al. (2023) wird eine effektive Early-Fusion-Methode vorgestellt. Hier werden Daten mit unterschiedlicher räumlicher Auflösung interpoliert und somit auf eine einheitliche Auflösung skaliert. S2-Daten dienen als Referenzpunkte und stellen jeweils ein Zeitintervall dar. Für jedes Zeitintervall werden weitere Modalitäten verknüpft. Zeitreihendaten, wie Wetterdaten oder Simulationsdaten, werden entlang der Zeitintervalle aggregiert. Die multivariate Zeitreihe stellt jeweils ein Rasterpixel der Ertragskarte dar und wird zum Trainieren eines KI-Modells verwendet. Abb. 7.3 stellt diesen Prozess schematisch dar.

Verschiedene Simulationsmodelle haben sich in der Praxis bewährt und können als zusätzliche Trainingsdaten verwendet werden. Hierzu zählen etwa WOFOST (Van Diepen et al. 1989), APSIM (Holzworth et al. 2014) und FAO-56 (Allen et al. 1998). Der Einsatz hängt stark von individuellen Bedürfnissen, Region und der Kulturart ab und sollte sorgfältig geplant werden. Durch den Einsatz von Simulationsmodellen ist es

²Mehrere Modalitäten mit unterschiedlichen räumlichen und zeitlichen Auflösungen werden auf Eingabeebene kombiniert. Ein maschinelles Lernmodell wird trainiert, um Ertragsprognosen in 10 m-Auflösung zu erstellen.

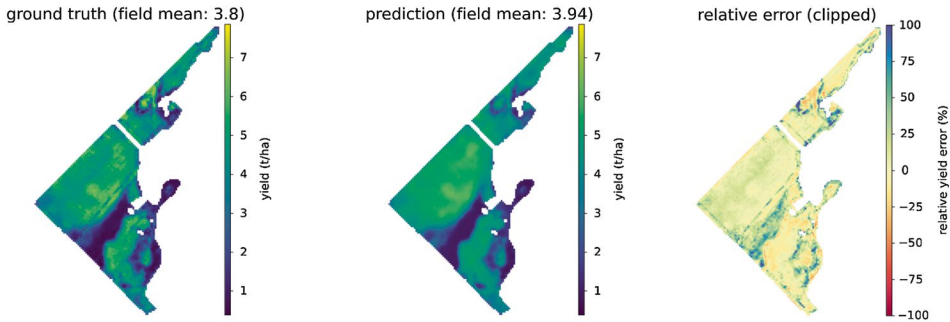


Abb. 7.4 Ertragsvorhersage eines Feldes (Pathak et al. 2023)³

möglich, sich auf verschiedene umweltbedingte Einschränkungen zu fokussieren, wie Temperatur, Licht, Wasser und Makronährstoffe. Besonders in trockenen Gebieten ist die Integration von Wasserstressfaktoren hilfreich. Alle genannten Simulationsmodelle sind für diesen Einsatz geeignet. In Abb. 7.4 wird exemplarisch eine Ertragskarte und deren Vorhersage dargestellt.

Gradient Boosting: Gradient Boosting Decision Trees (GBDT) werden in der Praxis häufig verwendet. Extreme Gradient Boosting (XGBoost; Chen et al. 2016) und Light-Gradient Boosting Machines (LightGBM; Ke et al. 2017) sind auf Gradient Boosting basierende Ensemble-Methoden. Diese Arten von Ensemble-Methoden machen sequenziell Vorhersagen und versuchen schwache Modelle zu kombinieren und aus ihren Fehlern zu lernen. Auf Gradient Boosting basierende Modelle sind besonders effizient und effektiv. Sie eignen sich daher gut für die Anwendung, da in vielen Bereichen der Landwirtschaft rechenstarke Systeme noch keinen Einzug gehalten haben. Auch sind diese besonders gut erklärbar und interpretierbar. Jede Entscheidung kann eindeutig nachvollzogen werden. Dies ist gegenüber herkömmlichen Methoden des maschinellen Lernens ein großer Vorteil, da diese oftmals als Blackbox-Systeme agieren. In systemkritischen und sensiblen Bereichen wie der Landwirtschaft spielen solche Eigenschaften eine wichtige Rolle. In der Praxis haben sich LightGBM-Modelle als praxisreif erwiesen und übertreffen im Bereich der Ertragsvorhersage andere GBDT-Modelle.

Rekurrente neuronale Netze: Neben Gradient-Boosting-Methoden spielen tiefe neuronale Netze eine wichtige Rolle, besonders rekurrente neuronale Netze (RNN). Diese zeichnen sich, im Gegensatz zu herkömmlichen neuronalen Netzen, durch zusätzliche Verbindungen zwischen Neuronen derselben und den vorhergegangenen Neuronen aus. Die rekurrente Verschaltung ermöglicht das Verarbeiten von zeitlichen Daten und

³Abgebildet sind Ertragsdaten von Soja in Argentinien, geerntet im Jahr 2021. Das Modell wurde mit multimodalen Trainingsdaten trainiert, basierend auf einem LSTM Modell. Links: Ertragskarte, Mitte: Ertragsvorhersage innerhalb eines Feldes, rechts: Streudiagramm zum Vergleich der Vorhersagen mit den gemessenen Ertragswerten.

somit das Entdecken von zeitlichen Abhängigkeiten. Long-short-Term-Memory-(LSTM-) Modelle sind eine Ableitung von RNN, welche Zeitreihendaten sequenziell verarbeiten können und dabei Informationen über einen längeren Zeitraum speichern. Sogenannte Gates kontrollieren den Informationsfluss und entscheiden darüber, welche Informationen gespeichert und welche vergessen werden. In der Landwirtschaft spielen zeitliche Abhängigkeiten eine besondere Rolle und somit erweisen sich RNN als besonders effektiv. So haben Temperaturen zeitlich gesehen einen unterschiedlich bedeutsamen Einfluss auf die Ertragsbildung. RNN sind durch ihre intrinsische Architektur in der Lage, diese Abhängigkeiten zu erlernen.

7.5.2 Transfer Learning

Das SimLearn-Konsortium erprobte erfolgreich, dass ein Convolutional Neural Network (CNN) in der Lage ist, die Vorhersageperformanz des Simulationssystems MPMAS_XN (Troost et al. 2020) bei gleicher Ein- und Ausgabe zu imitieren, indem es mit den Daten des Simulationssystems trainiert wird (Muenzberg et al. 2022). MPMAS_XN ist eine Kombination aus dem auf mathematischer Programmierung basierenden Multiagentensystemmodell (MPMAS), das optimale Produktions- und Investitionspläne für landwirtschaftliche Betriebe berechnet und deren wirtschaftliche Ergebnisse im Zeitverlauf dynamisch simuliert, und dem Boden-Pflanzen-Simulator Expert-N (XN), der das Wachstum von Pflanzen sowie Bodenprozesse im Zeitverlauf simuliert, einschließlich der tatsächlichen Erträge, die durch die von MPMAS ausgewählten Produktionspläne erzielt werden (Troost et al. 2020). Als externe Inputs stützt sich MPMAS_XN u. a. auf einen regelmäßig aktualisierten KTBL-Expertendatensatz mit technischen Merkmalen von Tausenden von Feldoperationen und Maschinen- und Gerätetypen, täglichen Zeitreihen beobachteter oder simulierter Wetterdaten und mehrdimensionalen Bodenprofilen. MPMAS_XN erzeugt verschiedene Arten von Ausgaben, darunter multivariate Daten auf Betriebsebene in relationalen Datenstrukturen (z. B. optimale Produktions- und Investitionspläne für einen bestimmten Betrieb) und Zeitreihen auf Parzellenebene, d. h. multivariate Zeitreihen mit täglicher Auflösung für den Pflanzenstatus (Biomasse, Phänologie, Reife, Stickstoffgehalt usw.), den Bodenstatus (Feuchtigkeit, Kohlenstoffgehalt usw.) und die Bodenqualität (Feuchtigkeit, Kohlenstoffvorräte, Stickstoff usw.), der angepassten Bewirtschaftung während der Saison (Feldarbeitstermine, Düngemengen) und der endgültigen Erträge für einen bestimmten Bewirtschaftungsplan, des anfänglichen Bodenzustands und der exogenen Wetterzeitreihen (u. a. Temperatur, Niederschlag, Feuchtigkeit und Windgeschwindigkeit).

Für den Transfer-Learning-Prozess wurden Realdaten eines experimentellen Betriebes der Universität Hohenheim bereitgestellt. Diese wurden zunächst semistrukturiert als Excel-Tabelle zusammengefasst. Die Datensätze enthalten Informationen über die jeweiligen Felder und Parzellen sowie Feldarbeitsprozessdaten. Des Weiteren wurden Daten mit Informationen über Boden- und Pflanzencharakterisierung sowie Daten von ver-

schiedenen Wetterstationen hinzugefügt. Um die Daten in eine einheitliche und strukturierte Form zu überführen und für Training und Vorhersage durch das CNN anwendbar zu machen, wurden die Daten analysiert, neu gelabelt und durch R2RML-Mapping in einen Wissensgraphen im Resource-Description-Framework-(RDF-)Format überführt (Reinosch et al. 2023). In dieser Struktur können die benötigten Datenkomponenten mithilfe der SQL-ähnlichen Abfragesprache SPARQL selektiert werden – speziell für Wissensgraphen. Durch Hinzufügen von weiterem Expertenwissen aus dem KTBL-Bestand können Inkonsistenzen aufgedeckt und fehlende Werte ermittelt werden (Reinosch et al. 2023). Die Daten werden nach dem Selektieren in die gleiche Form wie die simulierten Daten überführt. Einzelne Werte wie Stickstoffzusammensetzung von Düngemitteln oder die Saatgutmenge (Körneranzahl) pro Kilogramm werden mithilfe des KTBL-Datenbestandes errechnet (Muenzberg et al. 2023).

Fehlende Boden- und Pflanzenwachstumzustände in den Daten werden durch die Verwendung eines Autoencoders interpoliert (Muenzberg et al. 2023). Abb. 7.5 beschreibt den Trainings- und Vorhersageprozess in grafischer Darstellung: Nach Training

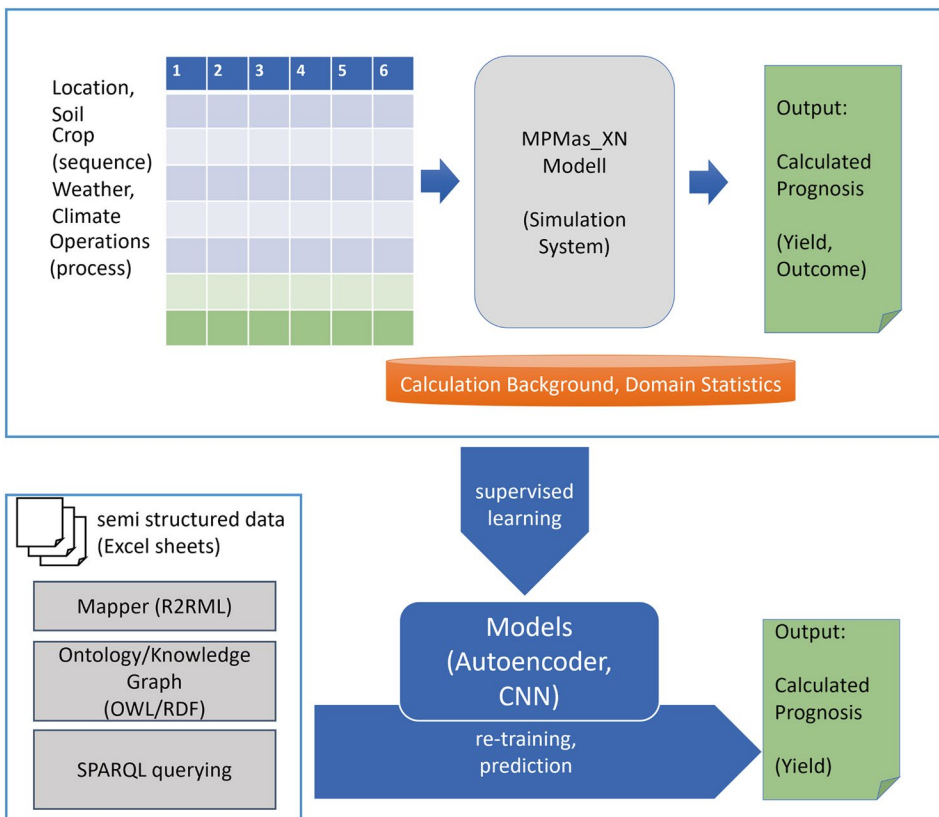


Abb. 7.5 Transfer-Learning-Prozess

des CNN und des Autoencoders anhand von Daten des MPMAS_XN-Simulationssystems wird das CNN im Sinne des Transfer Learning mit selektierten Realdaten angepasst (retrainiert). Der Autoencoder interpoliert vorab die fehlenden Datenelemente. Das angepasste CNN ist nun in der Lage, genauere Vorhersagen für weitere betriebsspezifische Realdaten zu liefern als das ursprünglich trainierte Netz. Dieser Sachverhalt wurde in Muenzberg et al. (2023) evaluiert.

Im Wesentlichen wurden die nachfolgend beschriebenen Technologien für das beschriebene Verfahren angewendet.

1D-Multistep-Ahead-CNN: Bisherige Veröffentlichungen haben gezeigt, dass sich sogenannte Multistep-Ahead-Modelle zur Vorhersage von Zeitreihen eignen. Diese liefern in vielen Ansätzen gleichwertige oder bessere Ergebnisse als Recurrent Neural Networks (RNN) (Chattha et al. 2019). Durch die Anwendung von Multistep-Ahead CNN wird zugleich der Vorteil der Faltungsnetze eingebunden, da diese durch ihre Faltungsoperationen in der Lage sind, wichtige korrelierende Datenmerkmale in großen Datenmengen mit vielen verschiedenen Variablen zu erfassen und zusammenzufassen, was das ganze Netz performanter im Hinblick auf Schnelligkeit und Ressourcenschonung macht. Die Faltungsoperation tastet Merkmalsbereiche ab und fasst diese durch Pooling zusammen (Muenzberg et al. 2022). Anschließend werden die dimensionierten Werte durch in künstlichen neuronalen Netzen übliche verdeckte Schichten verknüpft, in denen die Werte künstliche Neuronen mit einer Aktivierungsfunktion durchlaufen. Eine detaillierte Funktionsweise von CNN wird in O’Shea und Nash (2015) beschrieben.

Im Multistep-Ahead-Verfahren werden Zeitreihendaten vorab in aufeinanderfolgende Zeitbereiche zusammengefasst (Chandra et al. 2021), wobei bei der Vorhersage der Output bereits vorhergesagter Zeitschritte immer wieder iterativ als Input in die Vorhersage einfließt (Sliding-Window-Verfahren). So wird eine bestimmte Anzahl an Zeitreihentagen zusammengefasst (die Anzahl von 4 zusammengefassten Tagen hat sich im gegebenen Fall als optimal erwiesen) und der nachfolgende Tag wird vorhergesagt. Anschließend fließt der vorhergesagte Tag wieder als Input in die Vorhersage des nachfolgenden Tages ein. Das rekursive Verfahren wird fortgeführt, bis die gewünschte Anzahl an vorhergesagten Zeitschritten erreicht ist. Im Falle dieses Projektes sind dies die Tage von der Aussaat bis zur Ernte. Abb. 7.6 stellt diesen Prozess grafisch dar. Jeder vorhergesagte Zeitbereich durchläuft bei der Prognose das CNN. Zum Training wurden die Trainingsdaten entsprechend als Input- und Outputvektoren transformiert.

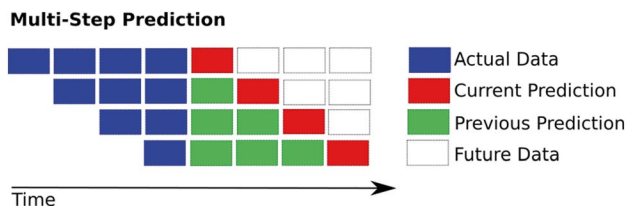


Abb. 7.6 Sliding-Window-Verfahren

Autoencoder: Autoencoder besitzen die Fähigkeit, eine Anzahl von Eingabedaten zu komprimieren (Encoding) und somit die Größe an Daten zu reduzieren bzw. die Anzahl von Datenattributen (Features) zu reduzieren. Dabei sollen die wichtigen Merkmale erhalten bleiben. Durch Decoding werden anschließend die dimensionierten Daten auf ihre ursprüngliche Größe der Eingabedaten zurückgeführt bzw. die Daten werden so dekomprimiert, dass sie den ursprünglichen Eingabedaten sehr ähnlich sind. Wir verwenden Autoencoder, um durch das Encoding/Decoding fehlender (auf 0 gesetzter) Werte zu interpolieren. In Bhadani (2021) wird dieses Verfahren näher beschrieben.

RDF Knowledge Graphs: Maschinenlesbare Wissensgraphen unter Verwendung des Resource Description Frameworks (RDF) werden angewendet, um dem ML-Prozess semantisch aufbereitete Daten für die einfachere Interpretation zu liefern (Reinosch et al. 2023). Als syntaktisches Zielformat wird die Terse RDF Triple Language (Turtle) genutzt (Prud'hommeaux und Carothers 2012). Durch den experimentellen Betrieb Lindenhöhe wurden reale Betriebsdaten im MS-Excel-Format geliefert, deren Tabellenblätter durch die standardisierte Mapping-Sprache R2RML (Das et al. 2012) und zugehörige Prozessierungswerkzeuge ausgelesen und in Wissensgraphen transformiert. Dieser Vorgang bietet den Vorteil, dass Änderungen einfach zu handhaben sind und dass Domänenexperten ohne großes Wissen über die Graphentechnologie und Ontologien unter Anleitung des Entwicklers mit bekannten Werkzeugen wie Microsoft Excel arbeiten können (Das et al. 2012). Zur Übertragung von tabellarischen Daten nach RDF wurden Regeln in R2RML formuliert und mit einer R2RML Engine (R2RML-F; Debruyne und O'Sullivan 2016) ausgeführt. Das Wissen der Daten aus der Excel-Datei kann somit einfach maschinell selektiert werden. Auf diese Art können sehr große Datenmengen mit einem vergleichsweise überschaubaren R2RML Mapping in einen Graphen übersetzt werden, dessen Inhalte von Tabellenspalten dabei als Subjekt und Objekt mit festgelegten Prädikaten verknüpft werden (Das et al. 2012). Somit wird eine maschinell lesbare Bedeutung (Semantik) der Daten modelliert. Für die Abfrage der Ontologie wird ein SPARQL-Endpoint (SPARQL Protocol and RDF Query Language) verwendet. Zusätzlich zu den betrieblichen Realdaten wurde landwirtschaftliches Wissen mit der RDF-Graphentechnologie modelliert. Dies sind z. B. Informationen über die Zusammensetzung von Stickstoff in verwendeten Düngern oder die Zusammensetzung von Böden (schluffige, lehmige oder sandige Anteile). Fehlende oder inkonsistente Werte in den genannten Excel-Dateien können somit durch Verknüpfungen ermittelt und korrigiert werden. Auch hier wurden geeignete Abfragen mittels SPARQL erstellt.

7.6 Empfehlungen

Durch Erkenntnisse, die bei der Entwicklung und Evaluierung verschiedener Technologien für die Wachstums- und Ertragsprognose in der Landwirtschaft erzielt wurden, konnten die nachfolgend beschriebenen Empfehlungen abgeleitet werden:

1. Durch Kombination von ML-Verfahren und ausreichend vorhandenem Expertenwissen können vorab trainierte Vorhersagemodelle auf unvollständige oder teilweise inkonsistente Daten angewendet werden.
2. Die Auswahl eines geeigneten Simulationsmodells ist von grundlegender Bedeutung und bedarf besonderer Aufmerksamkeit.
3. Simulationsmodelle sind oft ungenau und müssen daher ausreichend kalibriert werden.
4. Autoencoder sind geeignet, Lücken in Daten durch Interpolation zu schließen.
5. 1D-Multistep-Ahead-CNN sind einsetzbar bei der Prognose großer Zeitreihendaten mit vielen Features.
6. Mit Wissensgraphen können semistrukturierte Daten in ein strukturiertes und mit Metainformationen verknüpft, maschinenlesbares Format transformiert werden.

7.7 Fazit

Es wurden zwei Verfahren zur landwirtschaftlichen Wachstums- und Ertragsprognose vorgestellt. Im SimLearn-Projekt wird die Fähigkeit eines Simulationssystems, aus ausgewählten (symbolischen) Eingabedaten durch integriertes Expertenwissen eine symbolische Ausgabe zu generieren, im Gesamten durch ML-Verfahren reproduziert. Dabei wird erzieltes Wissen in Form von RDF-Graphen aufbereitet. Im Ansatz der Ertragsvorhersagen mittels Fernerkundung werden gesammelte Rohdaten zur Vorhersage durch ML-Verfahren verwendet. Zusätzlich fließen physikalische Gesetzmäßigkeiten als Expertenwissen in symbolischer Form in diese Verfahren ein, um die Vorhersagen weiter zu optimieren.

Beide Projekte zeigen erfolgreiche hybride Ansätze und sind im Kern ähnlich, auch wenn sie im Detail ihrer Architektur und Technologien signifikante Unterschiede aufweisen. Beide Verfahren integrieren Expertenwissen, sowohl aus Simulationssystemen als auch aus weiteren Quellen, in ihre Prozesse. Durch das zusätzliche Wissen wird somit der Umstand fehlender, für ML-Prozesse jedoch notwendiger Daten ausgeglichen. Dieses Wissen fließt, u. a. mit Rohdaten, als symbolische Repräsentation sowohl in das Training als auch in anschließende Prognosen der ML-Verfahren ein. Die Verfahren können fortlaufend durch strukturierte Daten mittels Transfer-Lernen angepasst werden und sind somit fähig, betriebspezifische Vorhersagen zu treffen.

Literatur

- Bhadani, R. (2021): AutoEncoder for Interpolation. arXiv, <https://arxiv.org/abs/2101.00853>.
- Chandra, R., Goyal, S., Gupta, R. (2021): Evaluation of deep learning models for multi-step ahead time series prediction. arXiv Computing Research Repository (CoRR), <https://arxiv.org/abs/2103.14250>.

- Chattha, M. A., Siddiqui, S. A., Munir, M., Malik, M. I., van Elst, L., Dengel, A., Ahmed, S. (2019): DeepEX: Bridging the Gap Between Knowledge and Data Driven Techniques for Time Series Forecasting. International Conference on Artificial Neural Networks (ICANN-2019), September 17–19, Munich, Germany.
- Das, S., Sundara, S., Cyganiak, R. (2012): R2RML: RDB to RDF Mapping Language, <https://www.w3.org/TR/r2rml/>, accessed 24.10.2022.
- Debruyne, C., O’Sullivan, D. (2016): R2RML-F: Towards Sharing and Executing Domain Logic in R2RML Mappings. In: Proceedings of the Workshop on Linked Data on the Web, LDOW 2016, co-located with the 25th International World Wide Web Conference (WWW 2016), Montreal, Canada.
- Muenzberg, A., et al. (2022): Machine Learning on Simulated and Real Farm Data Based on an Ontology-Controlled Data Infrastructure. In (Martin, A., et al., ed.): Proceedings of the AAAI 2022 Spring Symposium on Machine Learning and Knowledge Engineering for Hybrid Intelligence (AAAI-MAKE 2022). <http://ceur-ws.org/Vol-3121/paper18.pdf>, Stanford University, Palo Alto, California, USA.
- Muenzberg, A., Troost, C., Bernardi, A. (2023): Landwirtschaftliche Ertragsvorhersage im Kontext begrenzter realer Trainingsdatensätze: ein Transfer-Learning-Ansatz unter Verwendung tiefer neuronaler Netze. In: Demmler, D., Krupka, D., Federrath, H. (Eds.), 4. Workshop KI in der Umweltinformatik der 53. Jahrestagung der Gesellschaft für Informatik – Informatik in den Naturwissenschaften (INFORMATIK-2023), Berlin, Germany.
- O’Shea, K., Nash, R. (2015): An introduction to convolutional neural networks, arXiv preprint [arXiv:1511.08458](https://arxiv.org/abs/1511.08458).
- Prud’hommeaux, E., Carothers, G. (2012): Turtle: Terse RDF Triple Language, <http://www.w3.org/TR/2012/WD-turtle-20120710/>, accessed: 24.10.2022.
- Reinosch, N., Münzberg, A., Martini, D., Niehus, A., Seuring, L., Troost, C., Srivastava, R. K., Berger, T., Streck, T., Bernardi, A. (2023): SIMLEARN – Betriebliche Entscheidungsunterstützung durch ontologiegestützte Integration von Simulationsmodellen, Systemen für maschinelles Lernen und Planungsdaten. In: 43. Jahrestagung der Gesellschaft für Informatik in der Land-, Forst- und Ernährungswirtschaft e. V. (GIL-Jahrestagung 2023), Osnabrück, Germany, Februar 13–14.
- Troost, C., et al. (2020): The Bioeconomic Modelling System MPMAS_XN: Simulating Short and Long-term Feedback Between Climate, Crop growth, Crop Management and Farm Management. In: van Griensven, A., Nossent, J., Ames, D.P. (Eds.) 10th International Congress on Environmental Modelling and Software, Brussels, Belgium.
- Jeong, S., Ko, J., Shin, T. and Yeom, J-m. (2022) “Incorporation of machine learning and deep neural network approaches into a remote sensing-integrated crop model for the simulation of rice growth,” Scientific Reports, vol. 12, no. 1, pp. 9030.
- Shahhosseini, M., Hu, G., Huber, I. and Archontoulis, S.V. (2021) “Coupling Machine learning and crop modeling improves crop yield prediction in the us corn belt,” Scientific reports, vol. 11, no. 1, pp. 1–15.
- D. Pathak, M. Miranda, F. Mena, C. Sanchez, P. Helber, B. Bischke, P. Habelitz, H. Najjar, J. Sidamsetty, D. Arenas, M. Vollmer, M. Charfuelan, M. Nuske, and A. Dengel (2023) “Predicting Crop Yield with Machine Learning: An Extensive Analysis of Input Modalities and Models on a Field and Sub-Field Level,” in IGARSS 2023–2023 IEEE International Geoscience and Remote Sensing Symposium, 2023, pp. 2767–2770.
- C. Sanchez, D. Pathak, M. Miranda, P. Helber, B. Bischke, P. Habelitz, H. Najjar, F. Mena, J. Sidamsetty, D. Arenas, M. Vollmer, M. Charfuelan, M. Nuske and A. Dengel (2023) “Influence of data cleaning techniques on sub-field yield predictions,” in IGARSS 2023–2023 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2023, pp. 4852–4855.



- Van Diepen, C.A., Van Wolf, J., Van Keulen, H. and Rappoldt, C. (1989) “Wofost: a simulation model of crop production,” *Soil use and management*, vol. 5, no. 1, pp. 16–24.
- Allen, R.G., Pereira, L.S., Raes, D., Smith, M., et al. (1998) “Crop evapotranspiration-guidelines for computing crop water requirements-fao irrigation and drainage paper 56,” *Fao, Rome*, vol. 300, no. 9, pp. D05109, 1998.
- Holzworth, D.P., Huth, N.I., deVoil, P.G., Zurcher, E.J., Herrmann, N.I., McLean, G., Chenu, K., van Oosterom, E.J., Snow, V., Murphy, C., et al. (2014) “Apsim–evolution towards a new generation of agricultural systems simulation,” *Environmental Modelling & Software*, vol. 62, pp. 327–350.
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T-Y. (2017) “Lightgbm: A highly efficient gradient boosting decision tree,” *Advances in neural information processing systems*, vol. 30.
- Chen, T. and Guestrin, C. (2016) “Xgboost: A scalable tree boosting system,” in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.
- Shuai G. and Basso, B. (2022) “Subfield maize yield prediction improves when in-season crop water deficit is included in remote sensing imagery-based models,” *Remote Sensing of Environment*, vol. 272, pp. 112938.
- Von Rueden, L., Mayer, S., Beckh, K., Georgiev, B., Giesselbach, S., Heese, R., Kirsch, B., Pfrommer, J., Pick, A., Ramamurthy, R. and Walczak, M. (2021) Informed machine learning—a taxonomy and survey of integrating prior knowledge into learning systems. *IEEE Transactions on Knowledge and Data Engineering*, 35(1), pp. 614–633.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Alexander Zender  und Bernhard G. Humm 

Kernaussagen

1. Effektive und effiziente Machine-Learning-(ML-)Modelle zu entwickeln ist komplex und erfordert große Expertise.
2. Automated Machine Learning (AutoML) automatisiert die Suche nach optimierten ML-Modellen. AutoML ist ein aktives Forschungsfeld; die Entwicklung von State-of-the-Art-AutoML-Lösungen unterliegt einer hohen Dynamik.
3. Meta AutoML ist ein innovatives Konzept zur Steigerung der Effektivität von ML-Modellen sowie zur Vermeidung von Vendor-Lock-in einzelner AutoML-Lösungen.
4. OMA-ML (Ontology-based Meta AutoML) ist eine Plattformimplementierung von Meta AutoML mit dem Ziel, effektive und effiziente ML-Modelle möglichst einfach zu entwickeln. Die Plattform wendet sich sowohl an ML-Experten als auch an Fachexperten einer Anwendungsdomäne wie Medizin.
5. Methoden der wissensbasierten KI, insbesondere Ontologien und Regelsysteme, können eingesetzt werden, um die Erstellung von ML-Modellen zu optimieren.
6. Eine ML-Ontologie stellt umfangreiches Wissen über das Gebiet von ML bereit. Sie formalisiert ML-Teilgebiete, -Verfahren, -Metriken und -Lösungen und kann vielseitig eingesetzt werden, z. B. bei OMA-ML zur Unterstützung von Anwendern bei der interaktiven Konfiguration und Hilfe.

A. Zender (✉) · B. G. Humm
Hochschule Darmstadt, Darmstadt, Deutschland
E-Mail: alexander.zender@h-da.de

B. G. Humm
E-Mail: bernhard.humm@h-da.de

7. Ein Regelsystem in Kombination mit einer Blackboard-Architektur kann zur Optimierung von Prozessschritten eingesetzt werden, z. B. bei OMA-ML zur Optimierung von ML-Trainingsläufen und damit zur Reduktion von Energieverbrauch und Umweltbelastung.

8.1 Einleitung

Machine Learning (ML) ist in aller Munde als derzeit prominentestes Teilgebiet der künstlichen Intelligenz (KI). Zahlreiche ML-Anwendungen sind im täglichen Einsatz, von digitalen Assistenten über Entscheidungsunterstützungssysteme bis hin zu teilautonomen Fahrzeugen. Auch werden Datenanalysen in zahlreichen Unternehmen ML-basiert durchgeführt. Aber effektive und effiziente ML-Anwendungen zu entwickeln ist höchst anspruchsvoll und erfordert eine große Expertise.

In diesem Kapitel stellen wir die Plattform *OMA-ML (Ontology-based Meta AutoML; Humm und Zender 2021)* vor, mit dem Claim: “Effective machine learning made easy!” (effektives Machine Learning einfach gemacht). Ziel dieser Plattform ist es, die Hürden für den Einsatz von ML zu senken bei gleichzeitig hoher Effektivität und Effizienz.

Aber was hat das mit hybrider KI zu tun? Interessanterweise setzen wir wissensbasierte KI ein, um die Effektivität und Effizienz von ML zu erhöhen. Insbesondere setzen wir einen Wissensgraphen (ML Ontology) für die Konfiguration von ML ein sowie ein Regelsystem mit Blackboard-Architektur, um Strategien für Ressourceneffizienz umzusetzen.

Abb. 8.1 ordnet dieses Kapitel thematisch ein bezüglich der Branche, Anwendung, Daten sowie Technologien.

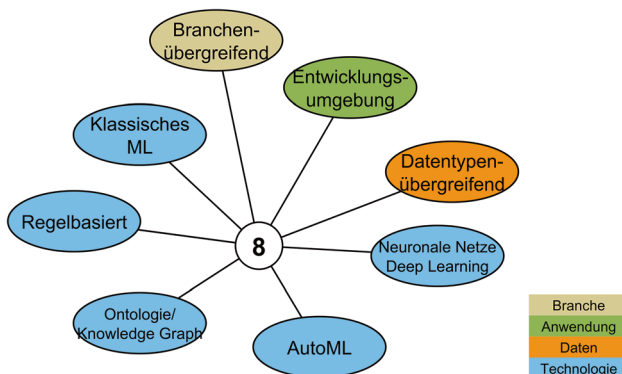


Abb. 8.1 Thematische Einordnung des Kap. 8

OMA-ML ist eine Webanwendung, die zwei Anwendergruppen adressiert: (a) *ML-Experten* haben KI- und Programmierkenntnisse und erwarten von OMA-ML Unterstützung bei der Entwicklung effektiver ML-Anwendungen. (b) *Fachexperten* in einer Anwendungsdomäne, z. B. Medizin, möchten ihre Daten analysieren und einfache Vorhersagen treffen. Ein häufig referenziertes Anwendungsbeispiel ist die Vorhersage von Krankheiten wie Diabetes auf der Basis von Patienteneigenschaften wie Geschlecht, Alter, Blutzucker etc. (Miao 2021). Wir verwenden diesen Anwendungsfall beispielhaft in diesem Kapitel.

OMA-ML erlaubt das Hochladen von Datensätzen, z. B. mit Patienteninformationen und Angaben zu Diabetes-Erkrankungen auf die Plattform. Mittels ML können Vorhersagemodelle interaktiv mittels ML-Training erstellt werden. Abb. 8.2 zeigt einen Screenshot mit dem Wizard zur Konfiguration des ML-Trainings für den Diabetes-Datensatz.

Der Fachexperte braucht nur drei Angaben zum Start des ML-Trainings zu machen: (a) die eigentliche Aufgabe (ML Task): “tabular classification”, d. h. die Klassifikation von Patienten auf der Basis tabellarischer Daten (hier bezüglich einer Diabetes-Erkrankung); (b) das Ziel der Vorhersage (Target): die Spalte “Outcome” der Tabelle; (c) die maximale Trainingszeit (Runtime): 3 min.

Home / Datasets / Dataset diabetes-Dataset.csv / Train

Train

Configuration

Save Schema PREVIOUS FINISH

Dataset preparation Configuration

1 ML task *
Tabular classification

2 Target *
Outcome

3 Maximum runtime [minute]
3

Detail configuration (optional)

Perform model analysis

ML libraries

Scikit-learn CatBoost LightGBM XGBoost Pytorch MXNET Keras Prophet

AutoML solutions

Autogluon EvalML FLAML GAMA MLJAR Pycaret TPOT LAMA Autokeras

Training strategies

Top 3 Models Multi Fidelity PCA Feature Extraction

LightGBM, short for light gradient-boosting machine, is a free and open-source distributed gradient-boosting framework for machine learning, originally developed by Microsoft. It is based on decision tree algorithms and used for ranking, classification and other machine learning tasks. The development focus is on performance and scalability. [More...](#)

Abb. 8.2 Interaktive Tooltip-Hilfe am Beispiel der ML-Bibliothek LightGBM

Der untere Bereich des Training-Wizards bietet optionale Parameter an. Diese Parameter werden erst sichtbar, wenn der Detailbereich aufgeklappt ist. Der ML-Experte kann hier eine große Anzahl von Detailkonfigurationen vornehmen, wie ML-Bibliotheken (ML Libraries), AutoML-Lösungen (AutoML Solutions – siehe Folgeabschnitt), Strategien (Enabled Strategies) und dutzende weitere Konfigurationen. Hier sieht man bereits den Einsatz wissensbasierter KI: Welche Konfigurationen für eine ML-Aufgabe sinnvoll möglich sind, wird aus einer ML-Ontologie gelesen: eine formale Repräsentation von Wissen über ML-Begriffe und deren Beziehungen (Humm et al. 2021). Zusatzinformationen werden aus einem verknüpften Wissensgraphen (hier: WikiData¹) herangezogen. Wenn der Nutzer beispielsweise über das Feld mit der ML-Bibliothek *Keras* fährt, wird ein Tooltip mit weiteren Informationen zu Keras geöffnet (siehe Screenshot). Durch die interaktive Hilfe erhalten Nutzer schnellen und einfachen Zugriff auf Informationen zu Begriffen, die ihnen unbekannt sind.

Schließlich können Trainingsstrategien aktiviert werden, um die Effizienz des Trainings zu steigern und so Energie zu sparen (Zender und Humm 2023). Im Beispiel von Abb. 8.3 wird die *Top-3-Strategie* angeboten. Bei dieser wird das Training in zwei Phasen aufgeteilt. In der ersten Phase suchen alle AutoML-Lösungen nach dem besten ML-Modell; dabei werden eine reduzierte Trainingszeit und Trainingsdatenmenge verwendet. In der zweiten Phase trainieren die besten drei AutoML-Lösungen aus der ersten Phase je ein neues ML-Modell mit der verbleibenden Zeit und allen Trainingsdaten.

Nach Abschluss des Trainings werden dem Nutzer die besten ML-Modelle in Form eines Dashboards dargestellt (siehe Screenshot in Abb. 8.3)

Auf diesem Dashboard werden dem Nutzer Informationen über ein trainiertes ML-Modell bereitgestellt, wie die verwendete ML-Bibliothek und das ML-Verfahren, auf dem das ML-Modell basiert. Mediziner können das ML-Modell innerhalb von OMA-ML für Vorhersagen über neue Patienten nutzen; ML-Experten können das ML-Modell als ausführbaren Python-Code herunterladen und in eine KI-Anwendung für Mediziner integrieren. Darüber hinaus werden die Analyseergebnisse des ML-Modells visualisiert, beispielsweise durch ein *Feature-Importance*-Diagramm, das den Einfluss einzelner Kenngrößen auf die Vorhersage zeigt, was sowohl für Mediziner als auch für ML-Experten interessant sein kann.

Abb. 8.4 zeigt das Boxology-Modell² des hybriden KI-Systems. OMA-ML als ML-Plattform führt klassisches Machine Learning durch, mit Daten als Input und einem ML-Modell als Output. Der Trainingsprozess an sich wird konfiguriert durch den Benutzer; die Konfiguration wird unterstützt durch wissensbasierte KI, insbesondere die ML Ontologie. Der Trainingsprozess selbst kann mittels Trainingsstrategien regelbasiert optimiert werden.

¹ <https://www.wikidata.org>

² Für eine Einführung in die Boxology siehe Kap. 1.

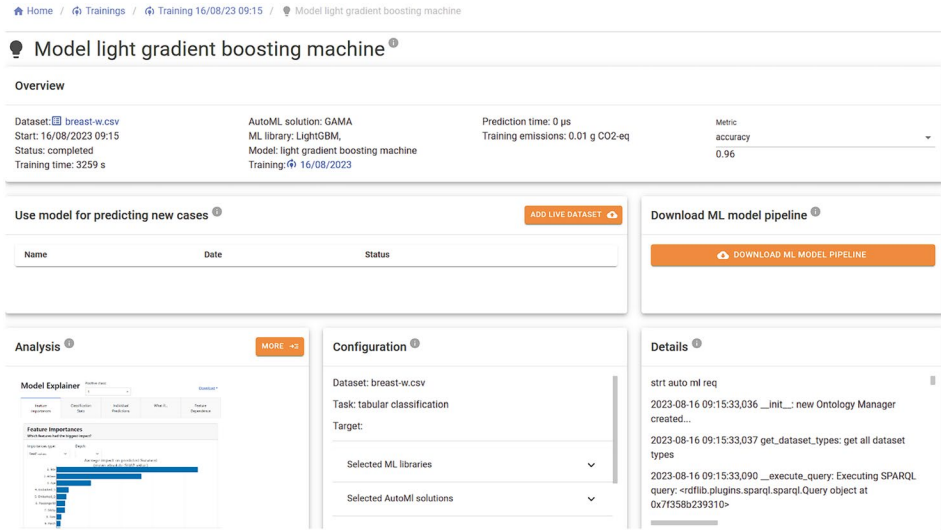


Abb. 8.3 OMA-ML-Modell-Dashboard

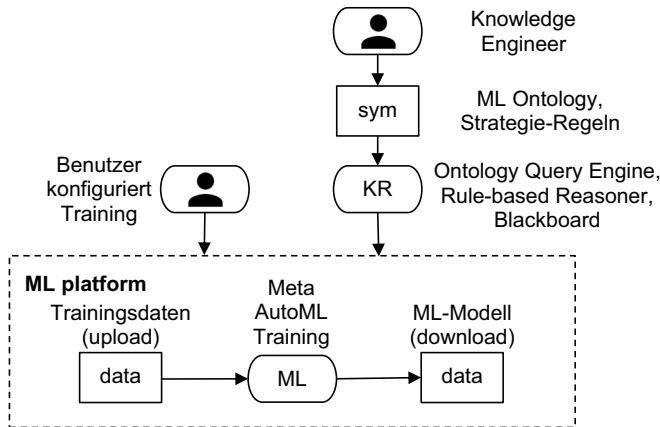


Abb. 8.4 Boxology-Modell des hybriden KI-Systems

Im weiteren Verlauf des Kapitels werden die Grundlagen von Machine Learning, AutoML und eine Übersicht von über 30 verfügbaren AutoML-Lösungen präsentiert. Wir stellen auch das Konzept von Meta AutoML vor, welches die Grundlage der Plattform OMA-ML ist. Anschließend werden die verschiedenen wissensbasierten KI-Komponenten von OMA-ML erläutert. Die Architektur und Technologien von OMA-ML werden ebenfalls behandelt. Abschließend zeigen wir anhand einer Evaluation auf, wie der

Abb. 8.5 ML Workflow und Anwendung

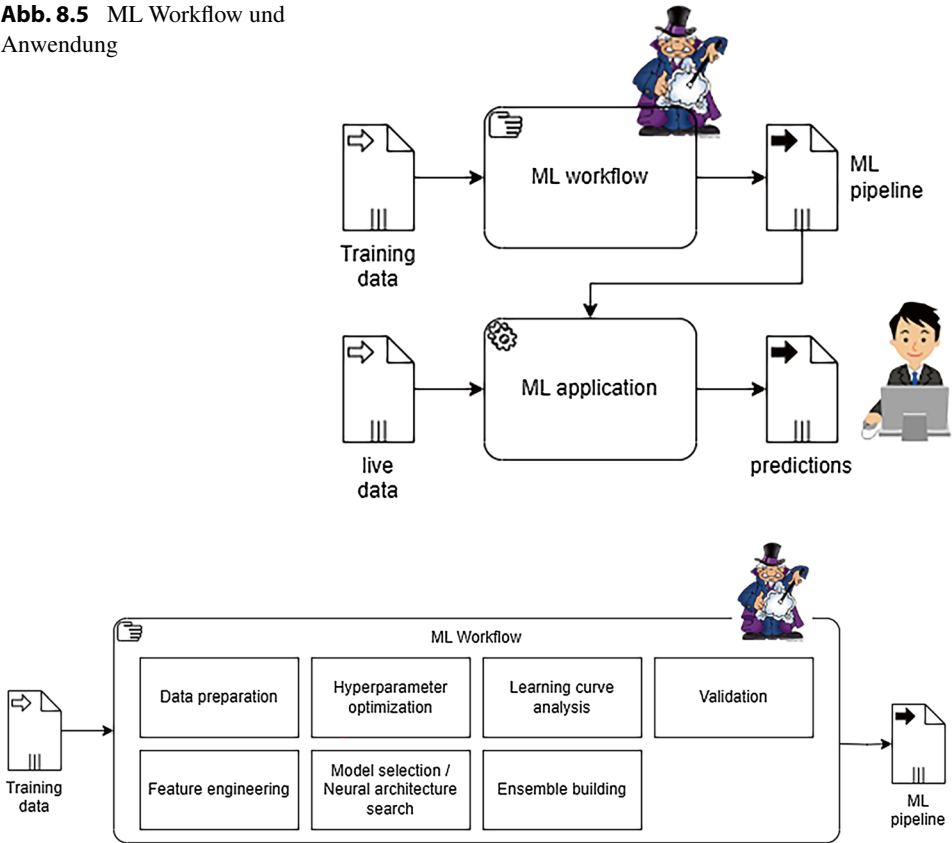


Abb. 8.6 Schritte im ML Workflow

Einsatz wissensbasierter KI die Trainingseffizienz steigern kann. Wir beschließen das Kapitel mit Empfehlungen für die Entwicklung hybrider KI-Anwendungen basierend auf unseren Erfahrungen mit OMA-ML und einem Fazit.

8.2 Machine Learning und AutoML

Im folgenden Abschnitt führen wir den traditionellen Workflow in ML-Projekten ein, dessen Automatisierung durch Automated Machine Learning (AutoML) und die Weiterentwicklung von AutoML: Meta AutoML.

8.2.1 ML Workflow

Der *Machine Learning Workflow* (ML Workflow) ist ein Prozess, mit dem ein ML-Experte aus *Training Data* (Trainingsdaten) eine *ML Pipeline* erstellt. Eine Übersicht dieses Ablaufs ist in Abb. 8.5 dargestellt. Die Trainingsdaten sind eine Sammlung von Beobachtungen von Phänomenen der realen Welt (Zheng und Casari 2018). Bei der Vorhersage einer Diabetes-Erkrankung sind das beispielsweise eine Sammlung von Patientendaten wie Geschlecht, Alter, BMI, Blutzucker sowie Informationen über eine Diabetes-Erkrankung. Dabei wird zwischen *Features* (Eingangsparameter) und dem *Target* (Vorhersageziel) unterschieden. Bei dem vorhergehenden Beispiel wären die *Features* Geschlecht, Alter, BMI, Blutzucker und das *Target* die Diabetes-Diagnose. Eine ML Pipeline umfasst das eigentliche ML-Modell sowie das notwendige *Preprocessing* (Datenvorbereitung) und *Postprocessing* (Datennachbereitung). Das ML-Modell kondensiert Wissen aus den Trainingsdaten in numerischer Form. Sobald eine ML Pipeline erstellt und getestet ist, kann diese in eine *ML Application* (ML-Anwendung) integriert werden und kann *Predictions* (Vorhersagen) auf neue *Live Data* (Daten aus dem Tagesgeschäft) treffen.

Der ML Workflow ist ein komplexer Prozess, bestehend aus mehreren Schritten, die häufig auch iterativ durchgeführt werden. Um die bestmögliche ML Pipeline zu finden, greift der ML-Experte auf umfangreiches Wissen und Erfahrung zurück. Abb. 8.6 zeigt wesentliche Schritte im ML Workflow auf.

Der erste Schritt ist die *Data Preparation*. Dabei werden die Trainingsdaten gesäubert und aufbereitet, damit diese von ML-Verfahren verarbeitet werden können (Potdar und Pai 2017). Zum Beispiel werden fehlende Daten behandelt oder inkompatible Datentypen transformiert. Im Beispiel der Diabetes-Vorhersage macht ein Feature Angaben zu einer Vorerkrankung bei den Eltern mit 3 Ausprägungen: nein, ein Elternteil, beide Elternteile. Diese sind als Zeichenketten (Strings) repräsentiert, mit denen allerdings viele ML-Verfahren nicht umgehen können. Mittels Verfahren wie One-Hot-Encoding können diese in numerische Werte transformiert werden (Potdar und Pai 2017).

Im nächsten Schritt, dem *Feature Engineering*, werden die Trainingsdaten optimiert. Dadurch soll die Trainingsdauer reduziert (Garcia et al. 2015) sowie die Effektivität (Jain und Zongker 1997) des ML-Modells gesteigert werden. Dabei können verschiedene Ansätze angewendet werden, beispielsweise *Feature-Extraktion* oder *Dimensionsreduktion*. Beide Ansätze basieren auf demselben Prinzip: Es wird versucht, die Komplexität der Trainingsdaten zu reduzieren. Bei der Feature-Extraktion werden aus bestehenden *Features* neue, bedeutsamere *Features* generiert (Khalid et al. 2014). Bei der Dimensionsreduktion wird die Anzahl der *Features* verringert, ohne dabei wichtige Eigenschaften der Originaldaten zu verlieren (Abdi und Williams 2010). *Data Preparation* und *Feature Engineering* bilden zusammen das *Preprocessing*.

Mit den vorbereiteten Trainingsdaten kann ein ML-Experte das bestmögliche ML-Verfahren (*Model Selection*) sowie dessen Hyperparameter (*Hyperparameter Opti-*

mization) festlegen. Ein ML-Verfahren bildet die Basis eines ML-Modells; es existiert eine Vielzahl an ML-Verfahren mit jeweils Vor- und Nachteilen. *Artificial Neural Networks* (ANN) haben meist eine hohe Effektivität, jedoch benötigen diese eine sehr große Menge an Daten und sind ressourcenhungrig. Außerdem erfordert die Erstellung einer geeigneten Architektur (*Neural Architecture Search*) viel Erfahrung.

Hyperparameter nennt man die Stellschrauben, mit denen ML-Verfahren angepasst werden können. Abhängig von der Aufgabe kann eine Vielzahl von ML-Verfahren eingesetzt werden. Jedes dieser ML-Verfahren hat wiederum eine Vielzahl von Hyperparametern, deren Werte sich auf die Effektivität des ML-Modells auswirken. In einem ANN ist dies beispielsweise die Anzahl von Neuronen in einem Layer.

Nachdem das ML-Verfahren ausgewählt und die Hyperparameter bestimmt wurden, beginnt das Modelltraining. Mithilfe der *Learning Curve Analysis* können Informationen zur Optimierung der Hyperparameter gewonnen werden, z. B. ob es sich um ein Overfitting-Szenario (das Modell ist zu komplex) oder ein Underfitting-Szenario (das Modell ist zu einfach) handelt (Nusrat und Jang 2018).

Im Fall, dass ein *Ensemble* aus mehreren ML-Modellen gebildet werden soll, muss das neue Modell in dieses integriert werden (*Ensemble Building*).

Am Ende jeder Trainingsiteration wird das ML-Modell mit Testdaten validiert, um dessen Effektivität zu messen. Die Testdaten sind ein Subset der Trainingsdaten, die von dem ML-Modell noch nicht vorhergesehen wurden. Aus der Vorhersage der Testdaten wird ein Gütemaß berechnet, mit dem der Experte entscheidet, ob die Effektivität des ML-Modells den Anforderungen entspricht. Bei einem Klassifikationsproblem kann beispielsweise die *Accuracy* als Gütemaß verwendet werden. Die *Accuracy* berechnet sich aus der Anzahl an richtigen Vorhersagen durch die Anzahl der gesamten Vorhersagen. Auch die Auswahl des passenden Gütemaßes erfordert viel Erfahrung.

Wenn der Experte mit der Effektivität des finalen ML-Modells zufrieden ist, wird die ML Pipeline finalisiert und der ML Workflow ist beendet. Sollte jedoch das ML-Modell nicht den Erwartungen entsprechen, muss eine neue Iteration mit angepassten ML-Verfahren oder Hyperparametern durchgeführt werden.

Effektive ML-Modelle zu erstellen hat etwas von einer „Geheimkunst“, da sie auf umfangreicher Expertise und oft dem „Bauchgefühl“ des ML-Experten basiert. In den Abb. 8.5 und 8.6 haben wir das durch einen Magier symbolisiert. Im nächsten Abschnitt beschreiben wir ein Verfahren, das diese Expertise in Teilen automatisiert und so bildlich den Magier teilweise durch eine Maschine ersetzt.

8.2.2 AutoML

Automated Machine Learning (AutoML; Zöller und Huber 2021) ist ein alternativer Ansatz zum traditionellen, manuellen Vorgehen im ML Workflow. Durch den Einsatz von AutoML-Lösungen wird dieser Prozess in Teilen automatisiert, wie in Abb. 8.7

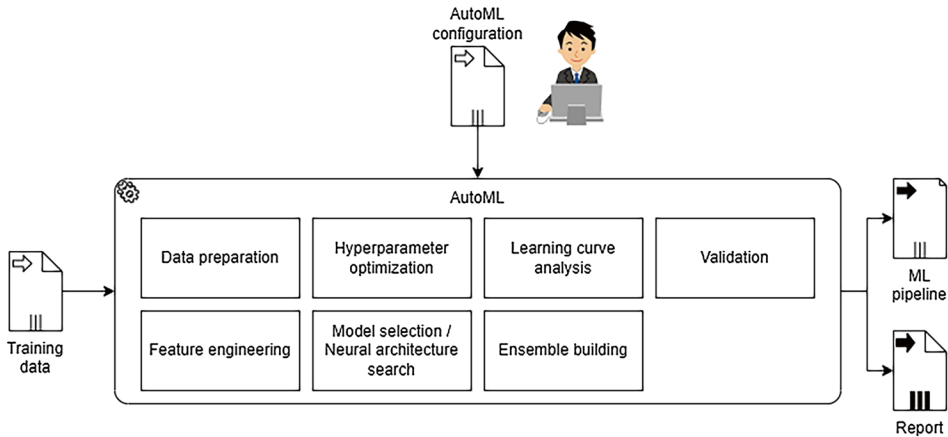


Abb. 8.7 ML Workflow bei Verwendung von AutoML (nach Zender und Humm (2022))

dargestellt. AutoML sucht dabei nach einer Lösung für das *Combined-Algorithm-Selection-and-Hyperparameter-Optimization-(CASH)-Problem* (Thornton et al. 2013), bei dem sowohl das beste ML-Verfahren als auch die beste Einstellung der Hyperparameter gesucht werden.

Zusätzlich zu den Trainingsdaten wird eine AutoML-Konfiguration erwartet, die die gewünschte ML-Aufgabe mit den erforderlichen Parametern definiert. Bei einem Klassifikationsproblem wäre beispielsweise das Target angegeben. Je nach AutoML-Lösung können auch weitere optionale Parameter festgelegt werden. In AutoKeras kann beispielsweise die maximale Anzahl von Versuchen (`max_trials`) begrenzt werden, um die Anzahl der generierten und verglichenen ML-Modelle einzuschränken (Autokeras 2023).

Nach Abschluss des AutoML-Prozesses wird die ML Pipeline sowie ein *Report* zurückgegeben. Der Report kann verschiedene Informationen enthalten, von einer Zusammenfassung der Effektivität der ML Pipeline bis hin zu detaillierten Trainingsübersichten und grafischen Auswertungen des Trainingsprozesses. Der Detailgrad des Reports hängt von der verwendeten AutoML-Lösung ab (Zender und Humm 2022).

AutoML-Lösungen werben damit, dass sie jedem ermöglichen, effektive ML-Modelle zu generieren, unabhängig von der ML-Expertise.³ Es gibt webbasierte AutoML-Lösungen, die ohne Installation oder Programmierkenntnisse verwendet werden können. Die meisten AutoML-Lösungen sind jedoch Softwarebibliotheken, die Programmierkenntnisse erfordern. Ein Beispiel für eine AutoML-Softwarebibliothek ist *AutoSklearn* (Feurer et al. 2020). Abb. 8.8 zeigt einen Code-Ausschnitt für eine minimale Implementierung.

³<https://aws.amazon.com/de/sagemaker/>

```
import autosklearn.classification
cls = autosklearn.classification.AutoSklearnClassifier()
cls.fit(X_train, y_train)
predictions = cls.predict(X_test)
```

Abb. 8.8 Minimale Implementierung für die AutoML-Lösung AutoSklearn (nach Humm und Zender (2021))

Für die Verwendung von AutoSklearn reichen vier Zeilen Python-Code aus. Als Erstes wird die AutoSklearn-Softwarebibliothek eingebunden. Danach wird eine neue Instanz des AutoML Classifier instanziiert und ein neues Training mit der *fit*-Funktion gestartet. Nach dem Training können mittels der *predict*-Funktion neue Vorhersagen auf Live-Daten getroffen werden.

AutoSklearn bietet dem Nutzer eine Vielzahl von Parametern, um den AutoML-Prozess anzupassen, jedoch erfordert dies ML-Expertise. Allen AutoML-Lösungen gemein ist ein gewisser Vendor Lock-in, also eine Bindung an ein technologisches Ökosystem – hier die ML-Bibliothek Skikit-learn. Die Portierung in ein anderes Ökosystem bzw. eine andere ML-Bibliothek ist entweder gar nicht oder nur bedingt möglich.

Wir haben eine umfangreiche Recherche aller bekannten AutoML-Lösungen durchgeführt (Zender und Humm 2022). Abb. 8.9 zeigt eine aktualisierte Übersicht von 31 verschiedenen Lösungen. Dargestellt wir jeweils:

- Art der Lösung (Type): Open Source (OS), kommerzielle Lösungen (C), Softwarebibliotheken (SL), Web Services (WS), lokale Applikationen (LA),
- Zielgruppe (Target Users): Fachexperten (Domain Expert), ML-Experten (ML Expert),
- Aufgaben (ML Task): Klassifikation, Regression etc. für verschiedene Datensatztypen wie Tabellen (Tabular Data), Texte, Bilder etc.,
- Reifegrad (Maturity): Softwarestand Pre-release (PR), Release (R), unbekannt (UK); Projektaktivität aktiv (AD), nicht aktiv (NAD), unbekannt (UK).

Die Mehrheit der AutoML-Lösungen sind Open-Source-(OS-)Softwarebibliotheken (SL), während kommerzielle Lösungen (C) als webbasierte (WS) oder lokale Applikationen (LA) verfügbar sind. Die unterstützten ML-Aufgaben variieren, wobei die meisten AutoML-Lösungen Klassifikation, Regression und Zeitreihenvorhersagen auf tabellarischen Daten unterstützen.

Eine deutliche Mehrheit der Open-Source-Lösungen eignet sich ausschließlich für die Verwendung durch ML-Experten, während kommerzielle AutoML-Lösungen sowohl Fachexperten als auch ML-Experten adressieren.

Bei kommerziellen AutoML-Lösungen sind meist keine öffentlichen Informationen über den internen Entwicklungsstand bekannt (UK). Bei Open-Source-Lösungen hingegen ist eine deutliche Mehrheit im Pre-release-Stadium, mit teils unter aktiver und teils

1. Durchführung von AutoML: Meta AutoML generiert ML Pipelines und Reports basierend auf den eingegebenen Trainingsdaten und der AutoML-Konfiguration.
2. Keine ML-Expertise ist erforderlich: Meta AutoML soll auch von Fachexperten, unabhängig von ihrer ML-Expertise, verwendet werden können. Dies wird durch eine grafische Benutzeroberfläche (GUI) ermöglicht.
3. Technologieunabhängigkeit: Jede ML-Bibliothek soll potenziell unterstützt werden.
4. Breite Abdeckung von ML-Aufgaben: Es sollen möglichst viele ML-Aufgaben unterstützt werden.

Der Meta-AutoML-Prozess erweitert den AutoML-Prozess. Wie in Abb. 8.10 dargestellt, gibt der Nutzer dieselben Eingaben ein: Trainingsdaten und eine AutoML-Konfiguration. Der Meta-Layer führt eventuelle Vorverarbeitungsschritte (Preprocessing) für individuelle AutoML-Lösungen durch, z. B. die Konvertierung von Datentypen, und übergibt die Eingaben an die AutoML-Lösungen. Diese suchen nach der besten ML Pipeline. Die Ergebnisse werden zusammengeführt und es wird Programmcode zur Ausführung der ML-Modelle generiert. Anschließend werden die ML-Modelle analysiert. Dem Nutzer stehen die ML Pipeline (ML-Modell und ausführbarer Programmcode) sowie der Report über das Training, einschließlich der Analyseergebnisse des ML-Modells, zur Verfügung.

Der Report enthält auch Informationen über die verwendete ML-Bibliothek und die ML-Verfahren, auf denen das ML-Modell basiert. Der Nutzer entscheidet, welche ML Pipeline am besten zu seinem Anwendungsfall passt, ohne vorher Kenntnisse über AutoML-Lösungen erwerben zu müssen. Durch die Verwendung einer benutzerfreundlichen GUI ist keine ML-Expertise erforderlich. Meta AutoML adressiert somit alle Probleme von AutoML, außer der Optimierung der AutoML-Konfiguration. Die AutoML-Lösungen können mit einer minimalen Konfiguration arbeiten, die jedoch optimiert werden

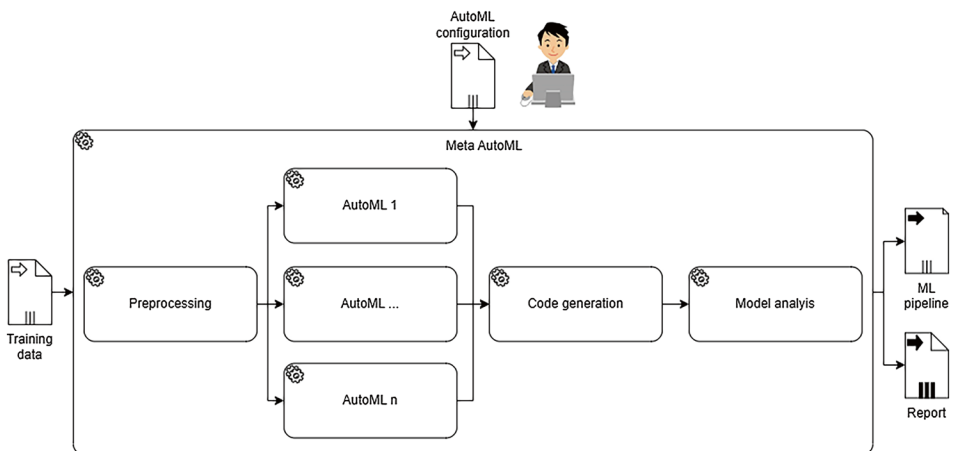


Abb. 8.10 Meta-AutoML-Konzept (nach Humm und Zender (2021))

kann, um effektivere oder effizientere ML-Modelle zu finden. Jede AutoML-Lösung bietet eine Vielzahl von Parametern, die angepasst werden können.

Es ist jedoch zu beachten, dass der Meta-AutoML-Prozess im Vergleich zu einer einzelnen AutoML-Lösung wesentlich mehr Rechenleistung erfordert. OMA-ML versucht, diese mithilfe von wissensbasierter KI zu reduzieren. Mithilfe einer Ontologie werden die AutoML-Konfigurationsparameter automatisch bereitgestellt. Darüber hinaus wird die Trainingseffizienz durch ein regelbasiertes System für AutoML-Trainingsstrategien optimiert. Beide Ansätze werden im nächsten Abschnitt vorgestellt.

8.3 Wissensbasierte Ansätze für Meta AutoML

8.3.1 ML-Ontologie

OMA-ML verwendet eine ML-Ontologie, die das Wissen über Machine Learning, inkl. AutoML, abbildet. Eine Ontologie ist eine formale Darstellung von Wissen in einem bestimmten Anwendungsfeld. Mittels einer Ontologie werden Konzepte, Beziehungen und Eigenschaften innerhalb dieses Fachgebiets strukturiert und maschinenlesbar gemacht. Die ML-Ontologie wird in RDF (Cyaniak et al. 2023) modelliert und nutzt SKOS (Miles und Bechhofer 2023). RDF ermöglicht die Beschreibung von Ressourcen und ihren Beziehungen zueinander durch *Tripel*, bestehend aus Subjekt, Prädikat und Objekt. Zum Beispiel beschreibt *AutoKeras a:AutoML_Solution*, dass AutoKeras eine AutoML-Lösung ist. SKOS ist eine einfache Ontologiesprache, die auf RDF basiert.

Die ML-Ontologie umfasst derzeit mehr als 4000 Tripel, darunter aktuell ca. 120 ML-Verfahren, 40 ML-Aufgaben, 120 Metriken, 25 ML-Bibliotheken, 30 AutoML-Lösungen sowie deren Konfigurationsparameter und Beziehungen zueinander. Die ML Ontology ist "open source" verfügbar unter GitHub.⁴ Das Klassen- und Beziehungsschema der ML-Ontologie wird in Abb. 8.11 gezeigt.

Im oberen Teil der Abbildung werden Klassen für allgemeine ML-Konzepte dargestellt. Die Klasse *ML Area* umfasst die Teilgebiete, in die Machine Learning unterteilt ist (Supervised Learning, Unsupervised Learning, Reinforcement Learning). Die Klasse *ML Task* bildet Aufgaben ab, die durch ML gelöst werden (z. B. Classification). Jede ML-Aufgabe ist mit einem Teilgebiet assoziiert, z. B. gehören Klassifikationsaufgaben zum Teilgebiet des Supervised Learning. Die Klasse *ML Approach* repräsentiert ML-Verfahren (z. B. Random Forest) und ist mit den ML-Aufgaben verknüpft, für die sie eingesetzt werden können. Zum Beispiel kann Random Forest für Klassifikationsaufgaben verwendet werden. Die Klasse *Preprocessing Approach* stellt Methoden für die Datenvorbereitung dar (z. B. One-Hot-Encoding) und zeigt die Beziehungen zu den ML-

⁴https://github.com/hochschule-darmstadt/MetaAutoML/blob/main/controller/managers/ontology/ML_Ontology.ttl

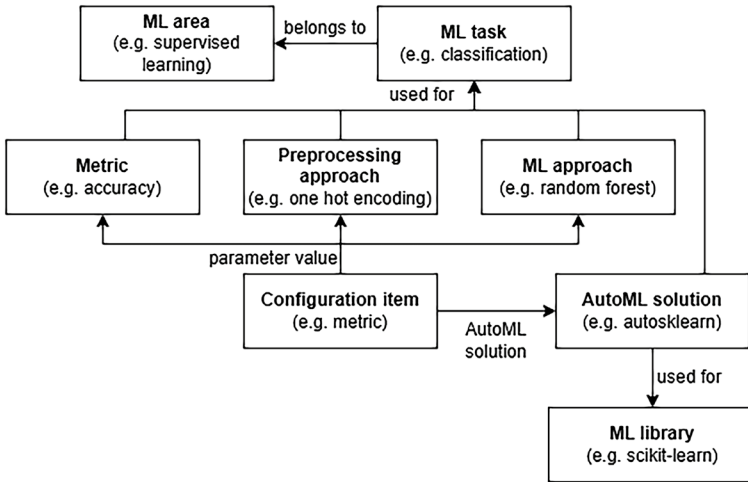


Abb. 8.11 Schema der ML-Ontologie (nach Zender und Humm (2022))

Aufgaben an, bei denen sie angewendet werden können. Die Klasse *Metric* umfasst eine Sammlung vorhandener ML-Gütemaße (z. B. *Accuracy*) und zeigt die Beziehungen zu den ML-Aufgaben an, für die diese verwendet werden können. Zum Beispiel kann die *Accuracy* für Klassifikationsaufgaben verwendet werden.

Im unteren Teil der Abbildung werden Klassen dargestellt, die Implementierungen von ML-Konzepten darstellen. Die Klasse *ML Library* bildet bestehende ML-Bibliotheken ab (z. B. *Scikit-learn*). Die Klasse *AutoML Solution* repräsentiert verfügbare AutoML-Lösungen (z. B. *AutoSklearn*). Jede AutoML-Lösung basiert auf einer oder mehreren ML-Bibliotheken und unterstützt eine oder mehrere ML-Aufgaben. Zum Beispiel basiert *AutoSklearn* auf der ML-Bibliothek *Scikit-learn* und kann für Klassifikationsaufgaben verwendet werden. Die Klasse *Configuration Item* schließlich repräsentiert Konfigurationsparameter, die individuelle AutoML-Lösungen anbieten (z. B. *Metric*). Diese Konfigurationsparameter können Beziehungen zu Metrics, Preprocessing Approaches, ML Approaches oder AutoML Solutions haben. Zum Beispiel wird der Parameter "metric" von *AutoSklearn* angeboten und kann den Wert "accuracy" haben.

Innerhalb der ML-Ontologie wird jede Instanz durch einen eindeutigen IRI (Internationalized Resource Identifier) identifiziert. Zusätzlich verfügen Instanzen über Annotationen, die Eigenschaften und Beziehungen einer Instanz abbilden. Abb. 8.12 zeigt beispielhaft die Instanz *Artificial Neural Network (ANN)* der Klasse *ML Approach* mit ihren Annotationen.

Im Beispiel sind folgende Annotationen angegeben:

- *skos:prefLabel*: die Bezeichnung der Instanz, hier *Artificial Neural Network*;
- *skos:altLabel*: alternative Bezeichnungen, hier die Abkürzung *ANN*;

IRI

http://h-da.de/ml-ontology/artificial_neural_network

Annotations

skos:prefLabel	artificial neural network
rdfs:comment	computational model used in machine learning, based on connected, hierarchical functions
skos:altLabel	neural network
skos:altLabel	ANN
rdfs:seeAlso	https://www.wikidata.org/wiki/Q192776
performance	long_training_time
performance	accurate
possible_if	numerical_features
used_for	time_series_classification
used_for	anomaly_detection
used_for	regression
used_for	classification
Enter property	Enter value

Types

ML_approach

Abb. 8.12 Annotationen der Instanz Artificial Neural Network

- *rdfs:comment*: eine Kurzbeschreibung;
- *rdfs:seeAlso*: ein Link zur entsprechenden Wikidata⁵-Seite mit Detailinformationen sowie Links zu Wikipedia-Artikeln über neuronale Netze;
- *used_for*: die ML Tasks, für die ANN eingesetzt werden können, z. B. Klassifikation (*Classification*), Regression (*Regression*), Zeitreihenanalysen (*time_series_classification*) oder Anomalieerkennung (*anomaly_detection*);
- *performance*: Leistungsmerkmale des ML-Verfahrens: ANN haben in der Regel eine hohe Vorhersagegüte (*Accurate*), aber benötigen eine vergleichsweise lange Trainingszeit (*long_training_time*);

⁵<https://www.wikidata.org>

- *possible_if*: Bedingungen, die erfüllt sein müssen, bevor ein ML-Verfahren verwendet werden kann: Bei ANN müssen die Features in numerischer Form vorliegen (*numerical_features*).

Die ML-Ontologie bildet die zentrale Wissensbasis für OMA-ML. Wie in der Einleitung beschrieben, bietet der Training-Wizard dutzende Konfigurationsmöglichkeiten für den ML-Experten an (Abb. 8.3). Dabei werden nur inhaltlich konsistente Konfigurationen angezeigt, abhängig von der Selektion des Nutzers. Was inhaltlich konsistent ist, ist in der ML-Ontologie festgelegt. Selektiert beispielsweise der Nutzer Scikit-learn als erwünschte ML Library (per Default sind alle unterstützten ML-Bibliotheken selektiert), dann werden automatisch die AutoML Solutions Autosklearn, Autogluon, EvalML, FLAM, GAMA, MLJAR, Pycaret und TPOT selektiert, da diese Code für Scikit-learn generieren. Deselektiert der Nutzer Scikit-learn, dann werden alle diese AutoML Solutions automatisch deselektiert. Umgekehrt kann der Benutzer auch einzelne AutoML Solutions manuell selektieren und die unterstützten ML Libraries werden automatisch angepasst. Jede AutoML Solution bietet dutzende Konfigurationsmöglichkeiten, z. B. Autosklearn u. a. *per_run_time_limit*, *ensemble_size*, *ensemble_class*, *memory_limit*, *include*, *exclude*, *metric* und viele mehr. Diese Konfigurationsparameter und ihre möglichen Werte sind in der ML Ontology hinterlegt, gemeinsam mit Informationen wie Hilfetexte und weiterführende Links, z. B. auf Wikidata. Aus den Informationen der ML Ontology werden automatisch die GUI-Elemente des Training-Wizards erstellt, inklusive der interaktiven Hilfe.

Zusätzlich zur ML-Ontologie verwendet OMA-ML ein Regelsystem für Trainingsstrategien, um die Trainingseffizienz zu steigern.

8.3.2 Regelsystem für Trainingsstrategien

Der Meta-AutoML-Trainingsprozess ist per se ineffizient, da für jedes Training alle verfügbaren AutoML-Lösungen parallel ausgeführt werden (Zender et al. 2023). Wenn beispielsweise neun AutoML-Lösungen unterstützt werden, werden neun verschiedene ML-Modelle generiert. Dies ermöglicht dem Benutzer die Auswahl des am besten geeigneten Modells. Allerdings wird in den meisten Fällen das effektivste Modell ausgewählt und die anderen Modelle bleiben ungenutzt.

Um die Trainingseffizienz zu steigern, nutzt OMA-ML ein Regelsystem für Trainingsstrategien (Zender et al. 2023). Eine einfache Strategie zur Effizienzsteigerung ist die Top-3-Strategie. Dabei wird das Training in zwei Phasen aufgeteilt. Zunächst erfolgt ein kürzeres Vortraining mit allen aktivierten AutoML-Lösungen. Hierbei wird den AutoML-Lösungen eine kleine Teilmenge der Trainingsdaten mit reduzierter Trainingszeit übergeben. Zum Beispiel erhält jede AutoML-Lösung nur 20 % der Zeit und Trainingsdaten. Nachdem alle AutoML-Lösungen ihre Modelle gefunden haben, werden diese miteinander verglichen. Die drei AutoML-Lösungen mit den besten ML-Modellen

starten einen neuen Trainingslauf mit der verbleibenden Zeit und allen Trainingsdaten. Am Ende des Trainings werden dem Benutzer die drei ML-Modelle aus dem zweiten Trainingslauf präsentiert.

Für die Trainingsverwaltung und Entscheidung über anzuwendende Trainingsstrategien implementiert OMA-ML ein Regelsystem basierend auf der Blackboard-Architektur (Penny 1980). Diese Architektur nutzt mehrere spezialisierte Module, um ein komplexes Problem zu lösen. Dabei gibt es drei Komponenten: das *Blackboard*, das den aktuellen Stand des Problems speichert, *Knowledge Sources*, die das Fachwissen bereitstellen, und den *Controller*, der den Ablauf steuert. Abb. 8.13 zeigt ein BPMN-Diagramm (BPMN 2013) für den Ablauf des Regelsystems.

Der Benutzer startet ein neues Training, indem er im Training-Wizard die AutoML-Konfiguration festlegt. Dabei setzt dieser Parameter wie maximale Trainingszeit, ML Task, Target und aktivierte AutoML Solutions (*Training Configuration*). Wenn Trainingsdaten auf die Plattform hochgeladen werden, analysiert OMA-ML diese automatisch und speichert die Ergebnisse (*Analysis Result*). Dabei werden Eigenschaften und Merkmale der Trainingsdaten extrahiert, wie die Anzahl der Zeilen, Spalten und doppelten Zeilen bei tabellarischen Daten.

Training Configuration und Analysis Result werden auf das Blackboard geschrieben und stellen den initialen Trainingsstatus dar. Das Blackboard liefert allen Komponenten Informationen über den aktuellen Trainingsstatus. Der *Reasoner*, der zusammen mit dem

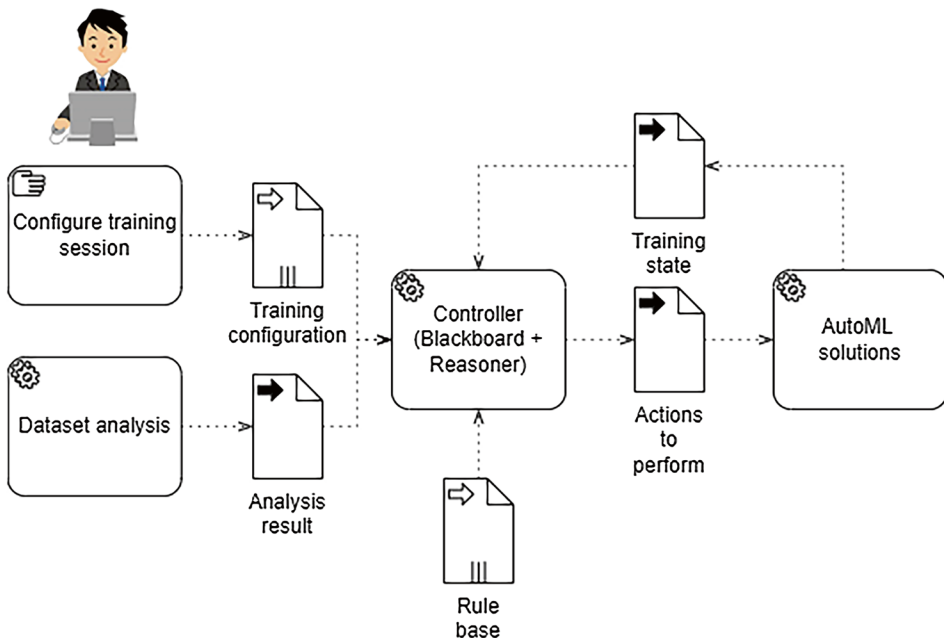


Abb. 8.13 Ablauf des Regelsystems (nach Zender et al. (2023))

Blackboard die Controller-Komponente bildet, entscheidet, welche Trainingsstrategien mit dem aktuellen Trainingsstatus kompatibel sind. Hierfür verwendet dieser die *Rule Base*, eine Sammlung von Trainingsstrategien, die von OMA-ML unterstützt werden. Jede Trainingsstrategie besteht aus einer Bedingung und einer Sammlung von Anweisungen. Zum Beispiel erfordert die Top-3-Strategie, dass mehr als drei AutoML-Lösungen aktiviert sind. Sobald der initiale Trainingsstatus verfügbar ist, überprüft der Reasoner, ob die Bedingungen einer Regel erfüllt sind. Falls ja, wird die Regel ausgeführt. Bei der Top-3-Strategie erhalten alle aktivierten AutoML-Lösungen die Anweisung (*Actions to Perform*), ein Vortraining zu starten, mit einer reduzierten Zeit und Datenmenge. Während des Trainings schreiben die AutoML-Lösungen ihren aktuellen Status (*Training State*) auf das Blackboard, z. B. die aktuelle Effektivität des ML-Modells. Sobald alle aktiven AutoML-Lösungen das Vortraining beendet haben, wird der zweite Teil der Top-3-Strategie ausgeführt. Dabei wird die Effektivität der Modelle ausgewertet und die drei AutoML-Lösungen mit den besten Modellen werden angewiesen, ein zweites Training mit der verbleibenden Zeit und allen Trainingsdaten zu starten.

Wir konnten zeigen, dass durch die Aktivierung der Top-3-Strategie eine Energieersparnis von bis zu 70 % erreicht werden kann bei nahezu gleichbleibender Effektivität der ML-Modelle. Mehr dazu im Abschn. 8.5 („Evaluation:OMA-ML in Aktion“).

Im nächsten Abschnitt stellen wir die Implementierung und die verschiedenen Technologien vor, auf denen OMA-ML basiert.

8.4 Implementierung der OMA-ML-Plattform

OMA-ML ist Open Source unter GitHub.⁶ Die Implementierung folgt einer 3-Schichten-Architektur. Abb. 8.14 zeigt die Software-Architektur als UML-Komponentendiagramm (UML 2012) mit den verwendeten Technologien.

Das OMA-ML *User Interface*, mit dem der Nutzer interagiert, ist in C# entwickelt und basiert auf dem Blazor Framework.⁷ Das User Interface bildet den *Presentation Layer* und ist über ein GRPC Interface⁸ mit dem *Logic Layer* verbunden. Der Logic Layer ist in Python entwickelt und basiert auf der Blackboard-Architektur. Die *Controller*-Komponente verwaltet das Meta-AutoML-Training und nutzt die *Rule-engine*-Bibliothek.⁹ Mit dieser Bibliothek wird der Reasoner implementiert, der basierend auf der Rule Base Schlussfolgerungen zieht. Die AutoML-Lösungen werden nach dem Adapter-Pattern (Gamma 2011) implementiert und können vom Controller einfach aufgerufen werden.

⁶<https://github.com/hochschule-darmstadt/MetaAutoML>

⁷<https://dotnet.microsoft.com/en-us/apps/aspnet/web-apps/blazor>

⁸<https://grpc.io>

⁹<https://pypi.org/project/rule-engine>

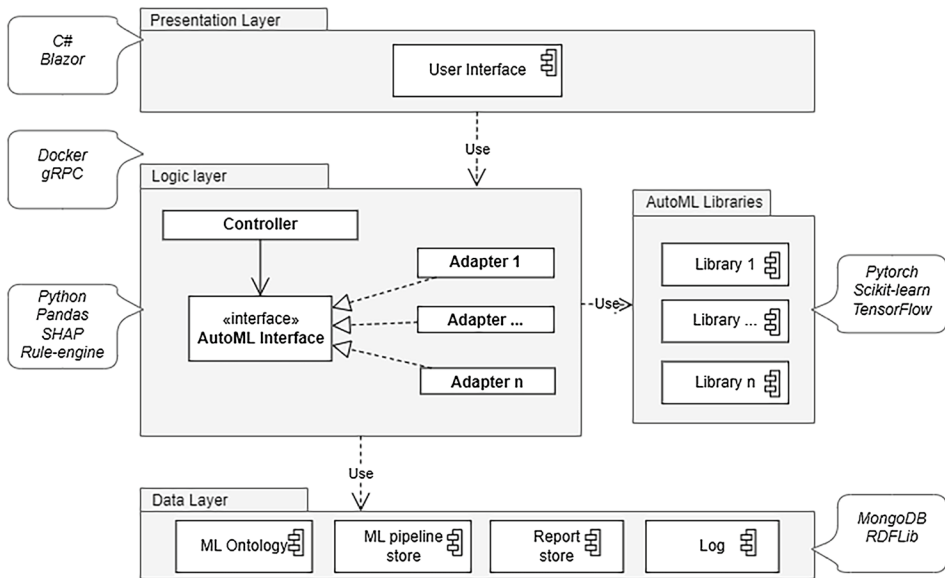


Abb. 8.14 OMA-ML-Software-Architektur (aktualisiert nach Humm und Zender (2021))

Der *Data Layer* stellt verschiedene Datenquellen für den Logic Layer bereit. Die ML-Ontologie ist im Data Layer platziert und wird mit der *RDFlib*-Bibliothek¹⁰ in den Controller geladen. OMA-ML verwendet die Dokumentendatenbank *MongoDB*¹¹ zur Persistierung aller generierten Daten des Meta-AutoML-Prozesses. Beispielsweise werden AutoML-Konfigurationen vergangener Trainings darin gespeichert. Die ML Pipelines, die während eines Trainings gefunden wurden, werden im *ML Pipelines Store* abgespeichert. Im *Report Store* werden Anhänge gespeichert, die für Reports generiert wurden. Nach einem AutoML-Training wird das ML-Modell mithilfe der *SHAP*-Bibliothek¹² analysiert, wobei verschiedene Diagramme automatisch generiert werden. Diese Diagramme werden im Report Store gespeichert. Schließlich werden während eines Trainings generierte Logs im *Log-Speicher* abgelegt.

Die komplette OMA-ML-Plattform ist als Docker-Container verfügbar und kann mit *Docker*¹³ auf jedem System ausgeführt werden.

¹⁰<https://rdflib.dev>

¹¹<https://www.mongodb.com>

¹²<https://pypi.org/project/shap>

¹³<https://www.docker.com>

8.5 Evaluation: OMA-ML in Aktion

In diesem Abschnitt zeigen wir anhand einer Evaluation, dass der Einsatz wissensbasierter KI die Effizienz von Meta AutoML deutlich steigern kann, ohne merklichen Verlust an Effektivität (Zender et al. 2023). Wir haben dafür beispielhaft die Top-3-Strategie mit 5 Benchmark-Datensätzen evaluiert. Zur Bewertung werden zwei Gütemaße verwendet:

- *Accuracy*: die Accuracy des besten ML-Modells. Die Accuracy ist ein einfaches ML-Gütemaß und berechnet sich aus der Anzahl an richtigen Vorhersagen durch die Anzahl der gesamten Vorhersagen;
- $CO_2\text{-eq}[g]$: die Summe der während des Trainings erzeugten CO_2 -Äquivalents in Gramm als Maß für den Energieverbrauch und damit den ökologischen Fußabdruck.

OMA-ML verwendet die Python-Bibliothek *Codecarbon*,¹⁴ um automatisiert den CO_2 -Äquivalentausstoß während des Trainings eines ML-Modells zu messen und dem Benutzer anzuzeigen. Dabei bewertet Codecarbon den Energieverbrauch der zugrunde liegenden Hardware in Kilowattstunden und multipliziert diesen Wert mit der Carbon-Intensität des Stromnetzes. Die Carbon-Intensität wird aus dem Durchschnitt der Emissionen verschiedener Energiequellen berechnet, die zur Erzeugung der verwendeten Elektrizität verwendet werden (z. B. Erdgas, Kohle, Wind).¹⁵

Für die Evaluation wurden fünf Datensätze („adult“,¹⁶ „amazon“,¹⁷ „sylvine“,¹⁸ „credit-g“,¹⁹ „kcl“²⁰) aus dem Open-Source AutoML Benchmark von Gijsbers et al. (2019) verwendet. Es handelt sich jeweils um ein Klassifikationsproblem; die Größe der Datensätze reicht von 1000 bis 48.000 Zeilen.

Für jeden Datensatz wurden zwei Trainingsläufe mit OMA-ML durchgeführt. Im ersten Lauf wurden keine Trainingsstrategien aktiviert und alle verfügbaren AutoML-Lösungen verwendet, während im zweiten Lauf die Top-3-Trainingsstrategie aktiviert wurde. Als Hardware wurde ein AMD Ryzen 7 5800 H @ 3,20 GHz CPU mit 64 GB RAM verwendet. Nach jedem Training wurden die Accuracy und die Summe der CO_2 -Äquivalente von allen ML-Modellen des Trainings vom besten ML-Modell dokumentiert. Eine Zusammenfassung der Trainingsergebnisse ist in Abb. 8.15 dargestellt.

¹⁴<https://github.com/mlco2/codecarbon>

¹⁵<https://mlco2.github.io/codecarbon/methodology.html>

¹⁶<https://openml.org/search?type=data&status=active&id=179>

¹⁷<https://openml.org/search?type=data&status=active&id=4135>

¹⁸<https://openml.org/search?type=data&status=active&id=41146>

¹⁹<https://openml.org/search?type=data&status=active&id=31>

²⁰<https://openml.org/search?type=data&status=active&id=1067>

Datensatz	Ohne Trainingsstrategie			Top-3 Trainingsstrategie			Differenz Accuracy [Prozentpunkte]	CO2-eq Ersparnis
	AutoML	Accuracy	CO2-eq [g]	AutoML	Accuracy	CO2-eq [g]		
sylvine	Autosklearn	95%	50,0	Autosklearn	95%	15,0	0	70%
credit-g	FLAML	78%	36,2	MLJAR	77%	11,7	-1	68%
kc1	Autogluon	87%	37,0	Autogluon	87%	13,1	0	65%
amazon	Autogluon	95%	45,8	Autogluon	95%	16,6	0	64%
adult	FLAML	88%	53,5	FLAML	87%	22,1	-1	59%

Abb. 8.15 Zusammenfassung der Trainingsergebnisse (nach Zender et al. (2023))

Für alle Datensätze konnte durch die Verwendung der Trainingsstrategie eine drastische Einsparung der CO₂-Äquivalente erreicht werden. Die Einsparungen reichten von mindestens 59 % beim Datensatz „adult“ bis zu 70 % beim Datensatz „sylvine“ im Vergleich zum nichtoptimierten Training. Zusätzlich wurde durch die Anwendung der Trainingsstrategie die Effektivität des besten ML-Modells nur um maximal 1 Prozentpunkt im Vergleich zum nichtoptimierten Training verringert. In 4 von 5 Datensätzen lieferte sowohl im optimierten als auch im nichtoptimierten Training dieselbe AutoML-Lösung das beste ML-Modell. Nur beim Datensatz „credit-g“ wurde im nichtoptimierten Training FLAML und im optimierten Training MLJAR als beste AutoML-Lösung identifiziert.

Durch die Verwendung von Trainingsstrategien wie der Top-3-Strategie kann OMA-ML effizienter werden. Jedoch wird weiterhin Energie für das Training von ML-Modellen verwendet, die keine Anwendung finden. Können wir nicht weitere KI-Verfahren einsetzen, um hier noch besser zu werden? Im folgenden Abschnitt stellen wir eine Idee als Ausblick vor.

8.6 Ausblick: Meta Learning für Meta AutoML

Für ein Meta-AutoML-Training stehen verschiedene AutoML-Lösungen zur Verfügung, von denen jede ihre individuellen Stärken und Schwächen hat. Bei jedem Training versuchen alle AutoML-Lösungen, das beste ML-Modell zu finden. Für den Benutzer ist jedoch nur das effektivste ML-Modell von Relevanz. Dies führt dazu, dass viele AutoML-Lösungen unnötig laufen und dabei viel Energie verschwenden. Ein Benutzer kann manuell die Anzahl der verwendeten AutoML-Lösungen reduzieren, indem er die Auswahl an ML-Bibliotheken oder AutoML-Lösungen einschränkt. Allerdings kann es ohne Kenntnis über die Effektivität der einzelnen AutoML-Lösungen dazu kommen, dass der Benutzer unbeabsichtigt die besten AutoML-Lösungen ausschließt.

OMA-ML speichert alle generierten AutoML-Konfigurationen und Trainingsergebnisse ab. Dadurch verfügt OMA-ML über eine breite und kontinuierlich wachsende Informationsbasis aus verschiedenen Datensätzen, AutoML-Konfigurationen und der Effektivität der ML-Modelle.

Name	Größe (Bytes)	ML Aufgabe	AutoML	Accuracy	CO2-eq (g)	Trainingszeit (min)
diabetes.csv	23875	:tabular_classification	:autokeras	0.97	10	37
diabetes.csv	23875	:tabular_classification	:autogluon	0.90	12	20
diabetes.csv	23875	:tabular_classification	:autosklearn	0.94	15	26

Abb. 8.16 Beispiel für ein Datensatz aus bisherigen OMA-ML-Trainings

Die Idee ist nun, alle Informationen über alle bisherigen OMA-ML-Trainings in einen großen Datensatz zu überführen und auf diesen ML-Verfahren anzuwenden. Abb. 8.16 zeigt einen Beispieldatensatz, bestehend aus vergangenen Trainingsinformationen.

Ein so trainiertes ML-Modell soll dann in der Lage sein, für einen neuen Datensatz Folgendes vorherzusagen:

1. Wie effektiv ist voraussichtlich jede einzelne AutoML-Lösung für diesen Datensatz, insbesondere welche AutoML-Lösung wird voraussichtlich das beste Modell generieren?
2. Wie effizient ist voraussichtlich jede einzelne AutoML-Lösung für diesen Datensatz, insbesondere welche AutoML-Lösung wird voraussichtlich am energiesparendsten sein, bezogen auf das Training und für Vorhersagen?
3. Wie lange sollte man sinnvollerweise die Trainingszeit einstellen, d. h., ab welcher Zeit ist keine Verbesserung der Effektivität mehr zu erwarten?

Es geht also um ein Meta Learning, basierend auf AutoML-Informationen. OMA-ML bietet die Datengrundlage für viele solcher KI-Verfahren und wir erwarten weitere Verbesserungen von Effektivität und Effizienz von ML gemäß dem Claim von OMA-ML: „Effective machine learning made easy!“

8.7 Empfehlungen

Bei der Entwicklung von OMA-ML haben wir viel gelernt. Die folgenden Empfehlungen möchten wir gerne an unsere Leser weitergeben:

1. Ontologien und insbesondere Knowledge Graphs eignen sich gut als zentrale Informationsbasis für wissensintensive Verfahren und Prozesse, wie beispielsweise ML. Sie können z. B. verwendet werden für komplexe Konfigurationen und interaktive Hilfe.
2. Bevor man eine Ontologie neu entwickelt, empfiehlt es sich, existierende Ontologien zu recherchieren. Falls für ein Anwendungsgebiet keine passende Ontologie existiert (wie es bei OMA-ML der Fall war), sollte die Neuentwicklung zielorientiert und so

einfach wie möglich erfolgen. Im Beispiel der ML Ontology reicht eine Handvoll von Klassen und Properties basierend auf SKOS aus, um eine umfassende und mächtige Ontologie zu erstellen.

3. Klassische regelbasierte Verfahren ergänzen Ontologien, wenn es um die Formalisierung komplexer Handlungsmuster geht. Im Beispiel von OMA-ML konnten ML-Trainingsstrategien als Regeln implementiert werden. Dies erlaubt die Separierung der Logik im Source Code. Es existieren produktionsstaugliche Python-Bibliotheken für Regelsysteme, wie z. B. Rule-engine.
4. Das Blackboard-Architekturmuster hat nicht ausgedient und kann hilfreich sein, wenn es um die Koordination von komplexen Handlungen mit viel parallelem Input geht. Im Beispiel von OMA-ML konnten Strategieregeln basierend auf Nutzerkonfigurationen, Datensatzanalysen sowie aktuellem ML-Trainingsfortschritt automatisch aktiviert werden. Das Resultat der Ausführung von Regeln wird wieder auf das Blackboard geschrieben, was neue Regelaktivierungen auslösen kann.
5. Machine Learning und insbesondere AutoML und Meta AutoML können effektiv, aber auch höchst ressourcenhungrig sein. Die Energieeffizienz sollte im gesamten ML-Entwicklungsprozess im Blick gehalten werden. OMA-ML kann dazu einen Beitrag leisten, indem der Ressourcenverbrauch ausgewiesen wird (Transparenz) sowie Strategien zur Reduktion desselben angeboten werden.

8.8 Fazit

Die Plattform OMA-ML ist eine innovative hybride KI-Lösung für das automatisierte Training von ML-Modellen. Jedoch verbraucht der Meta-AutoML-Trainingsprozess potenziell eine große Menge an Energie. Um die Effizienz des Trainings zu steigern, wird in OMA-ML der Trainingsprozess durch die Verwendung von wissensbasierter KI optimiert.

Eine ML-Ontologie stellt Wissen über Machine Learning und AutoML bereit. Die ML-Ontologie wird verwendet, um in der Benutzerschnittstelle die AutoML-Konfigurationsparameterauswahl zu optimieren. Dabei werden dem Benutzer nur schlüssige Auswahlmöglichkeiten dargestellt. Des Weiteren wird durch den Einsatz eines Regelsystems für Trainingsstrategien der Meta-AutoML-Trainingsvorgang optimiert. Eine einfache Trainingsstrategie ist die Top-3-Strategie. Mit dieser Strategie kann OMA-ML im Vergleich mit einem nichtoptimierten Training bis zu 70 % Energie bzw. CO₂-Emissionen einsparen bei nahezu gleichbleibender Effektivität.

OMA-ML bietet auch umfassende Informationen über Datensätze, Trainingskonfigurationen und ML-Modelle. Diese Informationen werden gespeichert und können für Analysen, Vorhersage und Optimierung des Trainingsprozesses genutzt werden.

Insgesamt ermöglicht die Kombination von wissensbasierten KI-Techniken und Meta AutoML eine effizientere Generierung von ML-Modellen. Durch die gezielte Auswahl

von AutoML-Lösungen, die Berücksichtigung von Umweltaspekten und die kontinuierliche Verbesserung der Plattform kann OMA-ML einen wertvollen Beitrag zur Entwicklung effektiver und ressourcenschonender ML-Lösungen leisten.

Literatur

- [Autokeras 2023] StructuredDataClassifier – AutoKeras 2023. zuletzt besucht am 13.06.2023: https://autokeras.com/structured_data_classifier/
- [Abdi & Williams 2010] H. Abdi and L. J. Williams, “Principal component analysis,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010. <https://doi.org/10.1002/wics.101>
- [BPMN 2013] Information technology—Object Management Group Business Process Model and Notation (2013), www.iso.org/standard/62652.html
- [Cyganiak et al. 2023] R. Cyganiak, D. Wood, M. Lanthaler: “Rdf 1.1 concepts and abstract syntax” <https://www.w3.org/TR/rdf11-concepts/>
- [Feuer et al. 2020] M. Feuer, K. Eggensperger, S. Falkner, M. Lindauer, & F. Hutter Auto-Sklearn 2.0: The Next Generation 2020. <https://arxiv.org/pdf/2007.04074>
- [Gamma 2011] E. Gamma, (2011). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley professional computing series (39. printing ed.). Boston: Addison-Wesley
- [Garcia et al. 2015] S. García, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*. Cham: Springer International Publishing, 2015, vol. 72. ISBN 978-3-319-10246-7
- [Gijbsbers et al. 2019] Gijbsbers, P., LeDell, E., Thomas, J., Poirier, S., Bischl, B. & Vanschoren, J. An Open Source AutoML Benchmark. <https://arxiv.org/pdf/1907.00909>.
- [Humm & Zender 2021] B. G. Humm and A. Zender, “An ontology-based concept for meta automl,” in *Artificial Intelligence Applications and Innovations*, ser. Springer eBook Collection, I. Maglogiannis, J. Macintyre, and L. Iliadis, Eds. Cham: Springer International Publishing and Imprint Springer, 2021, vol. 627, pp. 117–128.
- [Humm et al. 2021] B. G. Humm, H. Bense, M. Fuchs, B. Gernhardt, M. Hemmje, T. Hoppe, L. Kaupp, S. Lothary, K.-U. Schäfer, B. Thull, T. Vogel & T. Wenning, (2021). “Machine intelligence today: applications”, methodology, and technology. *Informatik Spektrum*, 1–11. <https://doi.org/10.1007/s00287-021-01343-1>. <https://link.springer.com/article/10.1007%2Fs00287-021-01343-1>.
- [Jain & Zongker 1997] A. Jain and D. Zongker, “Feature selection: evaluation, application, and small sample performance,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 19, no. 2, pp. 153–158, 1997. <https://doi.org/10.1109/34.574797>
- [Khalid et al. 2014] S. Khalid, T. Khalil, and S. Nasreen, “A survey of feature selection and feature extraction techniques in machine learning,” in *2014 Science and Information Conference*. IEEE, 2014. <https://doi.org/10.1109/SAI.2014.6918213>. ISBN 978-0-9893193-1-7 pp. 372–378.
- [Miao 2021] Y. Miao, “Using Machine Learning Algorithms to Predict Diabetes Mellitus Based on PIMA Indians Diabetes Dataset” in *2021 the 5th International Conference 2021* pp. 47–53 <https://doi.org/10.1145/3463914.3463922>
- [Miles & Bechhofer 2023] A. Miles, S. Bechhofer: Skos simple knowledge organization system namespace document, <https://www.w3.org/2009/08/skos-reference/skos.html>
- [Nusrat & Jang 2018] Nusrat, I., Jang, S. A Comparison of Regularization Techniques in Deep Neural Networks, *Symmetry* 2018, 10(11), 648; <https://doi.org/10.3390/sym10110648>.

- [Penny 1980] H. Penny Nii, “An introduction to knowledge engineering, blackboard model, and age,” 03.1980. [Online]. Available: <https://purl.stanford.edu/cq570jp5428>
- [Potdar & Pai 2017] T. S. K. Potdar & C. Pai, “A comparative study of categorical variable encoding techniques for neural network classifiers,” In *International Journal of Computer Applications*, vol. 175, no. 4, pp. 7–9, 2017. <https://doi.org/10.5120/ijca2017915495>
- [Thornton et al. 2013] C. Thornton, F. Hutter, H. H. Hoos & K. Leyton-Brown, (2013). AutoWEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms. In I.S. Dhillon (Ed.), *KDD’13: the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* : August 11–14, 2013, Chicago, Illinois, USA (pp. 847–855). ACM. <https://doi.org/10.1145/2487575.2487629>.
- [UML 2012] ISO/IEC 19505-2:2012(en), Information technology—Object Management Group Unified Modeling Language (OMG UML)—Part 2: Superstructure. <https://www.iso.org/obp/ui/#iso:std:iso-iec:19505:-2:ed-1:v1:en>
- [Zender et al. 2023] A. Zender, B. G. Humm, T. Pachmann, “Improving the Efficiency of Meta AutoML via Rule-based Training Strategies” in *Proceedings of the 18th Conference on Computer Science and Information Systems 2023 (FedCSIS 2023)* ISBN 978-83-967447-8-4, *Annals of Computer Science and Information Systems (ACSIS)* vol. 35, pp. 235–246. IEEE 2023. ISBN 978-83-967447-8-4
- [Zheng & Casari 2018] A. Zheng and A. Casari, Feature engineering for machine learning: “Principles and techniques for data scientists”. In O’Reilly, 2018. ISBN 978-149195324
- [Zöllner & Huber 2021] M.-A. Zöllner and M. F. Huber, “Benchmark and survey of automated machine learning frameworks,” *Journal of Artificial Intelligence Research* 70, 2021: <https://arxiv.org/pdf/1904.12054>
- [Zender & Humm 2022] A. Zender and B. G. Humm, “Ontology-based meta automl,” *Integrated Computer Aided Engineering*, vol. 29, no. 4, pp. 351–366, 2022

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.



Teil III

Inhärent hybride KI-Verfahren



Ermittlung elektiver Risikopatienten mittels Bayes'scher Netze

9

Thomas Hoppe , Isabel Mattig , Gina Barzen, Shirkouh Matini, Kerstin Köhler, Henryk Dreger  und Friedrich Köhler 

Kernaussagen

1. Traditionelle Lernverfahren und künstliche neuronale Netze können auf der Basis von Big Data in großen Hypothesenräumen sehr gute Ergebnisse produzieren; bei Small Data mit vielen Parametern neigen sie jedoch zu Übergeneralisierung.
2. Die Verarbeitung von Small Data in großen Hypothesenräumen erfordert zusätzliches Hintergrundwissen, um den Hypothesenraum einzuschränken.
3. Bayes'sche Netze kombinieren Wahrscheinlichkeitsverteilungen mit domänen-spezifischen Parameterabhängigkeiten, um den Suchraum der kombinierten Wahrscheinlichkeitsverteilung drastisch zu reduzieren.

T. Hoppe (✉) · S. Matini
Fraunhofer-Institut Fokus, Berlin, Deutschland
E-Mail: thomas.hoppe@fokus.fraunhofer.de

S. Matini
E-Mail: shirkouh.matini@fokus.fraunhofer.de

I. Mattig · G. Barzen · K. Köhler · H. Dreger · F. Köhler
Deutsches Herzzentrum der Charité, Berlin, Deutschland
E-Mail: isabel.mattig@charite.de

G. Barzen
E-Mail: gina.barzen@charite.de

K. Köhler
E-Mail: kerstin.koehler@charite.de

H. Dreger
E-Mail: henryk.dreger@charite.de

F. Köhler
E-Mail: friedrich.koehler@charite.de

© Der/die Autor(en) 2025

K. Hinkelmann et al. (Hrsg.), *Hybride KI mit Machine Learning und Knowledge Graphs*,
https://doi.org/10.1007/978-3-658-44781-6_9

197

4. Der Einsatz von Bayes'schen Netzen zur Lösung von Anwendungsproblemen erfordert eine intensive Kommunikation zwischen Anwendungsexperten und Entwicklern, um die zu erhebenden Daten bereits frühzeitig im Projekt zu identifizieren.
5. Eine intensive Kommunikation zwischen Medizinern und Informatikern, gerade am Anfang eines Projektes, ist notwendig, um ein gegenseitiges Verständnis der jeweiligen Terminologien, Zusammenhänge und Vorgehensweisen zu schaffen.

9.1 Einleitung

Die Covid-19-Pandemie 2020 hat gezeigt, dass kurzfristige lokale oder längerfristige nationale Krisensituationen zu einer Aus- und Überlastung der intensivmedizinischen Behandlungskapazitäten führen können. Die aktuellen Entwicklungen im Pflegebereich, bei denen Pflegekräfte zur Mangelressource werden, schränken die postoperative Versorgung von Patientinnen und Patienten weiter ein. In diesen Situationen müssen ggf. reguläre medizinische Behandlungen zurückgestellt sowie notwendige intensivmedizinische Behandlungen an die verfügbaren klinischen Kapazitäten und Behandlungsabläufe angepasst werden. Ziel hierbei ist, die Intensivstationen soweit wie möglich zu entlasten und trotzdem eine gute Behandlungsqualität der Patienten zu gewährleisten.

Pandemien stellen natürlich nur eine der möglichen Krisensituationen dar, in denen die Zahl der zu behandelnden Patienten schnell ansteigt und die Behandlungskapazitäten an ihre Grenzen stoßen. Andere Krisen, wie z. B. terroristische Anschläge, Naturkatastrophen mit einer großen Anzahl von Verletzten, aber auch längere Brownout- oder Blackout-Situationen,¹ in denen die Behandlungskapazitäten von Kliniken durch längere Stromunterbrechungen, Versorgungsengpässe der Notstromaggregate oder fehlendes Personal eingeschränkt werden, können die gleichen Konsequenzen nach sich ziehen.

In solchen Krisensituationen kann u. a. die Telemedizin ein Hilfsmittel darstellen, um Patienten mit einem hohen Risiko für eine Verschlechterung ihrer chronischen Erkrankungen von zu Hause aus gesundheitlich zu überwachen und medizinische Institutionen zu entlasten. Zur Prüfung dieses Lösungsansatzes wird aktuell im Rahmen des ResKriVer-Projektes² die klinische Studie „ResKriVer-TAVI“ (DRKS00027842) am Deutschen Herzzentrum der Charité durchgeführt. Es soll untersucht werden, ob und wie stark durch eine telemedizinische Mitbetreuung elektiver Hochrisikopatienten deren

¹Blackout bezeichnet ungeplante Stromausfälle, während als Brownout gezielte regionale Abschaltung zur Stabilisierung des Stromnetzes bezeichnet wird.

²Gefördert vom Bundesministerium für Wirtschaft und Klimaschutz, Fördernummer 01MK21006C.

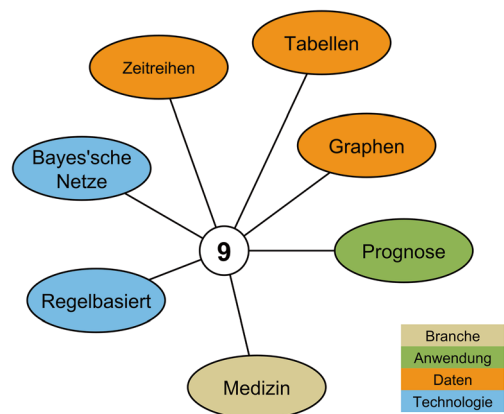
Mortalität während der Wartezeit vor einem geplanten Herzklappeneingriff gesenkt werden kann, um Kliniken in Krisensituationen soweit wie möglich zu entlasten. Die Entlastung würde einerseits durch die Vermeidung einer unnötigen Belegung von Betten im Rahmen der Überwachung des Gesundheitszustandes entstehen, andererseits durch das Hinauszögern der notwendigen Behandlung solange es der Zustand des Patienten erlaubt. Parallel hierzu erarbeitet das Fraunhofer-Institut FOKUS ein Vorhersagewerkzeug zur Priorisierung der Behandlung von telemedizinisch betreuten Patienten anhand der Studiendaten auf der Basis Bayes'scher Netze.

Ziele des Projektes sind hierbei, praktikable Methoden zur Unterstützung des medizinischen Personals in der Phase der telemedizinischen Überwachung zu erarbeiten. Diese Methodik wird zuerst bei Patienten mit Aortenklappenstenosen und Behandlung durch Transkatheter-Aortenklappen-Implantation (TAVI) erprobt. Des Weiteren wird geprüft, wie die Methodik auf andere Patientengruppen übertragen werden kann und welche Voraussetzungen hierzu zu erfüllen sind.

Das Projekt befindet sich zwar noch in der Datenerhebungs- und Modellierungsphase, dennoch sind bereits jetzt erste Erkenntnisse über das Zusammenspiel zwischen dem Design der Datenerhebung, der Modellierung von Bayes'schen Netzen, deren Vorteilen und der Übertragbarkeit auf andere intensivmedizinische Erkrankungen vorhanden, die auf ähnlich gelagerte Probleme übertragen werden können.

Ausgangspunkt des Projektes bilden medizinische Daten über den Gesundheitszustand von Patienten in Form von Tabellen mit Basisdaten und deren regelbasierter Diskretisierung, Zeitreihen der telemedizinischen Überwachung und Parameterabhängigkeiten. Letztere werden zur Modellierung von diskreten Bayes'schen Netzen verwendet, aus denen ein Prognosemodell erlernt wird (Abb. 9.1).

Abb. 9.1 Thematische Einordnung des Kap. 9



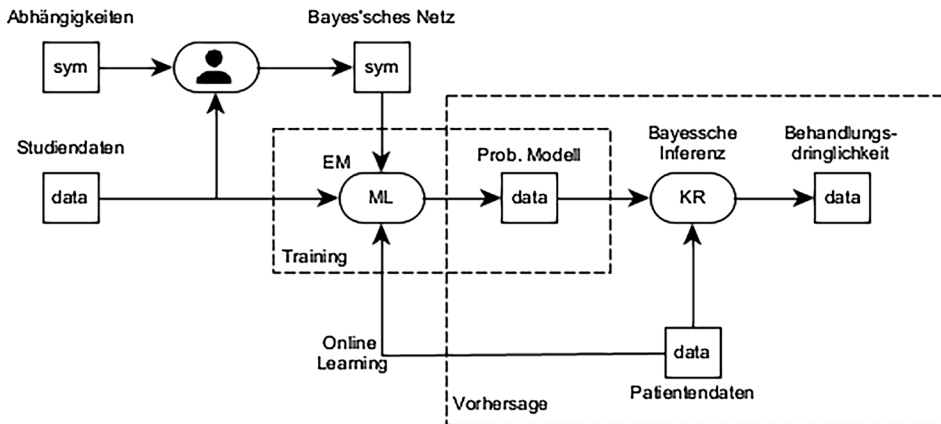


Abb. 9.2 Boxology-Modell des hybriden KI-Systems

9.1.1 Übersicht über das Vorhersageverfahren für elektive Risikopatienten

Mit Boxology haben van Harmelen und ten Teije (2019) und Witschel et al. (2021), wie im Kapitel Einleitung dargestellt, einen Ansatz zur vereinheitlichten Beschreibung von maschinellen Lernverfahren und wissensbasierten Systemen vorgestellt. Das Vorhersageverfahren kann als Boxology-Grafik skizziert werden (Abb. 9.2).

Anhand der erhobenen Studierendaten und der Abhängigkeiten ihrer medizinischen Parameter wird ein diskretes Bayes'sches Netz manuell modelliert. Dieses Netz geht zusammen mit den eigentlichen Studierendaten in den Trainingsprozess des Modells ein, der auf einem Expectation-Maximization-Algorithmus (EM) basiert. Die erlernten Wahrscheinlichkeiten des Modells dienen der Vorhersage der Wahrscheinlichkeit der Notwendigkeit einer Behandlung und deren potenzieller Verweildauern in der telemedizinischen Betreuung anhand der aktuellen Daten von TAVI-Patienten. Die Daten der behandelten Patienten können für das inkrementelle Training des Modells und damit zur Verbesserung der Vorhersagewahrscheinlichkeiten herangezogen werden.

9.2 Anwendungsfall

Als Anwendungsfall wurde die Aortenklappenstenose gewählt. Dabei handelt es sich um eine Herzkrankheit, bei der die Aortenklappe verengt ist. Diese Erkrankung tritt insbesondere bei älteren Patienten auf, beeinträchtigt die Lebensqualität und -erwartung der Patienten stark und ist durch eine Katheter-gestützte Intervention mit kurzem Krankenhausaufenthalt gut behandelbar. Der operative Eingriff – medizinisch als Transkatheter-Aortenklappen-Implantation (TAVI) bezeichnet – erfolgt durch Herzspezialisten in

einem Hybrid-OP (kombinierter Operationssaal, der sowohl Herzkathetereingriffe als auch Herzoperationen ermöglicht). Es gibt in Deutschland nur wenige medizinische Zentren, die die Voraussetzungen für solche hochspezialisierten Eingriffe erfüllen. In Berlin gibt es ein Zentrum am Deutsches Herzzentrum der Charité (DHZC); dementsprechend sind die Kapazitäten für diese Eingriffe regional begrenzt.

Die ResKriVer-TAVI-Studie³ untersucht den Einfluss einer telemedizinischen Mitbetreuung bei Patienten mit hochgradiger Aortenklappenstenose in der Wartezeit auf eine geplante TAVI im Vergleich zur Standardversorgung ohne Telemonitoring. Ziel ist es, am Beispiel der Aortenklappenstenose als Modellerkrankung zu prüfen, inwieweit eine telemedizinische Mitbetreuung bei eingeschränkten Behandlungskapazitäten die Versorgung der Patienten sicherstellen und ggf. sogar verbessern kann. Vorschläge dazu wurden bereits in die Empfehlungen erster europäischer Fachgesellschaften zur Therapie von Herzklappenerkrankungen aufgenommen, um während der SARS-CoV-2-Pandemie eine adäquate Versorgung zu gewährleisten (Plonska-Gosciniak et al. 2019). Während einer kurzfristigen lokalen oder längerfristigen nationalen Krisensituation zählt die Sicherung der medizinischen Infrastruktur zu den prioritären Aufgaben. Digitale Lösungsansätze haben hier ein großes Potenzial, die Versorgungssituation auch unter Krisenbedingungen aufrechtzuerhalten. Bedeutsam ist, dass die zu erarbeitenden Lösungsansätze ebenfalls unter normalen Arbeitsbedingungen eingesetzt werden können. Damit soll erreicht werden, dass Patienten mit chronischen, lebensbegrenzenden Erkrankungen und hoher Operationsdringlichkeit in einem Krankenhaus der Maximalversorgung (z. B. kardiovaskuläre oder onkologische Erkrankungen) jederzeit, auch unter Krisenbedingungen versorgt werden können. Bei Pandemie-, Katastrophen-bedingten oder anderweitig beschränkten Ressourcen für diese Eingriffe befinden sich diese Patienten unbehandelt zu Hause und werden auf einer Warteliste geführt. Mit einem intelligenten, telemedizinisch unterstützten Management soll bei diesen wartenden Patienten eine Übersterblichkeit aufgrund der Kapazitätsbeschränkungen vermieden werden (Köhler et al. 2022).

9.2.1 Aortenklappenstenosen und deren Behandlung durch TAVI

Die Aortenklappe ist eine der vier Herzklappen, die mittels ihrer Ventilfunktion die Richtung des Blutflusses im Herzen steuern, und befindet sich zwischen der linken Hauptkammer des Herzens und der großen Körperschlagader, der Aorta. Sie öffnet und schließt täglich im Mittel etwa 100.000-mal und ist von allen Herzklappen der größten Druckbelastung ausgesetzt. Aufgrund dieser hohen Beanspruchung sind degenerative Veränderungen der Klappe im höheren Lebensalter häufig. Insbesondere Verkalkungen können letztlich zu einer reduzierten Öffnungsbewegung der Klappe mit Abnahme der Auswurfleistung des Herzens führen: Man spricht von einer Aortenklappenstenose (Abb. 9.3). Mit Voranschreiten der Erkrankung führt die Aortenklappenstenose zu einer symptomatischen Herzinsuffizienz; unbehandelt sterben innerhalb von zwei Jahren bis zu 50 % der betroffenen Patienten. Aufgrund der häufigen degenerativen Veränderungen

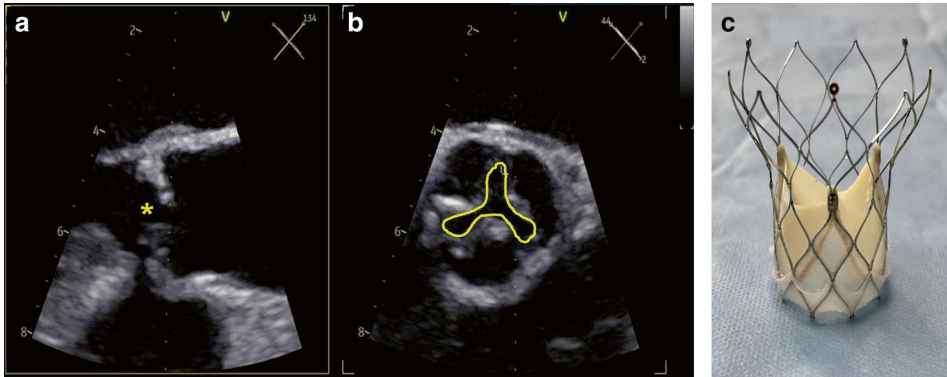


Abb. 9.3 Schluckultraschall (transösophageale Echokardiografie) mit Darstellung der verengten Aortenklappen (*) im Längs- (a) und Querschnitt (b) mit einer hochgradig reduzierten Öffnungsfläche von $0,8 \text{ cm}^2$ (gelbe Umrandung) und (c) Fotografie einer Herzklappenprothese³ (eigene Aufnahmen der Charité)

der Aortenklappe und ihrer schweren Konsequenzen für Lebensqualität und -erwartung der Patienten ist die Aortenklappenstenose der am häufigsten behandelte Klappenfehler in Industrienationen.

Seit vielen Jahrzehnten besteht die Möglichkeit, die Aortenklappe chirurgisch unter Zuhilfenahme einer Herz-Lungen-Maschine durch künstliche Klappenprothesen zu ersetzen. Dieser Eingriff ist weiterhin Mittel der Wahl für jüngere Patienten, stellt aber für ältere Patienten und Patienten mit Begleiterkrankungen eine große Belastung mit erhöhtem Risiko für intra- und postoperative Komplikationen dar. Daher wurde vor inzwischen über 20 Jahren die Transkatheter-Aortenklappen-Implantation (TAVI) entwickelt, wobei eine gefaltete Klappenprothese, meist durch einen wenige Millimeter großen Zugang in der Leiste, zum Herzen vorgebracht und im Bereich der erkrankten Aortenklappe implantiert wird. Die Freisetzung der Katheterklappe dauert nur wenige Sekunden und kann am wachen Patienten ohne Herz-Lungen-Maschine durchgeführt werden. Dementsprechend ist der Eingriff auch bei hochbetagten und schwer erkrankten Patienten durchführbar.

Aortenklappenstenose-Patienten mit geringen und stabilen Beschwerden werden derzeit überwiegend elektiv behandelt, d. h., der Termin der TAVI liegt – meist aus organisatorischen und Kapazitätsgründen – in der Regel 6 bis 12 Wochen nach der Feststellung der Aortenklappenstenose. Ein geringer, aber nicht unbedeutender Teil der Patienten verstirbt jedoch innerhalb dieser Wartezeit am plötzlichen Herztod. So zeigen Studien eine

³Herzklappenprothese des Herstellers Navitor, Abbott Vascular, Santa Clara, CA, USA.

Mortalität von bis zu 5 % bei einer Wartezeit auf den Eingriff von drei Monaten (Albasam et al. 2020).

9.2.2 Telemedizinische Überwachung

Entsprechend des Studiendesigns der ResKriVer-TAVI-Studie werden die Patienten 1:1 randomisiert, d. h., ca. die Hälfte der Patienten wird in der Wartezeit auf den Herzklappeneingriff telemedizinisch betreut. Die andere Hälfte der Patienten (Kontrollgruppe) erhält eine Standardversorgung ohne telemedizinische Mitbetreuung. Das in der ResKriVer-TAVI Studie verwendete Konzept der telemedizinischen Mitbetreuung wird durch den Arbeitsbereich Kardiovaskuläre Telemedizin (TMZ) des DHZC umgesetzt und beinhaltet eine Patientenschulung, das Telemonitoring von Vitalparametern (Körpergewicht, Blutdruck, EKG, Sauerstoffsättigung, Befinden) sowie telemedizinische Interventionen bei Verschlimmerung von Vitaldaten und/oder Symptomen. Im TMZ arbeiten sowohl Fachpflegekräfte als auch Fachärzte eng mit den behandelnden Ärzten der Patienten zusammen. Das in der Studie eingesetzte telemedizinische Mitbetreuungskonzept hat sich bereits in der TIM-HF2-Studie erfolgreich bewährt (Köhler et al. 2018, 2020).

Die Patienten, die eine telemedizinische Mitbetreuung erhalten, werden innerhalb der ersten Woche nach Studienstart durch Mitarbeiter des TMZ zu Hause in die Benutzung der telemedizinischen Heimmessgeräte (Körperwaage, Blutdruckmessgerät, Pulsoxymeter, EKG-Gerät, Tablet zur Datenübertragung) eingewiesen. Außerdem werden die Patienten zum Umgang mit ihrer Herzerkrankung beraten und zu Symptomüberwachung und Selbstmanagement ihrer bestehenden Aortenklappenstenose geschult. Die Patienten können sich bei medizinischen oder technischen Problemen täglich telefonisch an das medizinische Personal des TMZ wenden. Bis zur stationären Aufnahme zur TAVI messen die Patienten einmal täglich eigenständig jeweils das Körpergewicht, den Blutdruck, das EKG, die Sauerstoffsättigung und geben ihr Befinden über ein Tablet ein.

Die Übertragung der Messwerte an das TMZ wird über das Mobilfunknetzwerk gewährleistet und dauert in der Regel etwa 90 s. Im Anschluss werden die Daten in die elektronische Patientenakte automatisiert eingepflegt. Am telemedizinischen Arbeitsplatz des TMZ findet eine IT-gestützte Priorisierung der eingehenden Vitalparameter auf der Basis von Grenzwertverletzungen statt, sodass Patienten mit kritischen bzw. auffälligen Vitalparametern vorrangig bewertet werden können. Die abschließende Beurteilung erfolgt durch Fachpflegekräfte und Fachärzte. Bei Entdeckung auffälliger Befunde werden abhängig von der Schwere der Auffälligkeit durch die Ärzte des TMZ entsprechende Maßnahmen eingeleitet (z. B. Anpassen der medikamentösen Therapie, Empfehlung der Vorstellung beim ambulant behandelnden Arzt in der Notaufnahme oder im Krankenhaus, Verständigung des ärztlichen Bereitschaftsdienstes oder der Feuerwehr, Vorziehen der TAVI). Außerdem werden Patienten, die sich telefonisch mit zunehmenden oder neuen Symptomen an das TMZ wenden, fachmedizinisch beraten und notwendige Maßnahmen werden eingeleitet (s. o.). Insbesondere werden die Vitaldaten regelmäßig

darauflin bewertet, inwieweit der TAVI-Eingriff individuell bei den einzelnen Patienten abgewartet werden kann oder vorgezogen werden muss. Auslöser für ein Vorziehen des TAVI-Eingriffs können z. B. sein: lebensbedrohliche EKG-Veränderungen, anhaltend sehr hohe oder sehr niedrige Blutdruckwerte, eine deutliche Gewichtszunahme in wenigen Tagen, eine deutliche Abnahme der Sauerstoffsättigung im Blut, eine Verschlechterung des Befindens oder eine Zunahme der Herzbeschwerden.

9.2.3 Design der TAVI-Studie

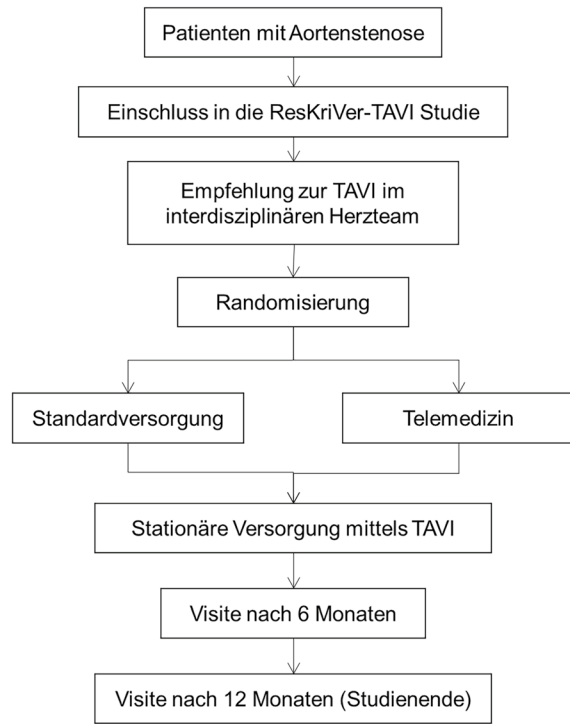
Die Studie „Telemedizinische Mitbetreuung von Patienten mit schwerer Aortenklappenstenose und Bedarf für eine Transkatheter-Aortenklappenimplantation (TAVI) unter Normalbedingungen und bei krisenbedingt eingeschränkten Behandlungskapazitäten“ (Kurztitel: ResKriVer-TAVI, DRKS00027842) wurde 2022 am Deutschen Herzzentrum der Charité initiiert. Gegenstand der ResKriVer-TAVI-Studie ist die Entwicklung digitaler Lösungsansätze für die Aufrechterhaltung der Versorgung von Patienten mit Aortenklappenstenose und Indikation zur TAVI sowohl unter Normalbedingungen als auch unter kurzfristigen lokalen oder längerfristigen nationalen Krisensituationen. Mit einem telemedizinischen, durch Methoden der künstlichen Intelligenz (KI) unterstützten Management während der Wartezeit auf den Herzklappeneingriff sollen bei diesen Patienten eine Übersterblichkeit und zusätzliche Krankenhauseinweisung reduziert werden.

In der prospektiven, randomisierten, kontrollierten Studie werden Patienten mit hochgradiger Aortenklappenstenose, die auf die Versorgung mit einer TAVI warten, eingeschlossen. Ziel ist es, mittels Telemedizin frühzeitig eine Verschlechterung zu erkennen und die Patienten zu identifizieren, die schnellstmöglich mit einer TAVI versorgt werden müssen. Die Einschlusskriterien umfassen

1. die Entscheidung zur TAVI bei hochgradiger Aortenklappenstenose durch ein Behandlungsteam, bestehend aus Kardiologen und Herzchirurgen (als „Herzteam“ bezeichnet),
2. die schriftliche Einwilligung der volljährigen Patienten.

Von der Studienteilnahme ausgeschlossen werden Patienten mit der Unfähigkeit oder der fehlenden Bereitschaft zur täglichen Vitaldatenmessung (z. B. bei Demenz, Blindheit, Gehörlosigkeit, Rollstuhlpflichtigkeit, unzureichenden Deutschkenntnissen und eingeschränkter informeller Selbstbestimmung) und Patienten, die bereits an weiteren Therapiestudien teilnehmen.

Nach Einschluss in die Studie erfolgt die Zuordnung (Randomisierung) der Patienten entweder in die Gruppe der Standardversorgung oder in die Gruppe mit telemedizinischer Mitbetreuung (1:1). Für die Randomisierung wird ein Software-Tool verwendet, welches Kriterien, wie Alter, Geschlecht und Beschwerden (Atemnot), be-

Abb. 9.4 Ablauf der ResKriVer-TAVI-Studie

rücksichtigt und eine gleichmäßige und zufällige Aufteilung der Patienten in die beiden Gruppen gewährleistet. Außerdem findet zum Studienstart eine Patientenuntersuchung (Basisvisite) statt, in der grundlegende medizinische Daten erhoben werden (Abb. 9.4).

Die Standardversorgung beinhaltet keine zusätzliche Betreuung der Patienten. Patienten, die der telemedizinischen Mitbetreuung zugeordnet wurden, werden telemedizinisch überwacht. Diese Überwachung beginnt nach der Herzteam-Entscheidung und dauert bis zur stationären Aufnahme zur TAVI. Weitere Visiten finden bei Aufnahme zur TAVI nach sechs sowie nach zwölf Monaten nach Studieneinschluss statt. Inwieweit eine telemedizinische Betreuung einen Nutzen für die Patienten darstellt, soll nach Studienende anhand der Summe der Tage, die die Patienten beider Studienarme innerhalb von 180 Tagen nach Studieneinschluss außerhalb des Krankenhauses erlebt (also nicht durch eine Krankenhausaufnahme oder vorzeitigen Tod verloren) haben, ermittelt werden.

9.2.4 Daten der TAVI-Studie

Im Rahmen der ResKriVer-TAVI-Studie wird eine Vielzahl an Parametern (aktuell mehr als 190) erhoben. Zunächst erfolgt während der Basisvisite die Erhebung der biologischen Daten wie Alter, Geschlecht, Größe und Gewicht zur Ermittlung des Body-

Mass-Index (BMI). Des Weiteren werden Symptome, die auf eine Aortenklappenstenose und/oder Herzinsuffizienz hinweisen, erhoben. Hierzu gehören Luftnot, das Auftreten von belastungsabhängigen Brustschmerzen oder ein plötzlich auftretender Bewusstseinsverlust, sogenannte Synkopen. Ebenso wird abgefragt, ob Wassereinlagerungen zum Beispiel an den Knöcheln und Unterschenkeln aufgefallen sind. Hieraus können unter anderem Rückschlüsse auf die Dringlichkeit des Eingriffes gezogen werden. Des Weiteren werden die Herzfrequenz und der Blutdruck gemessen. Erfasst werden zudem

- relevante Vorerkrankungen des Herz-Kreislauf-Systems, Herzinfarkt, ggf. mit Bypass-Operation oder Katheter-gestützter Stentimplantation, Bluthochdruck, Herzrhythmusstörungen oder Schrittmacherimplantationen,
- begleitende Erkrankungen mit erhöhtem Operationsrisiko, Lungenerkrankungen oder Zuckererkrankungen (Diabetes mellitus) und
- überstandene oder aktive Krebserkrankungen, Lebererkrankungen oder atherosklerotische Gefäßerkrankungen sowie
- die aktuelle Medikation des Patienten.

Hierbei wird der STS-Score (Society of Thoracic Surgeons) genutzt, der in der klinischen Praxis zur Abschätzung des Sterberisikos bei herzchirurgischen Eingriffen verwendet wird (Frilling et al. 2010). Darüber hinaus werden in der Basisvisite spezifische Herzparameter erhoben sowie Blutwerte, die auf eine Blutarmut, eine eingeschränkte Blutgerinnung oder Nierenfunktion hinweisen, erfasst.

Die Ergebnisse des zur Vorbereitung des TAVI-Eingriffs durchgeführten Herzultraschalls sowie der Herzkatheter-Untersuchung werden ebenfalls dokumentiert. Hier werden insbesondere die Parameter erhoben, die eine Aussage über die Herzklappenfunktion, die Ausprägung der Aortenklappenstenose und die Pumpleistung des Herzens erlauben.

Patienten in der Telemedizin-Kohorte erhalten das bereits beschriebene telemedizinische Monitoring. Hier erfolgt eine tägliche Vitaldatenmessung bis zur stationären Aufnahme zur TAVI. Diese wird nicht in der Standardversorgung durchgeführt.

In den folgenden Studienvisiten werden Veränderungen der medizinischen Befunde seit Studienbeginn und die Lebensqualität der Patienten per Fragebogen erhoben. Zudem werden Komplikationen im Rahmen des TAVI-Eingriffs sowie geplante und ungeplante Krankenhausaufenthalte dokumentiert.

Zu Studienbeginn wird für alle Patienten ein „Basisrisiko“ für das Ergebnis des TAVI-Eingriffs festgelegt. Dafür werden zum einen die zu Studienbeginn erhobenen individuellen medizinischen Parameter verwendet und zum anderen die bereits beschriebenen historischen Daten genutzt.

9.2.5 Historische Daten aus der Corona-Pandemie

Um zusätzliche Daten zu nutzen, werden historische Patientendaten aus dem Zeitraum 2021 bis 2022 ausgewertet. Hierbei werden die Sterberate und Dauer des Krankenhausaufenthaltes im Rahmen des TAVI-Eingriffs betrachtet. Ziel ist es, Risikofaktoren für einen möglichen komplizierten Verlauf zu identifizieren. Da der Erfassungszeitraum der historischen Patientendaten vor der Projektidee lag, sind lediglich Parameter aus der klinischen Routine heraus dokumentiert worden, sodass hierfür nur partiell Daten verwendet werden können.

9.3 Technische Umsetzung

Im Rahmen des ResKriVer-Projekts ist geplant, anhand der Daten der TAVI-Studie eine auf Methoden der künstlichen Intelligenz basierende Lösung zur Vorhersage der Dringlichkeit der Behandlung von Aortenstenose-Patienten exemplarisch zu entwickeln. Idealerweise sollte diese Lösung als Blaupause für die Entwicklung weiterer Lösungen für die telemedizinische Betreuung anderer intensivmedizinischer Patienten einerseits und für den Einsatz in Krisensituationen andererseits ausgelegt sein.

Aufgrund der zeitlichen Begrenzung der ResKriVer-TAVI-Studie war von Anfang an klar, dass nur für eine begrenzte Anzahl von Patienten überhaupt Daten erhoben werden können, sodass eine Lösung zu entwickeln ist, die sich inkrementell an zukünftig verfügbare Patientendaten anpassen kann und die damit auch für die unbekanntesten Konstellationen zukünftiger Krisensituationen adaptiert werden kann. Die Klasse der potenziell nutzbaren Verfahren war daher auf Online-Learning-Verfahren des maschinellen Lernens begrenzt.

Allein anhand der für die ResKriVer-TAVI-Studie berücksichtigten Patienten wird ersichtlich, dass eine auf künstlichen neuronalen Netzen basierende technische Lösung aufgrund der geringen Anzahl Patienten-spezifischer Datensätze nicht nutzbar ist.

Die während des telemedizinischen Monitorings erhobenen Daten bilden Zeitreihen, die vom medizinischen Personal auf Indikatoren untersucht werden, um zu entscheiden, ob Patienten zur TAVI einzubestellen sind. Für die Beurteilung des Patientenzustands sind insbesondere kurzfristige drastische Verschlechterungen als langsamer schleichende Verschlechterungen ausschlaggebend. Im Rahmen des Projekts wurde entschieden, hierfür zunächst mit zwei unterschiedlichen Zeitreihen der jeweils letzten 7 resp. 14 Tage zu arbeiten.

9.3.1 Problemstellung

Geringe Fallzahlen

Die Zahl der Patientendaten könnte weitaus höher sein, sofern auch die Daten der Patienten, die einer Nutzung *nicht* zugestimmt haben, genutzt werden würden,⁴ jedoch werden ihre Daten aus ethischen Gründen nicht erhoben. Zum Zeitpunkt der Verfassung des Kapitels⁵ haben von den 114 Patienten mit Aortenklappenstenosen, die die Ein- und Ausschlusskriterien erfüllen, lediglich 49 Patienten, d. h. rd. 43 %, der Nutzung ihrer Daten für die TAVI-Studie zugestimmt. Hochgerechnet auf die 400 Studienpatienten würden somit lediglich rund 172 Patienten einer Datennutzung zustimmen.

Größe des Hypothesenraums

Die Datensätze werden durch unterschiedliche Parameter, im maschinellen Lernen als Feature bezeichnet, beschrieben. Die Features können als Dimensionen eines hochdimensionalen Datenraums betrachtet werden. Die Größe dieses Datenraums wächst exponentiell mit der Anzahl der Features – sprich Daten – (die Dimensionen des Datenraums) und exponentiell mit der Größe ihrer Wertebereiche (der Kardinalität der Daten).

Im Rahmen der TAVI-Studie werden einerseits kategorielle Daten erhoben als auch Fließkommazahlen. Zur Vereinfachung der Abschätzung der Größe des Hypothesenraums wurden Fließkommazahlen jeweils auf ganze Zahlen gerundet. Tab. 9.1 fasst den Feature-Typ, dessen Kardinalität und die Anzahl der pro Studienphase erhobenen Daten zusammen. Die Tabelle bildet die Basis zur Abschätzung der Größe des Hypothesenraums.

Allein die Anzahl der Parameter und deren Kardinalitäten der Basisvisite und des Monitorings definieren einen Hypothesenraum, dessen Größe anhand der Daten aus Tab. 9.1 mit 2^{666} (7-tägiges Monitoring) resp. 2^{1165} (14-tägiges Monitoring) abgeschätzt werden kann. Zum Vergleich: Die Anzahl potenzieller Go-Partien – ein Spiel, das komplexer ist als Schach – beträgt 2^{169} . Die Anzahl der Atome im Universum wird auf 10^{84} – 10^{89} geschätzt⁶, umgerechnet auf Basis 2 ergibt dies 2^{279} – 2^{295} Atome. Der Parameterraum der TAVI-Studie übersteigt somit deren Komplexität bei weitem.

In diesem Parameterraum mit einem Datenbestand von schätzungsweise maximal 172 Patienten-spezifischen Datensätzen Regularitäten zu finden, führt zu einer hoffnungslosen Übergeneralisierung. Das Problem liegt darin, dass konventionelle Lernverfahren in der Regel lediglich die Daten verwenden, um Regularitäten zu identifizieren, und kein Hintergrundwissen über den Anwendungsbereich nutzen. Traditionelle Lernverfahren scheiden daher für die weitere Betrachtung komplett aus.

⁴Diese Möglichkeit wird von der DSGVO Art. 9 (2).i explizit eingeräumt.

⁵Stand 7.2.2023.

⁶<https://www.swr.de/wissen/1000-antworten/wie-viele-atome-gibt-es-im-universum-100.html> (Letzter Zugriff, 14.7.2023).

Tab. 9.1 Kardinalitäten der Feature (blau) und Anzahl der Feature des jeweiligen Datentyps (grün)

Feature		Anzahl der pro Studienphase erhobenen Daten						
Datentyp	Kardinalität	Basisvisite	Monitoring			Zwischenvisite		Endvisite
			Tägl	7d	14d	1	2	
Datum/Ti-meStamp		4	2	2	2	3	1	1
Kategoriiell	2	14	1	7	14	1	12	
	3	27	17	119	238	17	16	2
	4	4	2	14	28	1	3	
	5		1	7	14	3		
Ganzzahlig	5	1						
	7	2						
	20	3						
	30	3	1	7	14			
	60	1						
	80	1						
	100	1						
	120	1				1	1	
	150	1						
	170	1	1	7	14	1	1	
	180	1				1	1	
	200	2					1	
	300	3	3	21	42	3	3	
	1000	1					1	
	100.000	2					2	
m. Klärungsbedarf			1	1	1	3	1	1
Anzahl der Daten pro Phase		73	29	186	368	32	46	4

Technologische Basis: Bayes'sche Netze

Bayes'sche Netze – auch als gerichtete probabilistische grafische Modelle bezeichnet – sind eine seit den 1980er-Jahren entwickelte Form von trainierbaren Netzwerken, die Hintergrundwissen in Form von Parameterabhängigkeiten nutzen, um den Hypothesenraum stark zu verkleinern. Diese Netzwerke berechnen Wahrscheinlichkeiten von Ereignissen mittels des Satzes von Bayes. Dieser Satz erlaubt es, die vormalig aus Daten berechneten Wahrscheinlichkeiten von Zufallsvariablen oder Annahmen über deren

Wahrscheinlichkeiten – als A-priori-Wahrscheinlichkeiten bezeichnet – an neue Daten anzupassen, wenn diese verfügbar werden. Hierdurch ist dieser Ansatz inhärent inkrementell und für ein Online-Learning-Verfahren einsetzbar. Darüber hinaus ermöglicht es dieser Ansatz, bereits mit wenigen Daten Vorhersagen zu treffen, die jedoch eine große Varianz besitzen, welche mit zusätzlichen Daten verringert werden kann.

9.3.2 Geplante Architektur und Vorgehensweise

Zwar ist die Datenmenge für die Umsetzung eines klinisch nutzbaren Prognoseverfahrens für telemedizinisch betreute, elektive Aortenklappenstenose-Patienten noch zu gering, dennoch repräsentiert diese Situation die allgemeinen Rahmenbedingungen für die Umsetzung eines Prognoseverfahrens für telemedizinisch betreute Risikopatienten. Ausgehend von einer initialen Modellierung auf der Basis einer begrenzten Anzahl von Patienten kann ein initiales Prognosemodell konstruiert und trainiert werden, welches im Rahmen des Einsatzes zusätzlicher Patientendaten kontinuierlich weiter trainiert werden kann. Initial wird dieses Modell eine geringe Genauigkeit besitzen, die sich aber durch zusätzliche Fälle kontinuierlich der Wirklichkeit anpassen und Korrelationen zwischen den medizinischen Parametern zuverlässiger erfassen sollte.

Solange dieses Verfahren keine eigenständigen Entscheidungen trifft, sondern von Medizinern zur Priorisierung von Patienten genutzt wird, ist die Gefahr von automatisierten Fehlentscheidungen (insbesondere von False Negatives, Patienten, die der Behandlung bedürfen, die aber nicht vorgeschlagen werden) gering.

Architektur

Zur Reduktion des Hypothesenraums wurden die 73 Parameter der Basisvisite priorisiert und für eine erste Modellierung genutzt. Für eine erste Überprüfung der Vorhersagbarkeit der Verweildauer von TAVI-Patienten in der telemedizinischen Betreuung wurden die numerischen Parameter entsprechend gängiger klinischer Kategorien diskretisiert. Für Patienten, die umgehend behandelt werden müssen, ist eine Vorhersage der Verweildauer nicht notwendig; eine telemedizinische Betreuung erfolgt erst nach mehr als 3 Tagen. Die Verweildauer wurde daher in die Intervalle 3–5 Tage, 6–10 Tage, 11–15 Tage und 16 Tage und mehr unterschieden.

Zur Validierung, ob die Prognose der Verweildauer von Patienten auf Grundlage der in der Basisvisite erhobenen Daten möglich ist, wurde ein erstes Bayes'sches Modell abgeleitet (Abb. 9.5) und mit Daten⁷ von 41 von 47 Patienten, die während der TAVI-Studie telemedizinisch betreut und letztendlich behandelt wurden, trainiert.

Die Umsetzung der diskreten Bayes'schen Netze erfolgt mit dem System GeNIe Modeller von BayesFusion.⁸

⁸<https://www.bayesfusion.com/genie/> (letzter Zugriff 14.8.2023).

⁷Stand 7.2.2023.

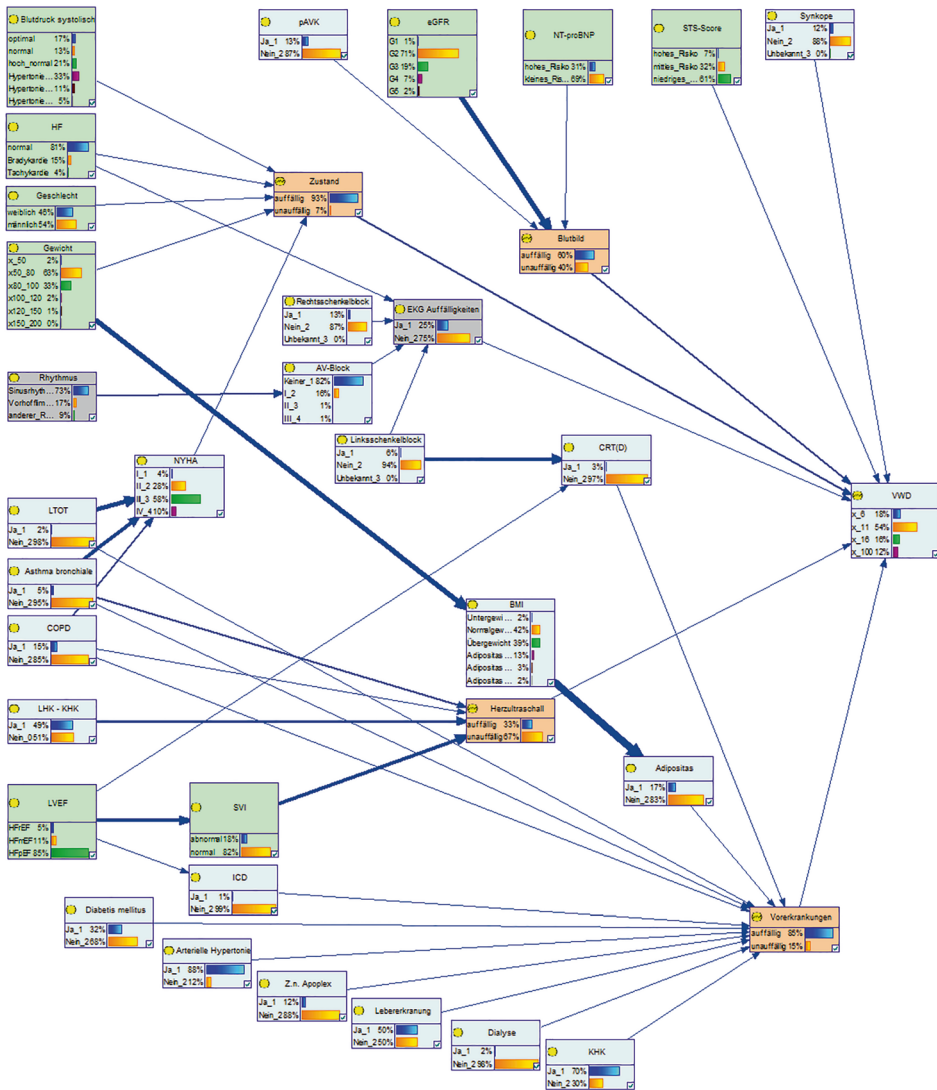


Abb. 9.5 Modell priorisierter Parameter zur Vorhersage der Verweildauer in telemedizinischer Betreuung

Das Bayes'sche Modell in Abb. 9.5 umfasst Zufallsvariablen (ZV) unterschiedlicher Typen: hellblaue Knoten stellen kategorielle ZV dar, grüne Knoten repräsentieren diskretisierte, numerische ZV, orange Knoten sind latente Zufallsvariablen, für die keine Mess-

werte vorliegen, die aber der Vereinfachung des Modells dienen. Pfeile geben die Abhängigkeiten der durch Zufallsvariablen repräsentierten TAVI-Parameter wieder.

Die latenten Variablen entsprechen einer initialen Gruppierung der Mediziner, so dass sie aus medizinischen Gesichtspunkten gerechtfertigt sind und helfen, auffällige von unauffälligen Parametern zu unterscheiden. Sie wurden als Noisy-Max-Knoten modelliert, um die Wahrscheinlichkeiten ihrer Elternknoten zu aggregieren. Der Knoten „Vorerkrankungen“ rechts unten beispielsweise repräsentiert, wie auffällig die Vorerkrankungen des Patienten sind. Patienten mit vielen Vorerkrankungen werden damit als auffälliger eingestuft als Patienten mit wenigen Vorerkrankungen.

Zur Analyse wurde das trainierte Netz einer Sensitivitätsanalyse unterzogen und die Einflussstärke der Abhängigkeiten zwischen den ZV visualisiert (Abb. 9.6). Die Stärke der Pfeile gibt hierbei die durchschnittliche Stärke des Einflusses (GeNIe 2023, S. 344 ff.) der unterschiedlichen Zustände der Zufallsvariablen wieder, der sich aus den Daten ergibt. Offensichtlich spiegelt sich der starke Einfluss des Body-Mass-Index (BMI) auf das Vorliegen einer Adipositas und dessen Abhängigkeit vom Gewicht des Patienten auch in den Daten wider. Auch wenn dieser Zusammenhang aus medizinischer Perspektive nahezu tautologisch ist, ist er dennoch ein wichtiger Indikator dafür, dass dieser Zusammenhang in den Daten richtig erkannt wurde. Im Umkehrschluss kann gefolgert werden, dass für Zufallsvariablen, für die nur ein schwacher Einfluss dargestellt wird (dünne Pfeile), auch in den Daten kaum Belege für Regularitäten identifizierbar sind.

Die Vorhersage der Verweildauer reagiert auf die medizinischen Parameter unterschiedlich sensitiv, das bedeutet, dass Veränderung der medizinischen Parameter unterschiedlich starke Änderung der prognostizierten Verweildauer nach sich zieht (GeNIe 2023, S. 356 ff.). Abb. 9.6 stellt den Grad der Sensitivität der abhängigen Parameter auf Änderungen in den Vorgängern als Rotschattierungen der Vorgänger dar. Je intensiver der Rotton, desto sensitiver reagieren die abhängigen Parameter.

Initiale Validierung

Eine erste 20-fache Kreuzvalidierung auf den 41 von 47 Patientendaten, für die die Verweildauer bekannt war, ergab hierbei eine Genauigkeit (Accuracy) von 75,6 % für die Prognose der Verweildauer. Wobei festzuhalten ist, dass von den 41 Fällen 31 Fälle in die Kategorie Verweildauer 6–10 Tage fallen, die alle korrekt klassifiziert werden. Allerdings wurde keine der anderen Kategorien von Verweildauern 3–5 Tage, 11–15 Tage und 16 Tage und mehr prognostiziert.

Ein erster Versuch, anstelle eines Bayes'schen Netzes ein konventionelles Random-Forest-Verfahren einzusetzen, ergab bei einer 10-fachen Kreuzvalidierung auf den gleichen Daten eine Genauigkeit von 51,5 % mit einer Standardabweichung von 0,122.

Offensichtlich liegt hier ein starkes Ungleichgewicht in der Verteilung der Kategorien in den Daten der Verweildauer vor. Die üblichen Verfahren des Down-Samplings oder Up-Samplings sind hier nicht anwendbar, da die Fallmenge bisher zu gering ist resp.

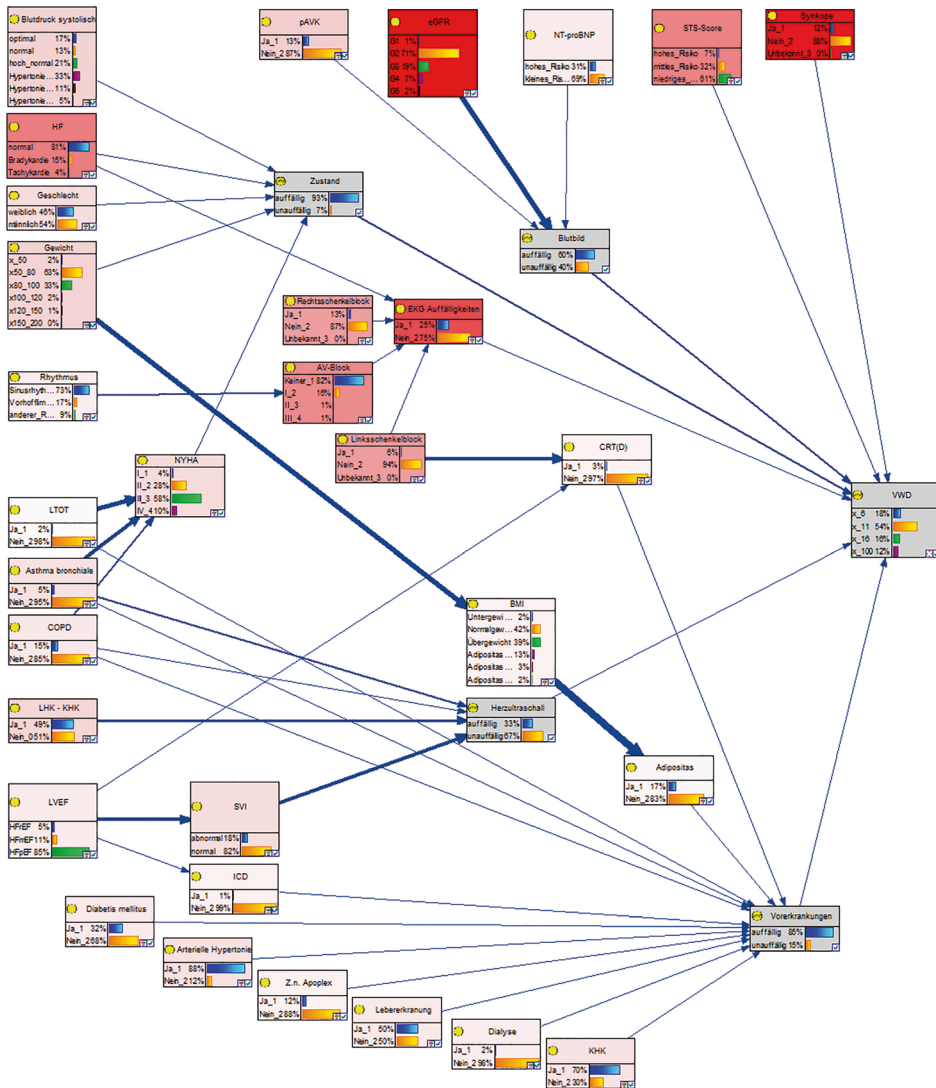


Abb. 9.6 Sensitivität der Parameter bezüglich der Verweildauer

die Generierung von Fällen für die minder repräsentierten Kategorien nicht die Realität widerspiegeln würde.

Validierung mit retrospektiven Daten

In einer weiteren 20-fachen Kreuzvalidierung mit 264 von 266 Patientendaten konnte eine Genauigkeit (Accuracy) von 53,4 % für die Prognose der Verweildauer erzielt werden. Auch hier ist festzuhalten, dass von den 264 Fällen, für die die Verweildauer be-

kannt war, 141 Fälle in die Kategorie Verweildauer 6–10 Tage korrekt klassifiziert werden konnten und für keine der anderen Kategorien eine korrekte Prognose erzielt werden konnte.

Auch mit diesen Daten konnte mit der Verwendung eines konventionellen Random-Forest-Verfahrens bei einer 20-fachen Kreuzvalidierung auf den gleichen Daten lediglich eine Genauigkeit von 52,3 % mit einer Standardabweichung von 0,04 erzielt werden.

Folgerung

Beide Experimente deuten darauf hin, dass eine Vorhersage der diskretisierten Verweildauer nicht erfolgreich sein wird. Darüber hinaus war ein zusätzlicher Versuch, die Verweildauer der retrospektiven Daten tagesgenau zu prognostizieren, ebenfalls nicht erfolgreich; die Genauigkeit sank auf 14,4 % mit einer sehr ungleichen Verteilung der Vorhersagegenauigkeit für eine Verweildauer zwischen 5–9 Tagen. Dies deutet darauf hin, dass es in den 264 Patientendaten noch keine verlässlichen Korrelationen unter den priorisierten Parametern der Basisvisite gibt, die verlässliche Prognosen der Verweildauer erlauben.

Dies erscheint auch nicht verwunderlich, da die Basisvisite lediglich eine Momentaufnahme des initialen Zustands eines Patienten darstellt. Im Wesentlichen wird die Verweildauer in der telemedizinischen Betreuung vom Verlauf der Erkrankung des Patienten bestimmt, sodass für Vorhersagen der Verweildauer der Patienten in der telemedizinischen Betreuung oder der Einbestellung zur TAVI die gesundheitliche Entwicklung des Patienten herangezogen werden muss.

9.3.3 Modellierung des telemedizinischen Monitorings

Während des telemedizinischen Monitorings werden die Patienten kontinuierlich, das heißt durch tägliche Messung und Beurteilung ihres Gesundheitszustands überwacht. Hierbei werden EKG, Gewichtsveränderungen, systolischer und diastolischer Blutdruck, Herzfrequenz und die Sauerstoffsättigung des Blutes vom Patienten zu Hause selbstständig gemessen. Diese Daten werden mindestens einmal täglich zusammen mit einer Selbsteinschätzung seines Gesundheitszustandes an das TMZ der Charité zur Beurteilung von Zustandsveränderungen des Patienten übermittelt.

Die Beurteilung des Patientenzustands erfolgt hierbei auf Basis der zeitlichen Entwicklung der erhobenen Parameter und insbesondere der Überschreitung parameterspezifischer Schwellenwerte. Abb. 9.7 stellt die numerischen Schwellenwerte dar. Überschreiten die Parameter kritische Werte, kontaktieren die Ärzte den Patienten und entscheiden über die Einbestellung zur TAVI.

Eine automatische Vorbeurteilung kann hierbei die Ärzte entlasten, sodass sie sich nur noch auf Patienten konzentrieren können, deren Entwicklung des Gesundheitszustands auffällig bzw. kritisch wird.

Parameter	Normwerte	1. Schwellenwert	2. Schwellenwert
<input checked="" type="checkbox"/> Puls aus RR-Messung	<input type="checkbox"/>	<input checked="" type="checkbox"/> < 50 bpm oder > 100 bpm	<input checked="" type="checkbox"/> < 40 bpm oder > 120 bpm
<input checked="" type="checkbox"/> Blutdruck	<input type="checkbox"/>	Systolisch: < 90 mmHg oder > 140 mmHg Diastolisch: < 40 mmHg oder > 90 mmHg	Systolisch: < 80 mmHg oder > 160 mmHg Diastolisch: < 30 mmHg oder > 100 mmHg
<input checked="" type="checkbox"/> Gewicht	<input type="checkbox"/>	> 1 kg in 1 Tag > 2 kg in 3 Tagen > 2,5 kg in 8 Tagen	> 1,5 kg in 1 Tag > 3 kg in 3 Tagen > 4 kg in 8 Tagen
<input checked="" type="checkbox"/> Sauerstoffsättigung	<input type="checkbox"/>	< 94 %	< 89 %
Wie fühlen Sie sich heute im Vergleich zu Ihrem üblichen Gesundheitszustand?	<input type="checkbox"/>	≥ 4 ↓ 3 Tage in Folge	≥ 5 ↓ 3 Tage in Folge
<input checked="" type="checkbox"/> Keine eingegangene Messung	<input checked="" type="checkbox"/>	Nach 1 Tagen	
<input checked="" type="checkbox"/> EKG eingegangen	<input checked="" type="checkbox"/>		

Abb. 9.7 Schwellenwerte der numerischen Parameter der telemedizinischen Überwachung. (Quelle: SEMDATEX GmbH, inCareNet HF Plattform⁹)

Durch die Überprüfung der Überschreitung der Schwellenwerte können die numerischen Parameter in die Werte „normal“, „auffällig“ und „kritisch“ diskretisiert werden, sodass – technisch betrachtet – ein dynamisches Bayes'sches Modell für die Vorhersage der Wahrscheinlichkeit zur Einbestellung zur TAVI und der wahrscheinlichen Verweildauer des Patienten im TMM eingesetzt werden kann.

Abb. 9.8 zeigt die Struktur des modellierten, dynamischen Bayes'schen Netzes. Ausgangspunkt bilden die während der Basisvisite erhobenen Parameter (links), von denen die initial während des telemedizinischen Monitorings erhobenen Parameter abhängen (Mitte). Diese Parameter hängen vom Zustand am vorausgegangenen Tag ab; dies wird durch die selbstreferenziellen Pfeile an jedem der Parameter in der Mitte modelliert. Sie bilden – bei diesem Modell – eine 7-tägige Zeitreihe, an deren Ende die Wahrscheinlichkeit zur Einbestellung der Patienten ermittelt wird (rechts). Hierfür werden Auffälligkeiten im EKG und Schwellenwertüberschreitung in latenten Variablen – ebenfalls als Noisy-Max-Knoten modelliert – zusammengefasst.

Im Gegensatz zu konventionellen Ansätzen der Zeitreihenanalyse und -vorhersage (Hyndman und Athanasopoulos 2018), die über beliebig lange Zeiträume Analysen und Vorhersagen ermöglichen und Punktschätzungen mit Konfidenzintervallen auf einfachen numerischen Zeitreihen erlauben, können dynamische Bayes'sche Netze aus Komplexitätsgründen nur für feste Zeiträume genutzt werden. Sie erlauben es jedoch auch, multivariate Zeitreihen und insbesondere Abhängigkeiten der Parameter zu verarbeiten.

Zum Training des Modells werden die Patientendaten über ein Sliding-Window-Verfahren¹⁰ aufbereitet. Hierbei werden aus den Zeitreihendaten der Parameter jeweils aufeinanderfolgende Sequenzen der Daten (Window) gebildet, deren Länge dem dynami-

⁹Mit freundlicher Genehmigung der SEMDATEX GmbH.

¹⁰Mitunter auch als Rolling-Window oder Moving-Window bezeichnet.

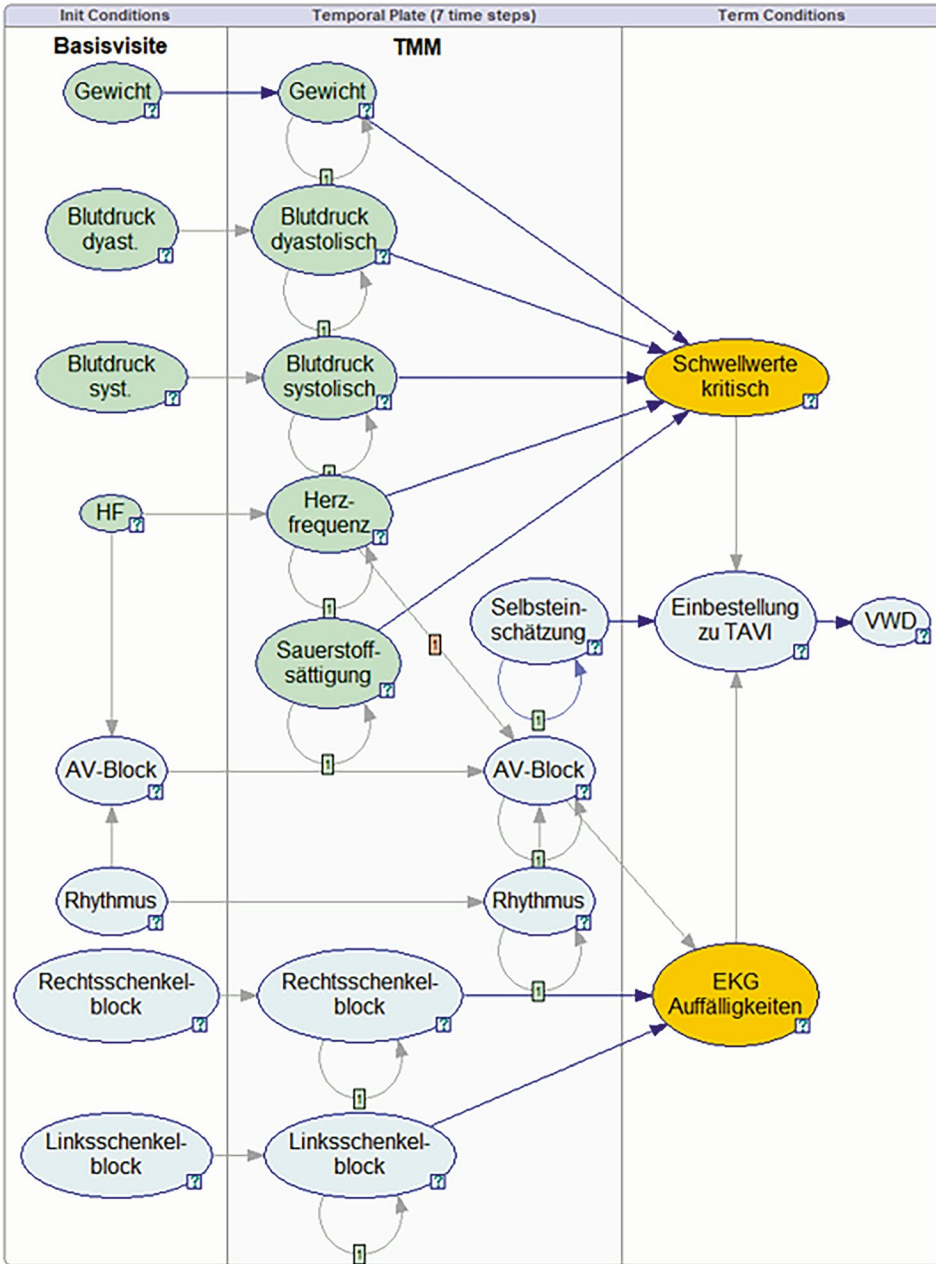


Abb. 9.8 Dynamisches Bayes'sches Modell zur Vorhersage auf der Basis 7-tägiger Zeitreihen des telemedizinischen Monitorings

schen Teil des Bayes'schen Netzes – hier 7 Tage – entspricht. Hierbei sind drei Fälle zu unterscheiden:

1. **Negative Sequenzen:** Am Ende der Sequenz wurde der Patient noch nicht zur TAVI einbestellt.
2. **Positive Sequenzen.** Am Ende der Sequenz wurde der Patient zur TAVI einbestellt.
3. **Unvollständige Sequenzen:** Der Patient war weniger als 7 Tage in telemedizinischer Betreuung.

Die ersten beiden Fälle sind unproblematisch umsetzbar. Im letzten Fall sind zusätzliche Maßnahmen erforderlich. Die Entscheidung, TAVI oder nicht, erfolgt bei der Verwendung dieses Verfahrens immer am Ende einer Sequenz, d. h., dass alle verfügbaren Messungen in den vorausgegangenen Tagen zu betrachten sind. Für den 3. Fall folgt daraus, dass

- a) am Anfang der Sequenz Daten fehlen und
- b) im Modell die ersten verfügbaren Messungen nicht mehr direkt von den Daten der Basisvisite abhängen.

Beide Probleme können dadurch gelöst werden, dass die Daten der Basisvisite bei der Aufbereitung in das Fenster verschoben werden und quasi als erster Tag der Sequenz betrachtet werden.

Konsequenz

Aus dieser Datenaufbereitung ergibt sich, dass es sehr wahrscheinlich mehr negative als positive Beispielsequenzen geben wird. Dies hat zur Folge, dass Situationen, in denen Patienten nicht einbestellt werden müssen, vermutlich mit größerer Sicherheit vorhergesagt werden als Situationen, in denen sie einzubestellen wären.

9.4 Empfehlungen

Zusammenfassend können folgenden Empfehlungen ausgesprochen werden:

1. Die Bildung eines interdisziplinären Projektteams, das nicht nur aus Medizinern und Informatikern besteht, sondern auch Medizininformatiker umfasst, die auf der klinischen Seite des Projekts arbeiten, hilft Reibungsverluste durch die unterschiedlichen Sichtweisen von Medizinern und Informatikern zu verringern.

2. Nur einem kleinen Teil der Mediziner scheint die Vorgehensweise der Bayes'schen Statistik, Analyse und Vorhersage vertraut zu sein, sodass am Projektanfang diese durch eine kurze Schulung hervorgehoben werden sollten.
3. Zeitliche Freiräume zum intensiven, interdisziplinären Austausch sollten gerade am Projektanfang zur Entwicklung einer gemeinsamen Verständnisbasis geschaffen werden. Dies setzt insbesondere bei Medizинern voraus, dass sie teilweise von klinischen Abläufen befreit werden.
4. Ein für beide Disziplinen verbindliches Data Dictionary sollte zum Projektbeginn vereinbart werden, um die Interpretation und die Übernahme von medizinischen Daten zu vereinfachen.
5. Die Erhebung von Abhängigkeiten zwischen den medizinischen Parametern und den Medizinern geläufigen Kategorisierungen von Parametern erweisen sich einerseits für das Verständnis der medizinischen Zusammenhänge durch die Informatiker als hilfreich, andererseits ermöglichen sie die Modellierung des Bayes'schen Netzes unabhängig von der zeitlich eingeschränkten Verfügbarkeit der Mediziner.
6. Eine zu große Genauigkeit numerischer Parameter ist häufig nicht nötig. Zur Komplexitätsreduktion sollte ihre Diskretisierung auf der Basis medizinischen Wissens erfolgen.
7. GeNIe erweist sich als ein hilfreiches, ausgereiftes Werkzeug, sowohl für die Modellierung unterschiedlicher Arten probabilistischer, grafischer Modelle, deren Training, Evaluation und Analyse. Bedingt durch seine historische Entwicklung, die nicht mehr zeitgemäße Verwendung eines englischsprachigen ASCII-Zeichensatzes für Bezeichner von Zufallsvariablen und deren Zustände und einige fehlende Möglichkeiten zur Datenmanipulation ist der Umgang mit diesem Werkzeug etwas gewöhnungsbedürftig.

9.5 Fazit

Das ResKriVer-Projekt startete im Juni 2021 nach der ersten Phase der Covid-19-Pandemie. Zwar begann sich das Leben langsam wieder zu normalisieren, Arbeitstreffen waren aber bedingt durch staatliche Vorgaben und die Schutzmaßnahmen in der Charité und bei Fraunhofer FOKUS nur in stark eingeschränktem Maß möglich.

Im Rahmen der Kooperation hat sich gezeigt, dass der Kommunikation der Problemstellung, der Daten und des notwendigen Hintergrundwissens über die medizinischen Zusammenhänge von Erkrankungen Raum eingeräumt werden muss. Idealerweise sollte dieser Austausch zu Beginn eines Projektes sehr intensiv erfolgen. Leider standen diesem Austausch die durch die Covid-19-Pandemie bedingten Einschränkungen entgegen, sodass der Austausch vornehmlich per E-Mail erfolgte.

Insbesondere am Anfang wäre die gemeinsame Festlegung eines Data Dictionary hilfreich gewesen, indem die betrachteten Parameter, deren Bedeutung und deren Wertebereiche verbindlich festgelegt werden.

Die Arbeiten zur Vorhersage von elektiven, telemedizinisch überwachten Patienten finden im Rahmen des noch laufenden ResKriVer-Projektes statt. Im Rahmen der TAVI-Studie des Charité-Teilprojekts werden Daten von an Aortenklappenstenosen erkrankten Patienten weiter erhoben, sodass die Anzahl verfügbarer Datensätze kontinuierlich steigt. Die Daten des telemedizinischen Monitorings befinden sich derzeit noch in der Phase der Aufbereitung, sodass zum Zeitpunkt des Schreibens des Kapitels noch keine weitergehenden Aussagen über die Umsetzbarkeit der Vorhersage resp. ihre Qualität möglich sind.

9.6 Ausblick

Insbesondere das entstehende Ungleichgewicht zwischen den negativen und positiven Trainingssequenzen muss anhand realer Daten – die sich derzeit in der Aufbereitung befinden – überprüft werden. Sollte sich herausstellen, dass dieses Ungleichgewicht einen großen Einfluss auf die Vorhersage besitzt und Negativsituationen sicherer resp. häufiger vorhergesagt werden, ist zu überlegen, welche der folgenden Maßnahmen Abhilfe schaffen können:

- **Up-Sampling der positiven Beispiele**

Technisch ist es möglich, die Unausgeglichenheit der Anzahl der positiven und negativen Sequenzen durch eine Vervielfachung der positiven Sequenzen durch identische Kopien auszugleichen.

- **Nutzung der zuverlässigeren Vorhersage negativer Situationen**

Ziel des intendierten Verfahrens ist es, die Patienten zu identifizieren, die die höchste Priorität resp. die größte Wahrscheinlichkeit zur Einbestellung zur Behandlung haben, sodass Mediziner die notwendigen Maßnahmen bei einer Verschlechterung des Patientenzustands ergreifen können. Wenn die Vorhersage der Patienten, die nicht zur Einbestellung in Betracht zu ziehen sind, zuverlässiger ist, könnte dieses Vorhersageprinzip durch ein Ausschlussprinzip ersetzt werden und die Patienten, die höchstwahrscheinlich nicht der Behandlung bedürfen, von der Begutachtung durch die Mediziner ausgeschlossen werden.

Welches dieser beiden Verfahren zweckmäßiger ist, bleibt mit den medizinischen Kolleginnen und Kollegen nach der Validierung der Zeitreihenvorhersage anhand der realen Daten noch zu diskutieren.

Ob und inwieweit die von der Kooperation zwischen Charité und Fraunhofer FOKUS anhand der telemedizinischen Betreuung von TAVI-Patienten entwickelte Lösung auf andere telemedizinisch betreute Risikoerkrankungen übertragbar ist, ist noch zu untersuchen. Sicherlich werden hierfür andere medizinische Parameter und deren Abhängigkeiten voneinander eine große Rolle spielen. Das Grundprinzip, Erfassung der relevanten Parameter, Erarbeitung der Parameterabhängigkeiten, Entwicklung eines angepassten

Modells mit anschließendem Training und dessen Evaluation, scheint jedoch übertragbar zu sein.

Literatur

- (Albassam et al., 2020) Albassam O, Henning KA, Qiu F, Cram P, Sheth TN, Ko DT, et al. Increasing Wait-Time Mortality for Severe Aortic Stenosis: A Population-Level Study of the Transition in Practice From Surgical Aortic Valve Replacement to Transcatheter Aortic Valve Replacement. *Circ Cardiovasc Interv.* 2020;13(11):e009297
- (van Harmelen & ten Teije, 2019) “A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems”, Frank van Harmelen, Annette ten Teije, *Journal of Web Engineering*, Vol. 18 1–3, 97–124. <https://doi.org/10.13052/jwe1540-9589.18133>
- (Witschel et al., 2021) “Visualization of patterns for hybrid learning and reasoning with human involvement”, Hans Friedrich Witschel, Charuta Pande, Andreas Martin, Emanuele Laurenzi, Knut Hinkelmann, In: Dornberger, R. (eds) *New Trends in Business Information Systems and Technology. Studies in Systems, Decision and Control*, vol 294. Springer, Cham. https://doi.org/10.1007/978-3-030-48332-6_13
- (Plonska-Gosciniak et al., 2020) Plonska-Gosciniak E, Suwalski P, Bartus S, Kukulski T, Komar M, Wojakowski W, et al. Management of valvular and structural heart diseases during the coronavirus disease 2019 pandemic: an expert opinion of the Working Group on Valvular Heart Diseases, the Working Group on Cardiac Surgery, and the Association of Cardiovascular Interventions of the Polish Cardiac Society. *Kardiol Pol.* 2020;78(5):498–507.
- (Köhler et al., 2022) <https://drks.de/search/de/trial/DRKS00027842>, letzter Zugriff am 14.8.2023
- (Köhler et al., 2018) Köhler F, Köhler K, Deckwart O, et al., Efficacy of telemedical interventional management in patients with heart failure (TIM-HF2): a randomised, controlled, parallel-group, unmasked trial. *Lancet* 392:1047-1057, [https://doi.org/10.1016/S0140-6736\(18\)31880-4](https://doi.org/10.1016/S0140-6736(18)31880-4) (Letzter Zugriff, 14.7.2023)
- (Köhler et al., 2020) Köhler F, Köhler K, Prescher S, et al., Mortality and morbidity 1 year after stopping a remote patient management intervention: extended follow-up results from the telemedical interventional management in patients with heart failure II (TIM-HF2) randomised trial. *Lancet Digit Health* 2:e16-e24, [https://doi.org/10.1016/S2589-7500\(19\)30195-5](https://doi.org/10.1016/S2589-7500(19)30195-5) (Letzter Zugriff, 14.7.2023)
- (Frilling et al., 2010) Birgit Frilling, Wolfgang von Renteln-Kruse, Friedrich-Christian Rieß, Evaluation of Operative Risk in Elderly Patients Undergoing Aortic Valve Replacement: The Predictive Value of Operative Risk Scores. *Cardiology* 116:213–218. 2010.
- (GeNIe, 2023) GeNIe Modeler – User Manual, Version 4.0.R6, 19.5.2023, BayesFusion, LLC, <https://support.bayesfusion.com/docs/GeNIe/>
- (Hyndman & Athanasopoulos, 2018) *Forecasting – Principles and Practice*, Rob J. Hyndman, George Athanasopoulos, OTexts, 2018.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Transfer-Lernen für die Klassifikation medizinischer Texte

10

Daniel Bruneß, Matthias Bay, Christian Schulze, Michael Guckert
und Mirjam Minor

Kernaussagen

1. Die Nutzung von Ontologien ermöglicht datenschutzkonformes Transfer-Lernen.
2. Auch Anwendungsgebiete mit wenig Daten können für Machine-Learning-basierte Klassifikation erschlossen werden.
3. Das ontologiebasierte Transfer-Lernen erlaubt die automatische Adaption an einen sich wandelnden Sprachgebrauch.
4. Die Anwendungen zur Klassifizierung von medizinischen Texten eignen sich besonders als Human-in-the-Loop-Assistenzsysteme.
5. Experimente mit Patientenakten aus Krankenhäusern und medizinischen Forentexten sind tauglich für den Produktiveinsatz der Lösung.

D. Bruneß (✉) · C. Schulze · M. Guckert
KITE, Technische Hochschule Mittelhessen, Friedberg, Deutschland
E-Mail: daniel.bruness@kite.thm.de

C. Schulze
E-Mail: christian.schulze@mnd.thm.de

M. Guckert
E-Mail: michael.guckert@mnd.thm.de

M. Bay
Synpulse8, Synpulse Deutschland GmbH, Düsseldorf, Deutschland
E-Mail: matthias.bay@synpulse.com

M. Minor
Institut für Informatik, Professur Wirtschaftsinformatik, Goethe-Universität,
Frankfurt am Main, Deutschland
E-Mail: minor@informatik.uni-frankfurt.de

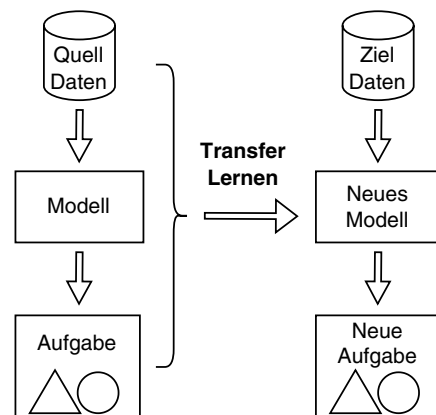
10.1 Einleitung

Die Klassifikation von Dokumenten ist eine typische Aufgabe des Natural Language Processing (NLP). Traditionelle Ansätze trainieren Machine-Learning-(ML-)Modelle auf Statistiken, die aus den Texten extrahiert wurden, oder verwenden geeignete Repräsentationen von Wörtern oder Sätzen für die Nutzung von Deep-Learning-(DL-)Modellen. Basis der Verfahren sind hinreichend große Datenmengen, die für das überwachte Trainieren (Supervised Learning) der Modelle zur Verfügung stehen müssen. Allerdings existieren relevante Anwendungsfälle, für die eine derartige Datenbasis nicht existiert oder nicht zur Verfügung gestellt werden kann. Ein Grund hierfür können zum Beispiel datenschutzrechtliche Restriktionen sein. Dies gilt offensichtlich im Besonderen für sensible Bereiche, wie etwa für medizinische Daten. Eine vollständige Anonymisierung eines Datensatzes kann nicht immer garantiert werden und in Einzelfällen kann von trainierten Modellen auf konkrete Daten zurück geschlossen werden (Song und Mittal 2021). Sollen etwa Dokumente aus Einrichtungen wie Krankenhäusern klassifiziert werden, schließt dieser Umstand sowohl die Nutzung von Dokumenten mehrerer Krankenhäuser zum Training der Modelle als auch den Transfer von in einem Krankenhaus trainierten Klassifikationsmodellen aus. Für eine Wiederverwendung der Klassifikationsmodelle muss gewährleistet werden können, dass keinerlei Daten mit Bezug zu Personen wie Patienten oder medizinischem Personal in den Modellen verwendet werden.

Mit Transfer-Lernen kann dem Problem einer für das Training von Klassifikatoren zu kleinen Datenmenge begegnet werden. Beim klassischen Ansatz des Transfer-Lernens im Kontext von ML werden jedoch Modelle aus einer Quellumgebung in eine Zielumgebung übertragen, um sie dort ggf. nach weiterem Training auf eine neue Problemstellung anzuwenden (Abb. 10.1).

Dieses Vorgehen kann aus den oben genannten Gründen in sensiblen Bereichen nicht angewendet werden.

Abb. 10.1 Transfer Learning
(nach Yang et al. 2020)

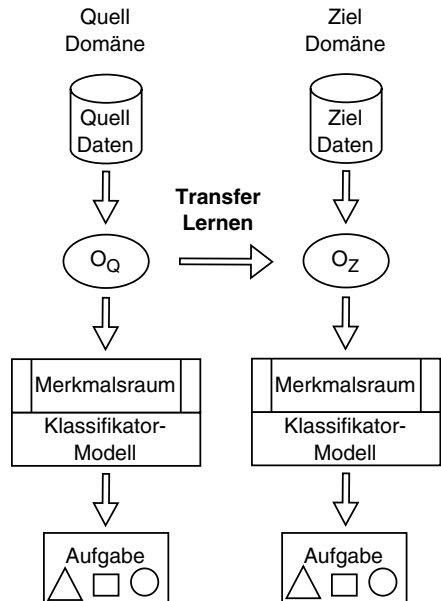


Ein hybrider Ansatz, bei dem ein kontrolliertes Vokabular genutzt wird, kann aber eine Wiederverwendung der trainierten Modelle ermöglichen. Extraktionsprozesse nutzen hier ausschließlich die in einer Ontologie definierten Begriffe und stellen damit vereinheitlichte Informationen für die Klassifikatoren zur Verfügung. Diese Klassifikatoren sind nur sensitiv für dieses Vokabular und können damit ohne Bedenken bzgl. der Datenschutzgrundverordnung (DSGVO) in anderen Umgebungen eingesetzt werden.

Dieses *Transfer by Copy* deckt aber vermutlich nur einen geringen Teil der Begriffe aus den Zieldaten ab. Es kann nun durch einen zusätzlichen Prozess zu *Transfer by Enrichment* erweitert werden: Für Domänen mit ähnlichen, aber abweichenden Begriffen wird die Basisontologie aus der Ausgangsdomäne (Quelldomäne) mit den Begriffen der neuen Zieldomäne angereichert. Bei dieser Konzeptnormalisierung werden Begriffe der Zieldomäne wieder auf die Terminologie der Ausgangsdomäne zurückgeführt. Der Extraktionsprozess stellt den Klassifikatoren damit wieder die Informationen auf Basis der bekannten und beim Training verwendeten Begriffe bereit und ist sensitiv für die möglicherweise abweichende Terminologie der Zieldomäne.

Durch den hybriden, ontologiebasierten Ansatz des Transfer-Lernens können trainierte Modelle also einfach in neue Umgebungen übernommen werden (Abb. 10.2). Anders als Ansätze des traditionellen Transfer-Lernens wird hier kein vortrainiertes Modell, wie etwa ein Klassifikator, von der Quell- in die Zieldomäne übertragen, um dann mit ggf. wenigen Daten aus der Zieldomäne in einem Feintuning weiter trainiert zu werden. Beim ontologiebasierten Transfer by Enrichment wird nicht der Klassifikator selbst an

Abb. 10.2 Prozessmodell des ontologiebasierten Transfer-Lernens



die Zieldomäne angepasst, sondern die Abbildung der Daten auf den Merkmalsraum. Der wohldefinierte Merkmalsraum erleichtert die Nachvollziehbarkeit der Klassifikation und liefert einen Schritt in Richtung der Erklärbarkeit der Modelle, da die von Klassifikatoren erkannten Muster – Kombinationen von Begriffen – über die Zusammenhänge der Ontologie dargestellt und mit den Erwartungen der Benutzer in Beziehung gesetzt werden können (Guckert et al. 2021). Dieses semantische Transfer-Lernen nutzt Ontologien als Brücke zwischen den Domänen. Durch die Vereinheitlichung der relevanten Merkmale können Klassifikatoren in beiden Domänen ohne zusätzliches Training angewendet werden.

In den folgenden Kapiteln wird ein möglicher Anwendungsfall aus dem Gesundheitswesen vorgestellt, der beide Herausforderungen, den Umgang mit einem hohen Datenschutzniveau und wenig Daten für ein Training, adressiert. Im Abschnitt Technologien werden die zugrunde liegenden Ideen für die Lösung erklärt, um dann in den Abschnitten Softwarearchitektur und Implementierung und Evaluierung den Aufbau des Frameworks *ontoTL* (ontologiebasiertes Transfer-Lernen) genauer zu erläutern. Ebenso wird eine Möglichkeit der Umsetzung gezeigt. Abschließend werden die Ergebnisse des Frameworks eingeordnet und ein Ausblick auf zukünftige Arbeiten vorgestellt.

Abb. 10.3 zeigt das Boxology-Modell¹ des in diesem Kapitel vorgestellten hybriden KI-Systems, bestehend aus zwei Bereichen. Zunächst werden während der Konzeptextraktion Texte aus elektronischen Patientenakten sowie Word-Embeddings als modellfreie Daten verwendet. Diese fließen mit symbolischen Daten der MeSH-Ontologie in einem Machine-Learning-Verfahren (NCR) zusammen. Das Resultat dieses Verfahrens

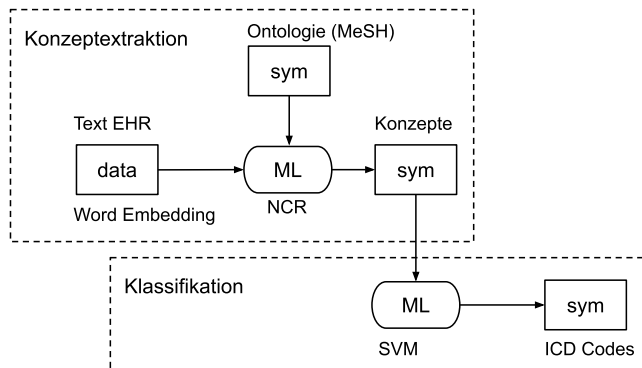
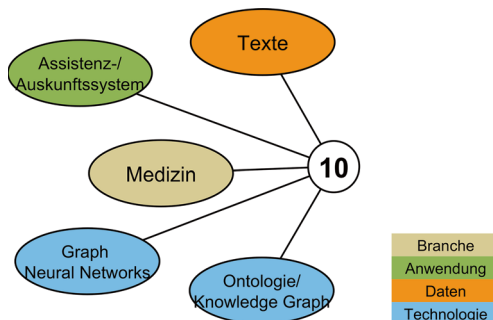


Abb. 10.3 Boxology-Modell des hybriden KI-Systems

¹Für eine Einführung in die Boxology siehe Kap. 1 sowie (Van Harmelen und Teije 2019).

Abb. 10.4 Thematische Einordnung des Kap. 10



sind ontologische Konzepte, welche symbolischer Natur sind. Diese Konzepte werden im zweiten Teil, der Klassifikation, weiterverwendet. Hier dienen sie einer ML-Methode (SVM) als Merkmalsraum. Das Ergebnis der Klassifikationsaufgabe sind ICD-Codes, welche durch ihre taxonomische Struktur ebenfalls als symbolische Daten interpretiert werden.

Abb. 10.4 zeigt die thematische Einordnung der Arbeit in die Kategorien Branche, Anwendung, Daten und Technologien. Die entwickelte Anwendung ist ein Assistenz- und Auskunftssystem, welches im medizinischen Kontext eingesetzt wird. Als Eingabe dienen unstrukturierte Textdaten. Zu den eingesetzten Methoden gehören Graph Neural Networks und Ontologien.

10.2 Anwendungsfall

Aufgrund des zunehmenden Kostendrucks im Gesundheitswesen wird gerade in der Verwaltung von Krankenhäusern zunehmend auf Digitalisierung gesetzt. Der vorliegende Anwendungsfall behandelt die automatisierte Beschriftung von Patientenakten mit Codenummern aus den gängigen medizinischen Klassifikationssystemen.

Im Rahmen ärztlicher Behandlungen sind Krankenhäuser verpflichtet, für jeden Patienten alle medizinisch relevanten Informationen in Patientenakten zu dokumentieren. Dies betrifft Diagnosen, chirurgische Eingriffe und andere Behandlungen. Für die Abrechnung der Leistungen mit den Krankenkassen müssen diese Informationen in standardisierter Form vorliegen, um eine Vergleichbarkeit zu gewährleisten. Für diese Standardisierung sorgen die in Deutschland verwendeten Kodiersysteme ICD (*International Classification of Diseases*) und OPS (*Operationen- und Prozedurenschlüssel*). Die Beschriftung der Dokumentationen mit Codenummern aus den standardisierten Klassifikationssystemen ist mit einem enormen manuellen Arbeitsaufwand verbunden: Krankenhäuser müssen teures medizinisches Fachpersonal wie Ärzte und Medizincontroller für diese Tätigkeit abstellen. Der Mangel an entsprechend qualifizierten

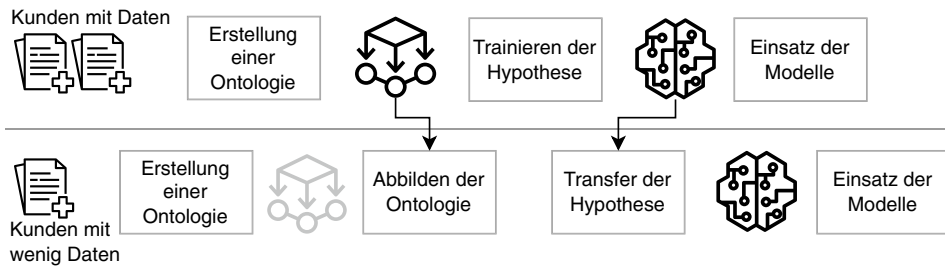


Abb. 10.5 Transfer-Lernen durch Ontologien im Gesundheitswesen

Arbeitskräften führt dabei entweder zu hohen Personalkosten oder im Falle unbesetzter Stellen zu Verzögerungen in der Rechnungstellung und fehlerhaften Abrechnungen.

Das Ziel innerhalb des Anwendungsfalls besteht darin, den manuellen Arbeitsaufwand bei der Beschriftung von Patientenakten mit medizinischen Codenummern drastisch zu reduzieren. Ausgangspunkt für die neue Lösung ist die Software *mm-care* von Synpulse. Die Software setzt bereits *Natural Language Processing* und *maschinelles Lernen* zur Klassifikation in der medizinischen Dokumentation ein: Nach einer initialen Lernphase auf der Grundlage historischer Daten eines Krankenhauses konnte rund die Hälfte aller Patientenfälle des betreffenden Krankenhauses automatisch kodiert werden.

Leider konnten die Ergebnisse der Lernphase eines Krankenhauses nicht auf andere Krankenhäuser übertragen werden. Gründe sind die jeweils individuelle Sprache der Krankenhäuser und der Schutz von Patientendaten nach DSGVO. Bei Einführung der Software wurde daher für jedes Krankenhaus eine neue initiale Lernphase durchgeführt, die eine ausreichend große Basis historischer Daten voraussetzte. Krankenhäuser ohne eine entsprechende Datenbasis konnten von der bestehenden Softwarelösung nicht profitieren.

Der Lösungsansatz war es, wie in Abb. 10.5 zu sehen, vorhandene Ontologien des Gesundheitswesens so anzureichern, dass ein Transfer-Lernen möglich wird: Die neue Lösung sollte sich automatisch an den fachlichen Sprachgebrauch eines neuen Krankenhauses anpassen und gleichzeitig eine Anonymisierung der Patientendaten ermöglichen. Die Klassifikation findet dabei nicht mehr auf den reinen Textdaten des Krankenhauses statt, sondern auf der Basis der erstellten Ontologie. Dies sollte die Übertragbarkeit des erlernten Wissens unter Berücksichtigung des Schutzes von Patientendaten ermöglichen.

10.3 Technologien

Das Prozessmodell des ontologiebasierten Transfer-Lernens (Abb. 10.2 Bruneß et al. (2022)) stellt eine Weiterentwicklung des traditionellen Transfer-Lernens (Yang et al. 2020) dar. In der Literatur werden erste Arbeiten des ontologiebasierten Transfer-Lernens beschrieben (Fumagalli et al. 2020; Herold et al. 2019; Minor et al. 2021).

Diese Technologie steckt noch in den Kinderschuhen, hat aber großes Potenzial für Anwendungen. Im Folgenden werden die Technologien vorgestellt, die im Prozessmodell des ontologiebasierten Transfer-Lernens (Abb. 10.2) zum Einsatz kommen.

10.3.1 Standardmethoden zur Textklassifikation

Textklassifikatoren, die auf Dokumenten eines medizinischen Anwendungsgebietes (Quelldomäne, D_Q) trainiert wurden und dort gut funktionieren, sollen auf Dokumenten eines anderen Anwendungsgebietes (Zieldomäne, D_Z) wiederverwendet werden. Ein Klassifikator, der beispielsweise auf das Erkennen einer „Gehirnerschütterung“ (ICD-Code S06.0) in den elektronischen Patientenakten eines Krankenhauses spezialisiert ist, soll diese auch in umgangssprachlichen Texten von Patient*innen in den sozialen Medien erkennen. Standardmethoden zur Textklassifikation wie Support-Vektor-Maschinen (SVM) oder K-nearest Neighbor (kNN) sind seit Jahrzehnten im Einsatz und technologisch sehr ausgereift. Sie gehören quasi zu den Klassikern des maschinellen Lernens. Ist eine ausreichend große Menge an Trainingsdaten verfügbar, liefern sie in der Regel sehr gute Klassifikationsergebnisse.

10.3.2 Ontologien

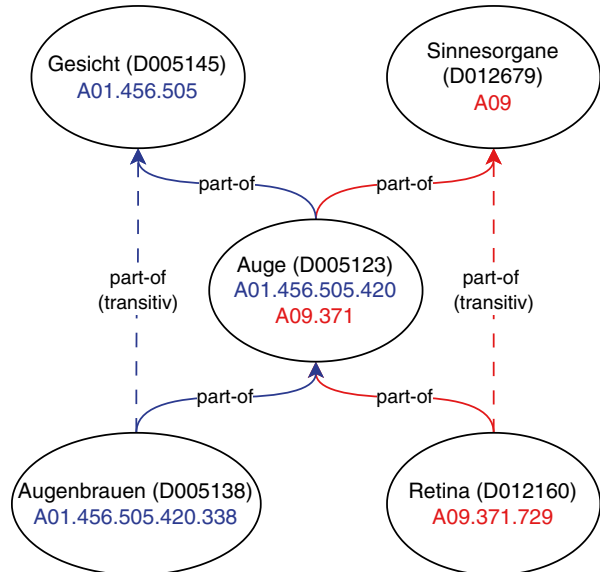
Der Begriff Ontologie ist aus der Philosophie entlehnt und bezeichnet dort die Disziplin, die das Seiende untersucht und eine systematische, umfassende Darstellung der gesamten Wirklichkeit anstrebt. Im Kontext wissensbasierter Systeme werden in einer Ontologie die Entitäten einer Domäne und deren Beziehungen als Konzepte explizit und formal beschrieben. Die Menge der in einer Ontologie repräsentierten Objekte definiert Vokabular und Struktur für die Nutzung durch wissensbasierte Systeme. Eine Ontologie ist eine explizite, formale Spezifikation der Konzeptualisierung der relevanten Objekte eines Diskursuniversums (Gruber 1993).

Die Medical Subject Headings (MeSH)² ist eine Ontologie medizinischen Fachvokabulars. Die deutsche Übersetzung wird seit 2020 von der Deutschen Zentralbibliothek für Medizin (ZB MED) veröffentlicht und bereitgestellt. Diese beinhaltet über 28.000 Hauptschlagwörter (Main Headings), mehr als 208.000 englische Entry Terms (Synonyme) und über 64.000 deutsche Entry Terms (Synonyme).

Die Hauptschlagwörter stehen miteinander in Beziehung. So lassen sich aus der definierten Baumstruktur Ober- und Unterbegriffe sowie Synonyme ableiten. Neben diesen Relationen sind auch Verweise zu anderen Einträgen als Metadaten vorhanden. Der MeSH-Baum besteht aus mehreren Einzelbäumen, die jeweils eine medizinische Kate-

²<https://www.nlm.nih.gov/mesh>

Abb. 10.6 Veranschaulichung der Einordnung des MeSH-Konzepts „Auge“ in mehreren Teilbäumen



gorie abbilden, z. B. Anatomie (A), Krankheiten (C) oder Disziplinen und Berufe (H). Dabei ist zu beachten, dass ein Konzept in mehreren Teilbäumen verortet werden kann (Abb. 10.6).

Im betrachteten Anwendungsfall wird die MeSH als Basisontologie verwendet.

Ontologien sind seit den 2000er-Jahren eine etablierte Technologie für wissensbasierte Systeme. In Zusammenhang mit modernen Methoden des Maschinellen Lernens und Deep Learning erleben sie derzeit eine Renaissance.

10.3.3 Ontologiereicherung

Ein naives Kopieren von Klassifikatoren aus einer Quell- in eine Zieldomäne führt zu schlechten Ergebnissen. Selbst wenn die Klassifikatoren auf dem einheitlichen Vokabular der Ontologie trainiert werden und die Terme der Ontologie den Merkmalsraum (Input Feature Space) bilden, ist zu erwarten, dass deren Ergebnisqualität in der Zieldomäne niedriger ist als in der Quelldomäne, obwohl die Nutzung der Ontologie die Ergebnisse verbessert. Bei diesem Transfer by Copy reagieren die Klassifikatoren nicht auf Terme aus der Zieldomäne, die in der Ontologie nicht gefunden werden. Durch Transfer by Enrichment werden diese Begriffe auf das Vokabular der Ontologie abgebildet und die Klassifikatoren sind damit sensitiv.

Dazu wird der aus den Konzepten der Ontologie gebildete Merkmalsraum der Klassifikatoren angereichert (Abb. 10.2). Die Ausprägungen der Konzepte (Literals) kommen als Terme in den Dokumenten vor. Die Ontologie der Quelldomäne O_Q beinhaltet in die-

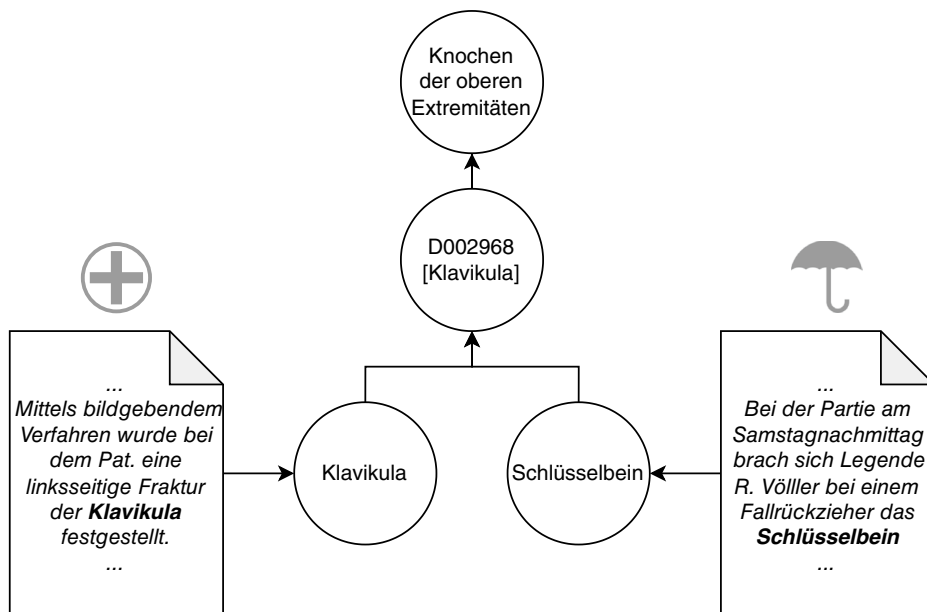


Abb. 10.7 Ausschnitt der Ontologie zum medizinischen Fachbegriff „Klavikula“

sem Anwendungsfall vor allem medizinische Fachbegriffe als Ausprägungen. Die Ontologie der Zieldomäne O_Z ist um umgangssprachliche Terme angereichert. Die ontologiebasierte Abbildung eines Dokuments auf den Merkmalsraum des Klassifikators kann in die Forschungsansätze der Concept Normalisation (Xu et al. 2020; Tutubalina et al. 2018; Niu et al. 2019) eingeordnet werden. Abb. 10.7 illustriert dies am Beispiel eines Schlüsselbeinbruchs.

In einer Patientenakte in der Quelldomäne wird eine „Fraktur der Klavikula“ beschrieben. Der umgangssprachliche Text in der Zieldomäne formuliert einen ähnlichen Sachverhalt wie „brach ... das Schlüsselbein“. Die Terme „Klavikula“ und „Schlüsselbein“ werden zum Konzept *D002968* (Klavikula) in der Ontologie MeSH normalisiert.

Wie in Abb. 10.8 gezeigt, verwendet der hier vorgestellte Ansatz lediglich die Informationen zu Ober- und Unterbegriffen sowie die Synonymrelationen. Letztere werden aus gemeinsamen Elternkonzepten zweier Terme abgeleitet. „Klavikula“ und „Schlüsselbein“ haben ja das gemeinsame Konzept *Klavikula*, woraus sich ihre Synonymität ableiten lässt.

Um bessere Konzepterkennungsraten in anderen Sprachdomänen zu erzielen, müssen die Termlisten der Konzepte um domänenspezifische Synonyme erweitert werden. Dazu wurden verschiedene Quellen herangezogen:

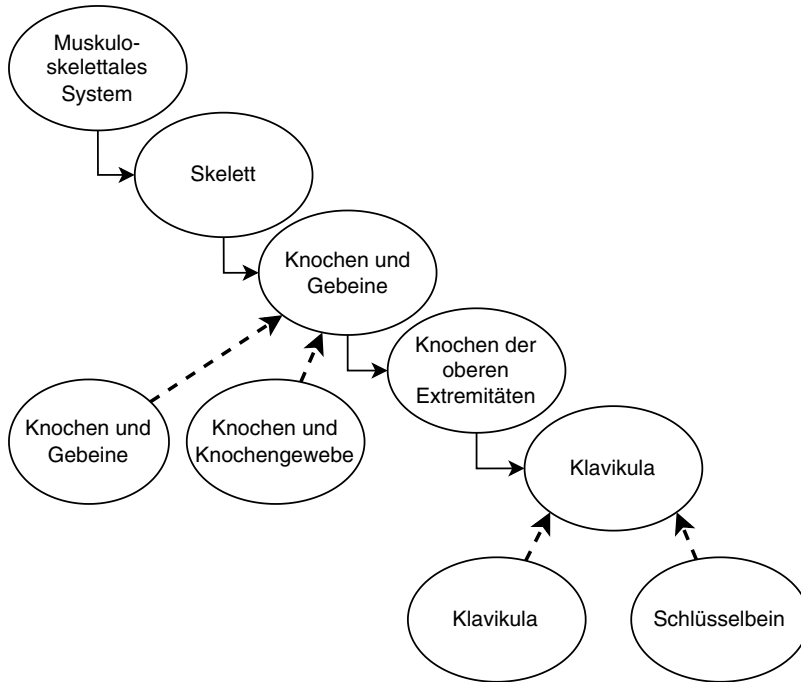


Abb. 10.8 Exemplarischer Ausschnitt aus MeSH mit Termen an Konzepten (gestrichelte Pfeile)

- WikiData Label,³
- OpenThesaurus,⁴
- Unified Medical Language System (UMLS).⁵

Die Datentabellen von WikiData und UMLS verfügen über direkte Verweise zu MeSH-Einträgen, wodurch unbekannte Terme direkt über den Identifier an ein Konzept gebunden werden können. Diese beiden Datenquellen werden primär für das Erweitern von fachsprachlichen Termen verwendet. Die Einträge im OpenThesaurus sollen vor allem die medizinische Laiensprache in der Ontologie abdecken und verfügen über keine bekannten Identifier. Hier wurde mit einfachen Stringvergleichen auf den lemmatisierten Einträgen aus MeSH und OpenThesaurus gearbeitet.

Da die Terme aus den oben genannten Quellen in der Ontologie persistiert werden, wird dieser Prozess als *Eager Enrichment* bezeichnet. Die Daten aus dem Eager Enrichment werden anschließend für das Training des Konzepterkennungsmodells verwendet.

³ [Wikidata.org/wiki](https://wikidata.org/wiki/)

⁴ [Openthesaurus.de](https://openthesaurus.de)

⁵ [Nlm.nih.gov/research/umls](https://nlm.nih.gov/research/umls)

Dieses Modell ist dann in der Lage, auch Konzepte im Text zu identifizieren, deren Schreibweise so nicht in der Ontologie persistiert ist, z. B. verschiedene Flexionen. Dieses Mapping-Wissen wird im Modell gehalten und nicht in der Ontologie persistiert. Aus diesem Grund wird das Vorgehen als *Lazy Enrichment* bezeichnet.

Im vorliegenden Anwendungsfall wird ein Modell von Arbabi et al. (2019) als Konzepterkennungsmodell verwendet. Die automatisierte Anreicherung von Ontologien (Ontology Enrichment) ist ein aktives Forschungsgebiet (Paulheim 2016).

10.3.4 Konzepterkennung

Für die Erkennung von Konzepten in unstrukturiertem Text wird die *Neural-Concept-Recognition-(NCR)-* Methode von Arbabi et al. (2019) eingesetzt. Dieses Modell wird auf einer hierarchischen Datenstruktur, zum Beispiel einer Taxonomie, trainiert und ordnet N-Grammen Konzepte mit einer bestimmten Wahrscheinlichkeit zu. Beispielsweise könnte NCR die Eingabesequenz „Schlüsselbeinbruch“ mit einer 98 %igen Wahrscheinlichkeit auf das Konzept *Klavikula* abbilden. Beim Training des Modells wird die hierarchische Struktur der Eingabetaxonomie berücksichtigt. Dies geschieht, indem jedes Konzept durch sich selbst und durch seine Vorfahren in einem Konzept-Embedding repräsentiert wird. Somit kann die Semantik von Termen genauer differenziert werden. Das Training des Modells findet lediglich auf der Ontologie/Taxonomie statt, es wird also kein annotiertes Korpus benötigt.

Für die Einbettung der Textdaten wird ein deutsches *fastText*-Word-Embedding-Modell verwendet. Dieses basiert auf Teilworteinbettungen und umgeht somit Out-of-Vocabulary-(OOV)-Probleme, welche in speziellen Sprachdomänen durchaus häufiger auftreten können.

Das Modell besteht aus einem Encoder und einem Concept Embedding. Der Encoder berechnet die Repräsentation der Eingabesequenz. Dazu werden die einzelnen Token mithilfe des *fastText*-Modells eingebettet und über ein 1D Convolution Layer mit einer ELU-Aktivierungsfunktion und einem max-Pooling gemittelt. Es folgen ein Fully Connected Layer und eine ReLU-Aktivierungsfunktion mit L2-Normalisierung. Das Concept Embedding wird durch eine Matrix dargestellt, wobei jede Zeile einem Konzept entspricht.

Die endgültige Einbettung eines Konzepts ist die Einbettung seines Elternteils (oder des Durchschnitts seiner Eltern im Falle der Mehrfachvererbung) plus seine eigene Roh-einbettung. Mit anderen Worten, das Elternkonzept liefert die globale Position im Einbettungsraum, während die Unterkonzepte ihre lokalen Positionen in Bezug auf diesen Raum lernen.

Das Concept Embedding lernt die Repräsentationen für Konzepte. Diese Konzept-einbettung wird verwendet, um die Ähnlichkeit zwischen einer Eingabephase und den Konzepten zu messen. Dazu wird das Punktprodukt zwischen diesen Repräsentationen und der kodierten Phrase berechnet.

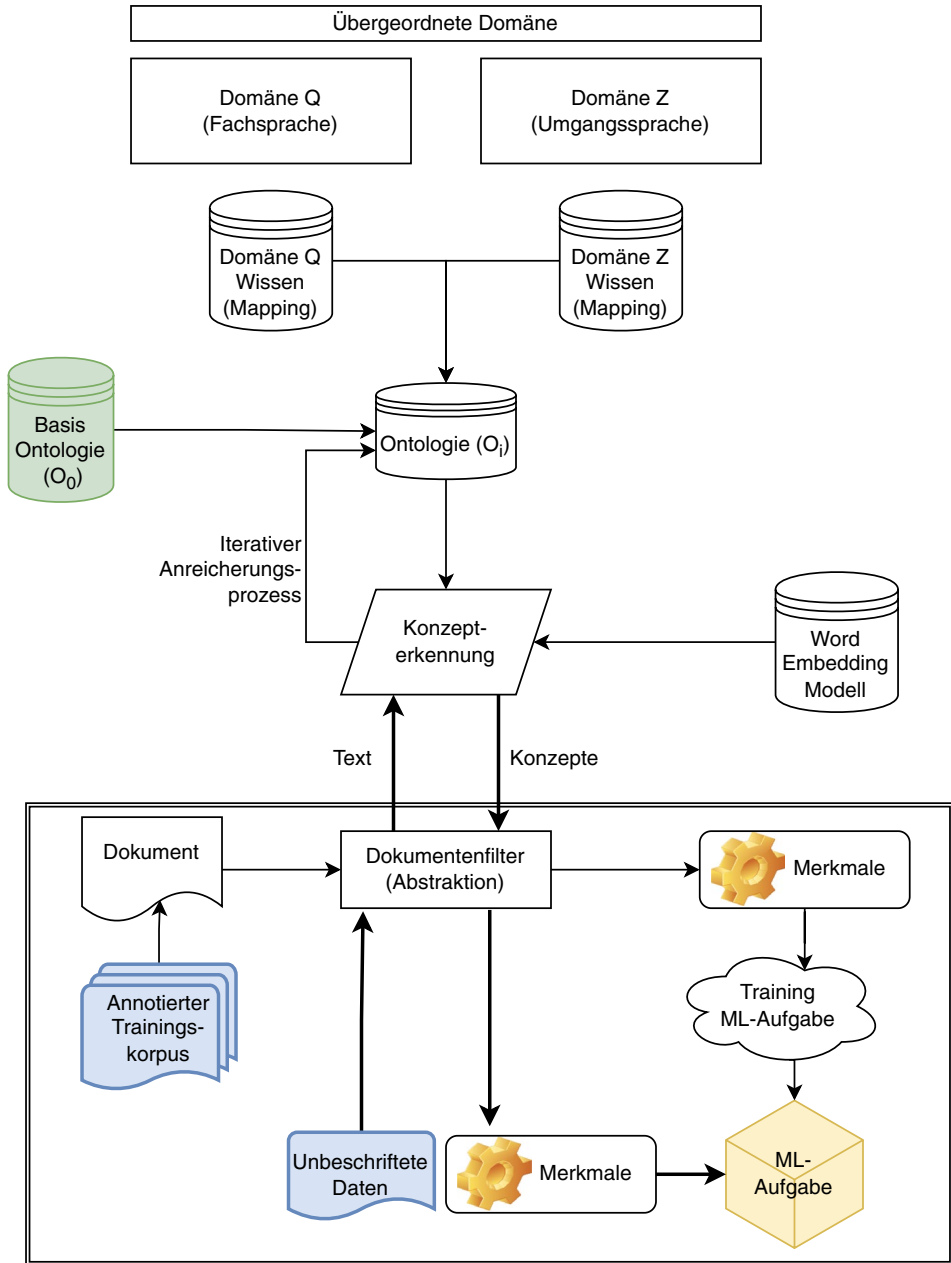


Abb. 10.9 Softwarearchitektur

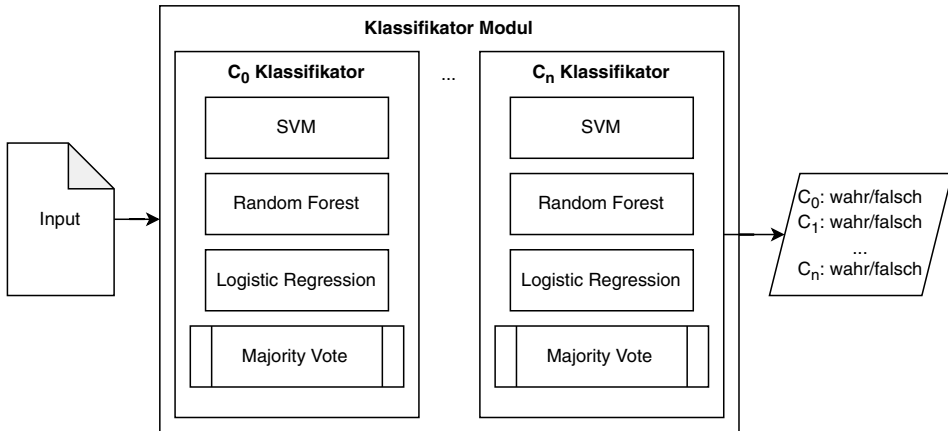


Abb. 10.10 Aufbau des Klassifikatormoduls

Die NCR-Methode ist relativ jung, wurde aber schon erfolgreich für Anwendungen in der Bioinformatik eingesetzt. Hier wird dieses Verfahren erstmals im Kontext von Transfer-Lernen erprobt und hat gute Ergebnisse erzielt.

10.4 Softwarearchitektur

Die ontoTL-Softwarearchitektur besteht im Wesentlichen aus den in Abb. 10.9 dargestellten Modulen. Dies ist zum einen der Klassifikator (ML-Aufgabe, unten rechts im Bild), der die eigentliche Aufgabe der Problemstellung löst. Des Weiteren hält das GraphDB-Modul die Ontologie (B_1 , in der Mitte) und ist die Schnittstelle für Abfragen. Das Abstraktionsmodul (NCR) dient zur Substitution von Textfragmenten zu Konzeptidentifikatoren der Ontologie (Dokumentenfilter).

Das Klassifikatormodul nimmt eine sogenannte One-vs-all-(OvA-)Klassifikation vor. Dies bedeutet, dass für jede Klasse ein binärer Klassifikator zur Verfügung steht (N Klassen, N binäre Klassifikatoren). Jeder dieser Klassifikatoren hat zur Aufgabe, eine Klasse von allen anderen Klassen zu unterscheiden.

Es hat sich herausgestellt, dass ein Ensemble aus mehreren einzelnen Klassifikationsalgorithmen pro binärem Klassifikator von Vorteil ist. So werden hier drei Algorithmen des maschinellen Lernens für jede Klasse eingesetzt und per Mehrheitsregel abgestimmt (Abb. 10.10). Konkret bedeutet dies den Einsatz einer Support Vector Machine, eines Random Forest und eines Logistic-Regression-Algorithmus. Durch ein sogenanntes Soft Voting wird das Klassenlabel bestimmt. Dieses Verfahren summiert die vorhergesagten Wahrscheinlichkeiten aller Klassifikatoren und bestimmt den argmax („argumentum maximi“, Argument des Maximums), um die letztendlich vorhergesagte Klasse zu ermitteln.

Bei der Verwendung eines Soft Voting kann auch eine Gewichtung der einzelnen Klassifikationsalgorithmen vorgenommen und so eine weitere Ebene für das Fine-Tuning der Hyperparameter geschaffen werden. Da es sich bei der Problemstellung um ein Multi-klassen- und Multilabelszenario handelt, wird das Abstimmungsverfahren für jedes Klassifikatorenensemble separat angewendet.

Ein weiteres Modul stellt die Ontologie für Lese- und Schreiboperationen bereit. Das Ontologiemodul besteht im Kern aus einer Ontotext-GraphDB⁶-Instanz, welche die Ontologie im RDF-Format hält. GraphDB ermöglicht es, mithilfe der Abfragesprache SPARQL semantische Abfragen an die Ontologie zu richten. Dadurch sind komplexe Abfragen nach z. B. transitiven Relationen im Graphen möglich.

Die Schnittstelle zur Graphdatenbank wird mit der Python-Bibliothek SPARQLWrapper⁷ realisiert. Dank der modularen Architektur können hier aber ebenfalls andere Graphdatenbanken, wie beispielsweise Neo4j oder OrientDB, zum Einsatz kommen.

Für die Normalisierung der Konzepte findet ein drittes Modul Verwendung. Dieses Modul hebt medizinische Texte auf ein höheres Abstraktionsniveau bzw. stellt die Merkmale für das Klassifikatormodul bereit. Generell kann auch hier die Konzeptnormalisierung auf verschiedene Weise realisiert werden.

Wie bereits erwähnt wurde darauf geachtet, eine möglichst modulare Architektur zu implementieren, um sowohl in der Evaluationsphase als auch im späteren Produktiveinsatz schnell auf neue Erkenntnisse bzgl. der Performance einzelner Komponenten reagieren zu können.

Ebenfalls sollte im späteren Produktivsystem eine einfache Skalierung der Anwendung möglich sein. Aus diesem Grund wird eine Apache-Kafka⁸-Umgebung eingesetzt, über welche die einzelnen Module kommunizieren. Kafka ist eine Plattform zur Verarbeitung von Datenströmen und wird im Normalfall auf einem Rechencluster aufgesetzt, aber auch der Single-node-Einsatz ist möglich.

Die oben beschriebenen Module werden als Docker⁹ Images abgelegt und können dann beliebig häufig, in Abhängigkeit von der aktuellen Last oder den Ansprüchen der Anwender, gestartet werden. Die Kafka-Umgebung sorgt für eine gleichmäßige Verteilung der Last unter den einzelnen Modulen. Da lediglich Open-Source-Python-Bibliotheken wie u. a. scikit-learn,¹⁰ sparql-query¹¹ und fastText¹² eingesetzt werden, ist ein hohes Maß an Reproduzierbarkeit gegeben.

⁶<https://www.ontotext.com>

⁷<https://github.com/RDFLib/sparqlwrapper>

⁸<https://kafka.apache.org>

⁹<https://www.docker.com>

¹⁰<https://scikit-learn.org>

¹¹<https://www.w3.org/TR/rdf-sparql-query>

¹²<https://fasttext.cc>

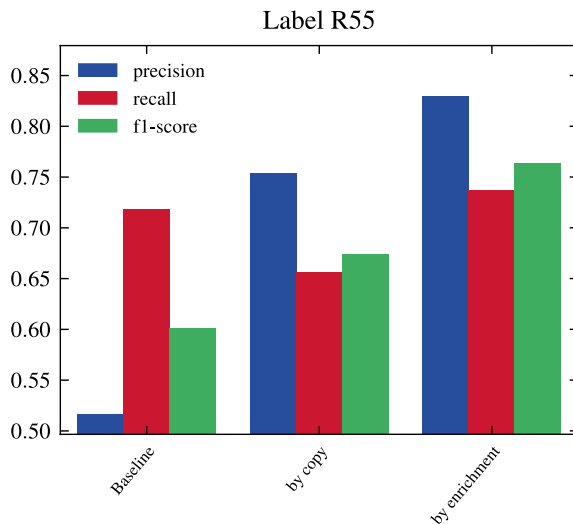


Abb. 10.11 Klassifikatorergebnisse in der Zieldomäne

10.5 Implementierung und Evaluierung

Der Klassifikator soll domänenübergreifend eingesetzt werden. Der Sprachduktus unterscheidet sich von Domäne zu Domäne, in Wortwahl, Grammatik und Komplexität. Vor allem die divergierende Wortwahl der Fachsprache gegenüber der Laiensprache macht den Einsatz von einfachen, frequenzbasierten Ansätzen zur Erstellung des Merkmalsraumes schwierig. Aus diesem Grund wird im ontoTL Framework der Merkmalsraum der Klassifikatoren aus der Menge der Ontologiekonzepte gebildet. Die Fachsprache wird mit der deutschsprachigen MeSH abgebildet. Dazu wurde MeSH aus dem XML- in das RDF-Format konvertiert und in der GraphDB persistiert.

Um den Spagat in die Domäne der Laiensprache zu bewerkstelligen, wird die Ontologie mit Termen aus dieser Domäne angereichert. Dies geschieht mithilfe von verschiedenen Mapping-Tabellen (UMLS, OpenThesaurus, WikiData). Die Inhalte der Mapping-Tabellen werden ebenfalls an den jeweils korrespondierenden Konzepten persistiert. Für den Fall, dass solche Mapping-Tabellen nicht zur Verfügung stehen, soll die Möglichkeit einer automatischen Taxonomieanreicherung geprüft werden.

Das Modell für die Konzeptnormalisierung wurde unter Verwendung des deutschen fastText Embeddings sowie unterschiedlicher Anreicherungsstadien der MeSH trainiert.

Die Methode wurde auf realen Daten aus Krankenhäusern (Patientenakten) sowie Forenbeiträgen aus Gesundheitsportalen evaluiert. Für das Training in der Quelldomäne standen rund 2500 Entlassungsbriefe in medizinischer Fachsprache zur Verfügung. Die einzelnen Dokumente lassen sich in drei Klassen unterteilen (S. 06, R55, kein ICD Code). Der ICD-Code S. 06 kodiert intrakranielle Verletzungen bzw. eine Gehirn-

erschütterung (*Commotio cerebri*). Hier lagen etwa 850 Trainingsdokumente vor. R55 kodiert *Synkope und Kollaps*, also einen Blackout/Ohnmacht. Hierzu lagen etwa 550 Dokumente im Trainingskorporus vor. Rund 1000 weitere Dokumente wurden ohne Kodierung als Negativproben verwendet.

Für die Evaluation in der Zieldomäne wurden 96 Foreneinträge von Experten mit den Labeln aus dem Trainingskorporus annotiert ($34 \times S. 06$, $37 \times R55$, $25 \times$ kein Label). Die Grundannahme war, dass ein Klassifikator **C** sehr gut in der Quelldomäne funktioniert. Diese Annahme konnte mithilfe einer Kreuzvalidierung bestätigt werden. Transferiert man nun **C** in die Zieldomäne, ohne weitere Modifikationen vorzunehmen, so lautet die Annahme, dass **C** ungenügende Ergebnisse produziert.

Die Haupthypothese war, dass die Anreicherung der Ontologie mit Wissen aus der Zieldomäne die Klassifikation verbessert. Die Ergebnisse der Experimente zeigten zum einen, dass die Abstraktion der Texte/Merkmale auf Ontologieebene gegenüber der Baseline (TF/IDF auf Wortebene) bereits eine enorme Verbesserung der Ergebnisse mit sich bringt (Abb. 10.11). Zum anderen verbessert das Anreichern der Ontologie um Wissen der Zieldomäne die Klassifikation noch weiter.

Die Anwendung wurde prototypisch für eine kleine Teilmenge der ICD-Codes getestet und zeigte Erfolg versprechende Ergebnisse. Durch die frühzeitige Entscheidung, das System modular und stark skalierbar aufzubauen, ist das Ausrollen auf weitere Codes/Klassen/Label jederzeit möglich. Auch die Adaption auf andere oder weitere Subdomänen lässt sich durch Anreicherung der Ontologie jederzeit realisieren. Weitere Details sind in Bruneß et al. (2022) zu finden.

10.6 Empfehlungen

Aus dem vorgestellten Anwendungsfall der automatisierten Annotation von Patientenakten durch ontologiebasiertes Transfer-Lernen können nachfolgende Empfehlungen abgeleitet werden:

1. Es muss nicht zwangsläufig Deep Learning sein. Traditionelle ML-Methoden zeigen teilweise hervorragende Ergebnisse, vor allem bei geringen Trainingsdaten.
2. Bei ontologiebasiertem Transfer-Lernen sollte die Basisontologie wohl überlegt gewählt werden. Sie ist die Grundlage für die darauffolgenden Schritte, wie beispielsweise Anreicherung, Training und Inferenz.
3. Eine ausgiebige Recherche von State-of-the-Art-Methoden und ML-Tools kann den Arbeitsaufwand reduzieren. Das Rad muss nicht neu erfunden werden, da viele gute State-of-the-Art-Methoden – oft Open Source – online zur Verfügung stehen.
4. Qualitativ hochwertige Daten (Textdaten, Ontologie, Anreicherungsdaten) sind der Schlüssel für gute Ergebnisse.
5. Compliance und Datenschutz spielen im Kontext sensibler Anwendungen eine immer größere Rolle.

10.7 Fazit

Das ontoTL Framework erlaubt es, auf einem abstrahierten Merkmalsraum trainierte Klassifikatoren in neue Domänen zu transferieren. Grundlage der Extraktion ist eine Ontologie, die das kontrollierte Vokabular bereitstellt. Die Extraktion der relevanten Merkmale in der neuen Domäne kann durch einen Anreicherungsprozess verbessert werden. Somit implementiert ontoTL ein *Transfer-by-Enrichment*-Paradigma. In Experimenten lässt sich der Effekt des Enrichment durch Evaluation der Qualität der Klassifikatoren nachweisen: Während sich die Genauigkeit eines kopierten Klassifikators unter Nutzung einer nicht angereicherten Ontologie in der Zieldomäne verschlechtert, sind die Kennzahlen bei Nutzung einer angereicherten Ontologie vergleichbar. Es sei an dieser Stelle angemerkt, dass ein Kopieren von Klassifikatoren, die ohne abstrahierten Merkmalsraum direkt auf den Begriffen im Text arbeiten, – wie zu erwarten – nur sehr schlecht funktioniert.

Die Ontologiereicherung hat somit einen positiven Effekt und erzielt praxistaugliche Klassifikationsergebnisse. Darüber hinaus genügt diese transparente Form des Transfer-Lernens hohen Anforderungen an Compliance und implementiert Datenschutz by Design, da die Klassifikatoren ausschließlich auf dem kontrollierten Merkmalsraum arbeiten und somit keine vertraulichen Informationen aus Dokumenten benötigt, extrahiert oder gespeichert werden. Transfer-Lernen ist einfacher zu implementieren als häufig befürchtet.

Zukünftige Arbeiten werden eine weitere Justierung bezüglich der Wahl der Parameter (Klassifikatoren, Vektorisierer, Filter) umfassen sowie eine automatische Ontologiereicherung für Zieldomänen ohne die oben beschriebenen Mapping-Tabellen untersuchen. Außerdem muss für Zieldomänen ohne eigene Ontologie oder eigenen Knowledge Graph eine Möglichkeit geschaffen werden, weiteres Wissen in der Hauptontologie anzureichern.

Die bisherigen Ergebnisse können in zukünftigen Arbeiten auch für die bessere Erklärbarkeit von Klassifikationsergebnissen herangezogen werden. Rudin (2019) fordert interpretierbare Modelle. Sie macht selbst einen Lösungsvorschlag für Aufgaben der Bildklassifikation, der mit prototypischen Bildern arbeitet, um ein Klassifikationsergebnis für menschliche Augen plausibel zu machen. Analog dazu könnten in Zukunft auch für Textklassifikationsaufgaben prototypische Begriffe dargestellt werden, die einen hohen Einfluss auf das Klassifikationsergebnis haben. Denkbar sind im Kontext medizinischer Anwendungen auch prototypische Symptome oder ganze Krankheitsbilder. Da im vorliegenden Transfer-Lernen-Ansatz der Merkmalsraum der Klassifikatoren durch den Einsatz einer Ontologie normalisiert ist, liefert er eine gute Grundlage für die Erklärbarkeit der Klassifikationsergebnisse.

Literatur

- Arbabi, A., Adams, D.R., Fidler, S., & Brudno, M. (2019). Identifying Clinical Terms in Medical Text Using Ontology-Guided Machine Learning. *JMIR Medical Informatics*, 7.
- Bruneß, D., Bay, M., Schulze, C., Guckert, M., Minor, M. (2022). An Ontology-based transfer learning method improving classification of medical documents. *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, 407–412.
- Fumagalli, M., Bella, G., Conti, S., & Giunchiglia, F. (2020). Ontology-Driven Cross-Domain Transfer Learning. *Formal Ontology in Information Systems*.
- Gruber, T. R. (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2), 199–220.
- Guckert, M., Gumpfer, N., Hannig, J., Keller, T., & Urquhart, N. (2021). A conceptual framework for establishing trust in real world intelligent systems. *Cognitive Systems Research*, 68, 143–155.
- Herold, M., & Minor, M. (2019). Ontology-based transfer learning in the airport and warehouse logistics domains. *ICCBR Workshops*.
- Minor, M., Herold, M., Rubbe, J., Dufner, S. & Brussas, G. (2021). Transfer Learning Operators for Process-oriented Cases. In *Proc. 2021 IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering AIKE 2021*, 9–16, 2021. IEEE Computer Society Press
- Niu, J., Yang, Y., Zhang, S., Sun, Z., & Zhang, W. (2019). Multi-task Character-Level Attentional Networks for Medical Concept Normalization. *Neural Processing Letters*. 49(3), 1239–1256.
- Paulheim, H. (2016). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 8, 489–508.
- Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5), 206–215.
- Song, L., & Mittal, P. (2021, August). Systematic Evaluation of Privacy Risks of Machine Learning Models. In *USENIX Security Symposium (Vol. 1, No. 2, p. 4)*.
- Tutubalina, E., Miftahutdinov, Z., Nikolenko, S.I., & Malykh, V. (2018). Medical concept normalization in social media posts with recurrent neural networks. *Journal of biomedical informatics*, 84, 93–102.
- Van Harmelen, F., & Teije, A. T. (2019). A boxology of design patterns for hybrid learning and reasoning systems. arXiv preprint [arXiv:1905.12389](https://arxiv.org/abs/1905.12389).
- Xu, D., Zhang, Z., & Bethard, S. (2020). A Generate-and-Rank Framework with Semantic Type Regularization for Biomedical Concept Normalization. *Annual Meeting of the Association for Computational Linguistics*.
- Yang, Q., Zhang, Y., Dai, W. and Pan, S. J. (2020) “Transfer Learning in Reinforcement Learning,” in *Transfer Learning*. Cambridge: Cambridge University Press, pp. 105–125. <https://doi.org/10.1017/9781139061773.010>.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Anomalie-Detektion in der verarbeitenden Industrie

11

Christian Götz  und Bernhard G. Humm 

Kernaussagen

1. Anomaliedetektion ist essenziell für die Realisierung von modernen und sicheren cyber-Physischen Produktionssystemen (CPPS).
2. Für einen praxisorientierten Einsatz von Machine Learning in der verarbeitenden Industrie müssen besondere Anforderungen der dort verwendeten Systeme berücksichtigt werden.
3. Um die Interpretierbarkeit und Transparenz der Ergebnisse zu gewährleisten, muss eine Anomaliedetektion Interaktionen und Beziehungen innerhalb der Anlage und Aspekte, wie die zu produzierenden Produkte, verwendete Bauteile und globale Zustände, berücksichtigen.
4. Durch die Einbindung von externem Wissen in die Anomaliedetektion ist eine Verbesserung der Überwachung möglich.
5. Graphen bieten einen guten Ansatz für die Kombination von Prozessdaten mit externem Wissen und dadurch die Modellierung von Interaktionen und Relationen des CPPS.
6. Neuartige Techniken aus dem Machine Learning, wie beispielsweise Graph Neural Networks, ermöglichen eine effiziente und erfolgreiche Verarbeitung von Graphdaten auch für die Anomaliedetektion.

C. Götz (✉)

Yaskawa Europe GmbH, Hattersheim am Main, Deutschland

E-Mail: Christian.Goetz@yaskawa.eu

B. G. Humm

Hochschule Darmstadt – University of Applied Sciences, Darmstadt, Deutschland

E-Mail: bernhard.humm@h-da.de

© Der/die Autor(en) 2025

K. Hinkelmann et al. (Hrsg.), *Hybride KI mit Machine Learning und Knowledge Graphs*,

https://doi.org/10.1007/978-3-658-44781-6_11

243

11.1 Einleitung

Angetrieben durch den Aufschwung der künstlichen Intelligenz, die Veränderung der Industrie, bedingt durch die Digitalisierung und den Klimawandel, befindet sich auch die verarbeitende Industrie in einem starken Umbruch. Dabei spielen intelligente, sichere und vollautomatisierte Produktionssysteme eine entscheidende Rolle. Um diese zu ermöglichen, steigt der Einsatz von *Cyber-physischen Systemen* (CPS) innerhalb der Produktionssysteme. Dabei realisieren CPS komplexe Prozesse und ermöglichen gleichzeitig die Ausführung von fortschrittlichen Funktionalitäten, wie das Sammeln von Daten in Echtzeit, intelligente Datenanalysen oder die direkte Prozessoptimierung anhand von System-Feedback (Rajkumar et al. 2010; Jazdi 2014). Typischerweise besteht ein CPS dabei aus einer Kombination eines *Embedded*-Systems mit Sensoren und Aktoren, welche einen physischen Prozess überwachen und steuern (Marwedel 2021).

Moderne Produktionssysteme, bei welchen CPS eingesetzt werden, werden auch *Cyber-physische Produktionssysteme* (CPPS) genannt (Müller et al. 2021). CPPS zeichnen sich im Vergleich zu traditionellen Produktionssystemen durch eine intelligente Datenverwaltung, Echtzeitanalysen, Integration und Anwendung höherer Funktionen und eine erweiterte Rechenleistung aus (Monostori 2014). Des Weiteren finden die Kommunikationen innerhalb der CPS eines CPPS nicht wie in einer traditionellen Automatisierungshierarchie nur vertikal zwischen Geräten der unteren Ebene zu Geräten der höheren Ebene statt, sondern auch horizontal zwischen Geräten derselben Ebene. Damit stellen CPPS ein Schlüsselkonzept für die Realisierung moderner und sicherer Produktionssysteme und eines erfolgreichen Wandels in der verarbeitenden Industrie dar.

Solch ein CPPS, das *Rotary Table Dispenser System*, ist in Abb. 11.1 dargestellt. Das System wurde dabei basierend auf einem echten Industrieprozess konzipiert und dient als Demonstrationseinheit bei Yaskawa. Innerhalb dieses Systems interagieren mehrere CPS miteinander, um komplexe Prozesse, wie beispielsweise den Transport-, Aufnahme- und Ablageprozess, zu realisieren. Der gesamte Produktionsprozess stellt dabei einen Sortiervorgang von Objekten in Containern, den Produkten, dar, welche sich durch das System bewegen. Dabei beginnt dieser Prozess mit der Aufnahme der Objekte von einem zentralen Rotationstisch durch eine Platziereinrichtung. Anschließend platziert die Platziereinrichtung die Objekte in die Container, welche sich dabei auf einem Förderband bewegen. Erreicht ein Container das Ende des Förderbandes, wird dieser von einem zentralen Produktionsroboter aufgenommen und die Objekte werden zurück auf den Rotationstisch entleert. Der Container wird anschließend auf eine zentrale Transferposition platziert, um die Übergabe an einen zweiten Industrieroboter, den Logistikroboter, zu ermöglichen. Dieser stellt die Container im Anschluss von der Übergabeposition in eine Ablagevorrichtung auf einen Linearschlitten. Nach dem Befüllen der Ablage können die Container durch den Linearschlitten wieder an den Anfang des Förderbandes bewegt werden. Zuletzt werden die Container durch den Logistikroboter zurück auf das Förderband platziert und der Prozess startet erneut.

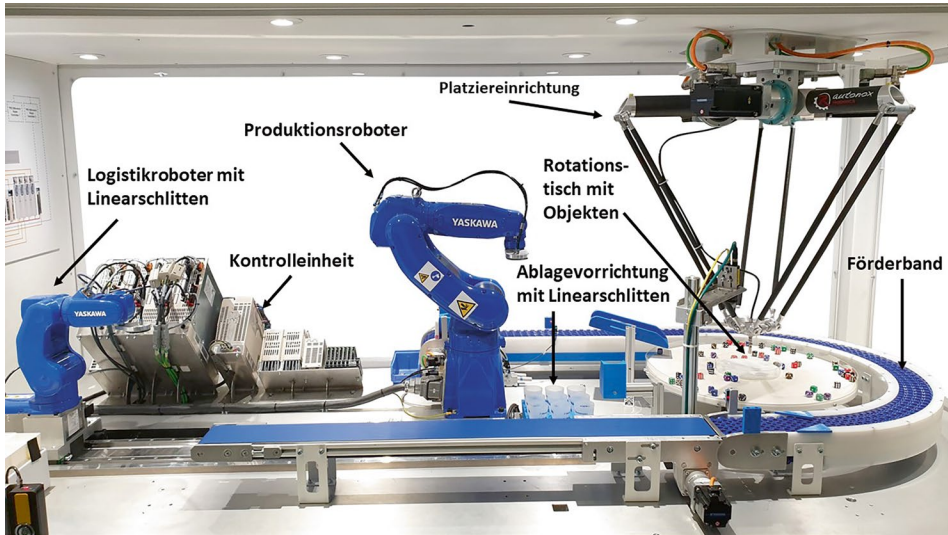


Abb. 11.1 Rotary Table Dispenser System (Goetz, C. und Humm, B. 2023)

Das CPPS ist dabei in drei Sektionen aufgeteilt, jede mit einer eigenen Prozesssequenz. Über eine externe Eingabe kann die Anzahl der zu sortierenden Objekte in jedem Container bestimmt werden. Insgesamt besteht das CPPS aus 9 CPS, 2 Robotersystemen, einer Platziereinrichtung, bestehend aus 3 Achsen, 4 Einzelachsen, und 9 Containern sowie einer zentralen Steuereinheit, welche den Produktionsprozess kontrolliert und eine intelligente Datensammlung¹ ermöglicht. Zusätzlich steht eine separate Recheneinheit zur Verfügung, mit der Funktionen wie ein intelligentes Ressourcenmanagement, Produktanalysen und Prozessoptimierungen realisiert werden können.

11.1.1 Anomaliedetektion in CPPS

Fehler innerhalb solch eines komplexen Systems können sich auf den gesamten Prozess auswirken und in einem beschädigten Produkt, einem Ausfall der Produktion oder in einer Verschleppung des Fehlers durch das gesamte System resultieren. Daher ist für die Realisierung von modernen und sicheren CPPS ein fehlerfreier Produktionsprozess essenziell (Ali et al. 2022).

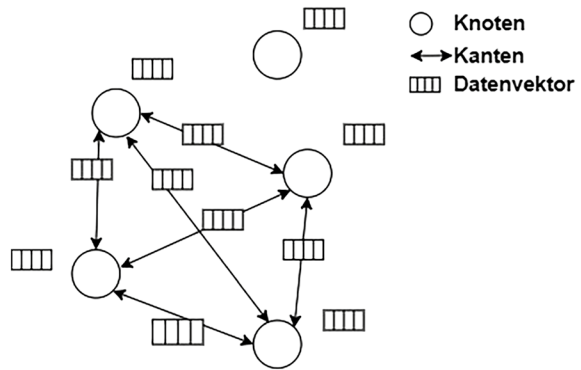
¹Eine intelligente Datensammlung zeichnet sich dabei durch eine synchronisierte Datenabfrage aller verbundenen CPS, sowie eine gleichzeitige Transformation der Daten innerhalb des Sammelprozesses, aus.

Anomalien, wie eine erhöhte Vibration an einer Achse eines Roboters oder ein unerwarteter Wechsel der Prozesssequenz (dies kann beispielsweise durch äußere Einflüsse, ein unerfahrener Benutzer ändert manuell den Zustand der Anlage oder interne Ursachen, Ausfall von Software- oder Hardwarekomponenten, resultieren), können dabei als mögliche Fehlerindikatoren dienen. *Anomaliedetektion* (AD) in CPPS bezieht sich dabei auf die Bestimmung von Systemcharakteristika, welche normalerweise während des regulären Betriebs des Systems nicht auftreten. Durch die frühzeitige Erkennung dieser Anzeichen besteht die Chance, auf eine Anomalie rechtzeitig zu reagieren, um so die Entstehung oder die Übertragung eines Fehlers auf die gesamte Produktion zu verhindern. Eine effektive AD bietet daher die Möglichkeit, einen möglichst fehlerfreien Produktionsprozess und somit auch die Entwicklung von intelligenten und sicheren CPPS sicherzustellen (Eiteneuer et al. 2019).

11.1.2 Einbindung von externem Wissen in die Anomaliedetektion

Um eine vollständige AD für ein CPPS zu realisieren, muss nicht nur die Erkennung von Anomalien in jedem integrierten CPS garantiert werden, sondern auch eine Erkennung innerhalb der Produkte, der Interaktionen und Abhängigkeiten und in den globalen Zuständen des Systems (Goetz, C. und Humm, B. G. 2023b). Dies kann beispielsweise die Abhängigkeit der verschiedenen Achsen des Roboters sein, wobei eine Anomalie innerhalb einer Achse auch die anderen beeinflussen kann. Darüber hinaus können auch Anomalien innerhalb von Interaktionen zwischen CPS und Produkten auftreten, wie z. B. eine unerwartete Kollision zwischen einem Container und dem Roboter während eines Aufnahmevorgangs. Produkte innerhalb des CPPS produzieren dabei meist keine eigenen Daten. Es besteht dennoch die Möglichkeit, den einzelnen Produkten Statusdaten, wie beispielsweise die aktuelle Position des Containers, zuzuweisen. Globale Parameter wie der aktuelle Zustand des Systems oder der momentane Produktionsschritt stellen weitere wichtige Informationen dar.

Die Entwicklung einer kompletten AD nur basierend auf Daten, welche von dem CPPS geliefert werden, ist kaum möglich. Strukturelles Wissen über die Anlage, wie beispielsweise die Anordnung der CPS innerhalb des Systems, muss von außen beige-steuert werden. Informationen beispielsweise über Bewegungsabläufe der Roboter liefern wichtige Erkenntnisse über die Interaktionen innerhalb des Prozesses. Hintergrundwissen über die integrierten CPS wie Anleitungen, Designs und Komponentenlisten können zusätzlich eingesetzt werden, um der AD zusätzliche Informationen bereitzustellen. Um diese Informationen nutzbar und innerhalb der AD einsetzbar zu machen, muss das Wissen aus zur Verfügung stehenden Datenquellen extrahiert, transformiert und in Verbindung mit den Echtzeitdaten des CPPS gebracht werden.

Abb. 11.2 Graphkonzept

11.1.3 Modellierung des Systems als Graph

Eine Möglichkeit, diese Verbindung zu realisieren, ist die Verwendung von Graphen. Graphen bieten eine effiziente Art, die strukturellen Charakteristiken eines Systems mit Echtzeitdaten und externem Hintergrundwissen zu verknüpfen. Dabei besteht ein Graph, wie in Abb. 11.2 dargestellt, aus mehreren Knoten, die über Kanten verknüpft sind. Unter Berücksichtigung des oben genannten CPPS können jedes CPS und jeder Container als separate Knoten modelliert werden. Interaktionen und Abhängigkeiten innerhalb des Systems stellen dabei die Kanten dar. Auch kann der Gesamtzustand der Anlage als separater Knoten dargestellt werden. Durch das Hinterlegen von externen Informationen und Echtzeitdaten in Form von Datenvektoren innerhalb der Graphkomponenten kann so eine vollständige Modellierung des Systems realisiert werden.

Im Folgenden stellen wir ein Konzept vor, welches es ermöglicht, das gesamte CPPS als einen Graphen zu modellieren, um dadurch Echtzeitdaten mit verfügbarem externen Wissen zu kombinieren. Dabei erläutern wir die verschiedenen internen und externen Datenquellen des CPPS sowie Techniken, die auf Graphen angewandt werden können. Weiter beschreiben wir eine Technik, welche die Erkennung und Evaluierung von Anomalien innerhalb dieser modellierten Graphen durch den Einsatz von *Graph Neural Networks* (GNN) ermöglicht. Dazu stellen wir das verwendete Modell sowie den internen Update-Prozess detailliert dar. Zuletzt erläutern wir einen Ansatz für eine korrelations- und interaktionsbewusste AD für CPPS basierend auf GNN und evaluieren diesen innerhalb eines realen industriellen Szenarios.

11.1.4 Übersicht über das Kapitel

Abb. 11.3 ordnet dieses Kapitel thematisch ein. Die behandelte Branche ist Industrie, die Anwendung ist Anomalieerkennung anhand von Zeitreihen; verwendete Technologien sind Deep Learning mit neuronalen Netzen und insbesondere Graph Neural Networks.

Abb. 11.3 Thematische Einordnung des Kap. 11

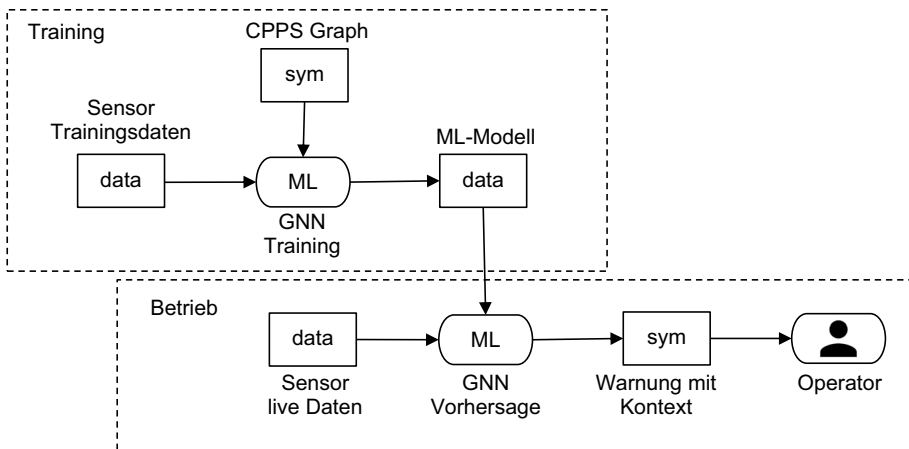
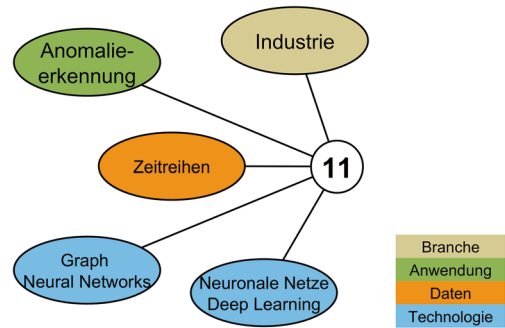


Abb. 11.4 Boxology-Modell des hybriden KI-Systems

Abb. 11.4 zeigt die Aspekte hybrider KI der vorgestellten Anwendung anhand eines Boxology-Modells.² Innerhalb des Trainings wird ein ML-Modell anhand der gesammelten Sensordaten in Kombinationen mit den modellierten Graphen des Systems trainiert. Anschließend wird das trainierte Modell zur Erkennung von Anomalien innerhalb des normalen Betriebs eingesetzt. Wird eine Anomalie erkannt, kann diese Erkennung mit zusätzlichen Informationen dem Benutzer zur Verfügung gestellt werden.

²Für eine Einführung in die Boxology siehe Kap. 1 sowie (van Harmelen und ten Teije 2019).

11.2 Datenquellen

Interne Daten des CPPS können in *Statusdaten* und *Prozessdaten* unterteilt werden. Statusdaten umfassen dabei alle Daten bezogen auf den Produktionsprozess, wie beispielsweise die Sequenz der einzelnen Produktionsschritte. Prozessdaten fassen die Daten jedes einzelnen integrierten CPS innerhalb des CPPS zusammen. Darunter fallen Echtzeitdaten wie das aktuelle Drehmoment und die Position des CPS. Externes Wissen kann in *strukturelles Wissen*, *Prozesswissen* und *Hintergrundwissen* aufgeteilt werden. Jede verfügbare Datenquelle bietet zusätzliche Informationen über das CPPS, die integrierten CPS und den Prozess. Dies kann für die Realisierung einer vollständigen AD genutzt werden und später zu einer Verbesserung der Interpretierbarkeit und Transparenz der Erkennung führen.

Statusdaten: Alle Daten, welche direkt in Verbindung mit der Produktion oder dem Status des CPPS stehen, definieren wir als Statusdaten. Dies sind Informationen über die Produkte, den Status der verschiedenen Prozesse oder den gesamten Zustand des CPPS, wie beispielsweise Umgebungsparameter der Anlage, Kontrollparameter und Materialinformationen. In den meisten Fällen sind diese Daten abhängig von internen Ereignissen und werden von der Kontrolleinheit des CPPS mit einer relativ langsamen Abtastrate von 20 ms aufgezeichnet.

Prozessdaten: Daten, die direkt von den CPS produziert werden, definieren wir als Prozessdaten. Im Fall eines CPS innerhalb des CPPS sind dies meist Zeitreihendaten der verschiedenen Variablen des CPS, wie beispielsweise die Position, Geschwindigkeit oder Drehmoment. Diese Daten werden meist regelmäßig mit einer hohen Abtastrate von 1 ms von jedem CPS aufgezeichnet.

Strukturelles Wissen: Als strukturelles Wissen können alle Informationen über den Aufbau, die Positionierung der CPS und das Layout des CPPS zusammengefasst werden. Diese Daten können dabei als Konzeptzeichnungen, Blaupausen, CAD-Zeichnungen oder in Form von digitalen Zwillingen zur Verfügung stehen.

Prozesswissen: Innerhalb eines CPPS werden verschiedene Prozesse, wie beispielsweise Bewegungsabläufe der Roboter oder Produktionsprozesse, von einzelnen CPS durchgeführt. Basierend auf diesen Informationen können Interaktionen und Abhängigkeiten zwischen den CPS und Produkten definiert werden.

Hintergrundwissen: Unter Hintergrundwissen fassen wir weitere Informationen zusammen, wie beispielsweise industrielle Standards, detaillierte Informationen über einzelne CPS oder historisches Wissen. Oft können diese Informationen in Serviceberichten, FAQs, Produktinformationen, Systembeschreibungen und Anleitungen gefunden werden.

11.3 Graphen

Wie zuvor beschrieben, bieten Graphen eine effiziente Art, die strukturellen Charakteristiken eines Systems mit Echtzeitdaten und externem Hintergrundwissen zu verknüpfen. Ein *Graph* $G = (V, E)$ kann, wie in Abb. 11.2 zu sehen, aus mehreren Knoten $v \in V$ und Kanten $e \in E$ bestehen (Scarselli et al. 2008). Knoten repräsentieren dabei einzelne Instanzen und Kanten stellen die Beziehungen der Knoten zueinander dar. Jede Kante kann dabei eine Richtung aufweisen, welche durch Sender- und Empfängerknoten spezifiziert wird. Innerhalb des Graphen können zusätzliche Charakteristika in Form von externen Informationen oder Datenvektoren den einzelnen Elementen hinterlegt werden.

11.3.1 Anwendungen

Mit Graphen können Strukturen aller Art modelliert werden, beispielsweise soziale Strukturen, Straßenkarten oder chemische Verbindungen (Ji et al. 2021). Auch technische Systeme mit ihren internen Strukturen, Interaktionen und Abhängigkeiten können durch Graphen repräsentiert werden (Zhang et al. 2020). Verschiedene Ansätze und Techniken ermöglichen es, einzelne Knoten, Kanten oder ganze Graphen automatisch zu klassifizieren, vorherzusagen oder zu verändern (Cai et al. 2018). Weiter können verschiedene Abschnitte innerhalb des Graphens gruppiert, Ähnlichkeiten zwischen Graphen festgestellt oder die kürzeste Verbindung zwischen Knoten gefunden werden (Wang et al. 2017). In jüngster Zeit finden Graphen auch vermehrt Anwendung im Machine Learning (ML; Wu et al. 2020).

Ein Anwendungsbereich, auf welchen speziell hier eingegangen werden soll, ist die AD innerhalb von Graphen. Dazu können *GNN* eingesetzt werden, die es ermöglichen, Techniken aus dem Feld des Deep Learning auf Graphen anzuwenden. Beispiele dafür können in Ma et al. (2021) gefunden werden. Durch den Einsatz von Techniken aus dem Deep Learning können große und komplexe Datenmengen effektiv verarbeitet und gleichzeitig durch das Graphkonzept externes Wissen eingebunden werden, wodurch sich diese Techniken besonders für die Anwendung innerhalb industrieller CPPS eignen (Deng und Hooi 2021).

11.3.2 Graphrepräsentation des CPPS

In Bezug auf das oben gezeigte Rotary Table Dispenser System können jedes CPS und jeder Container jeweils als individuelle Knoten dargestellt werden. Interaktionen und Wechselbeziehungen innerhalb des Systems können als Kanten zwischen den Knoten modelliert werden, dabei definiert die Richtung der Kante auch die Richtung der Interaktion. Globale Zustände des Systems können durch einen Kontextknoten dargestellt werden. Das Rotary Table Dispenser System ist in Abb. 11.5 als Graph dargestellt. Wich-

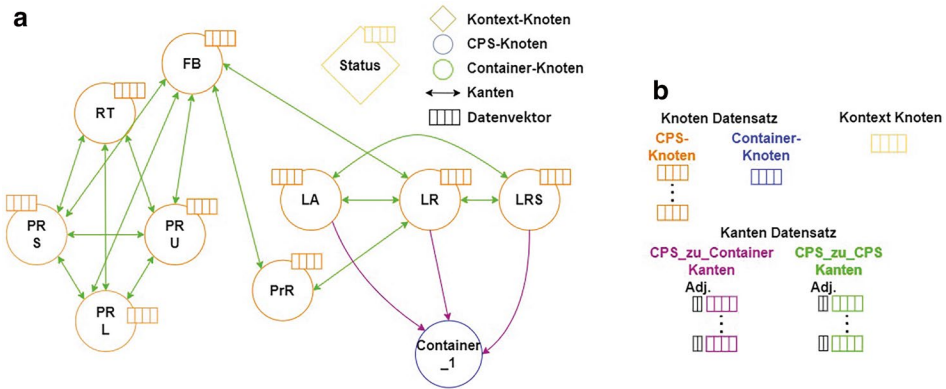


Abb. 11.5 Beispielgraph des Rotary Table Dispenser System mit einem Container. **a)** Graphdarstellung **b)** Datendarstellung. FB = Förderband; RT = Rotationstisch; PR S = Platziereinrichtung Achse S; PR U = Platziereinrichtung Achse U; PR L = Platziereinrichtung Achse L; PrR = Produktionsroboter; LA = Linearschlitten Ablage; LR = Logistikroboter; LRS = Linearschlitten Logistikroboter. (Angepasst von Goetz, C. und Humm, B. (2023b))

tige Daten, wie beispielsweise Prozessdaten, Charakteristika der Interaktionen oder globale Zustände, können dabei den einzelnen Komponenten des Graphen als Datenvektoren hinterlegt werden. Durch die Kombination aller Vektoren eines Graphenteils, z. B. aller CPS-Knoten, entstehen kombinierte Datensätze. Für die Datensätze der Kanten innerhalb des Graphen kann zusätzlich eine Adjazenzmatrix definiert werden, welche die jeweiligen Sender- und Empfängerknoten deklariert. Dies ermöglicht es, den gesamten Graphen in einem Datenformat zu repräsentieren, ohne essenzielle Charakteristika des Graphen zu vernachlässigen. Durch solch eine Modellierung ist es möglich, das gesamte CPPS, inklusive Echtzeitdaten, externes Wissen und Hintergrundinformationen als Graph darzustellen.

Darüber hinaus kann das externe Wissen dazu verwendet werden, eine Analyse nach der Erkennung einer Anomalie durchzuführen. Beispielsweise kann Hintergrundwissen, welches bei der Modellierung des Graphen verwendet wurde, nach einer Erkennung abgerufen und zur Verfügung gestellt werden. Dies kann den Benutzer der Anlage unterstützen, die Anomalie zu analysieren, und im besten Fall eine Lösung zur Fehlerbehebung ermöglichen.

11.4 Integrationskonzept

Für die erfolgreiche Umsetzung einer Anomaliedetektion für CPPS basierend auf GNN müssen verschiedene Prozesse etabliert werden. Dabei lassen sich diese in vier separate Pipelines, die *Wissens-Pipeline*, *Daten-Pipeline*, *Modell-Pipeline* und die *Anomalie-*

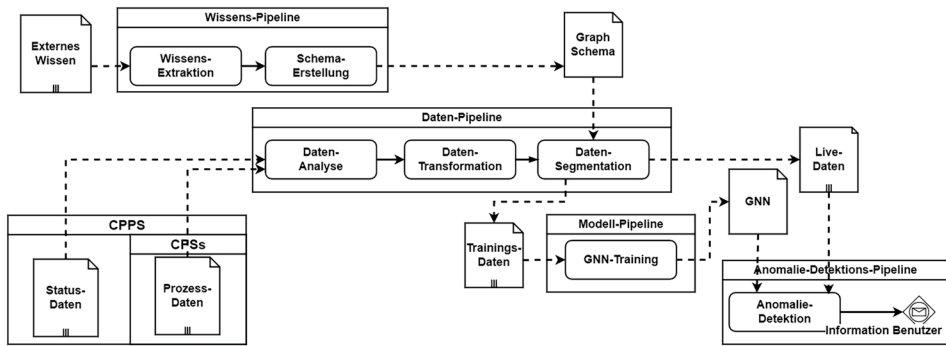


Abb. 11.6 Überblick Integrationskonzept. (Nach Goetz, C. und Humm, B. (2023b))

iedetektions-Pipeline, aufteilen. Abb. 11.6 zeigt die Abfolge der verschiedenen Prozesse innerhalb des Konzepts als BPMN-Diagramm.

11.4.1 Wissens-Pipeline

Im Prozessschritt *Wissensextraktion* wird zunächst das externe Wissen aus den entsprechenden Datenquellen extrahiert. Abhängig von den Daten müssen Extraktionskonzepte entwickelt werden. Innerhalb der *Schemaerstellung* wird das extrahierte Wissen verwendet, um ein Graphschema zu erstellen sowie wichtige Informationen für die Daten-Pipeline zu generieren. Innerhalb dieses Schemas werden die Graphteile mit Datentypen und Dimensionen des Datenvektors definiert.

11.4.2 Daten-Pipeline

Innerhalb des Prozessschrittes *Datenanalyse* werden die Status- und Prozessdaten von den individuellen CPS sowie dem CPPS gesammelt. Im Schritt der *Datentransformation* werden Vorverarbeitungsroutinen, z. B. Skalierungen, Normalisierungen und Synchronisierungen für die Daten, ausgeführt. In der *Datensegmentierung* werden die transformierten Daten mit dem Graphschema verbunden und darauf basierende Graphen generiert.

11.4.3 Modell-Pipeline

Für das *GNN-Training* werden ausschließlich normale Prozessdaten in Form von Trainingsdaten verwendet. Nach Abschluss des Trainings kann das Modell der Anomaliedetektions-Pipeline zur Verfügung gestellt werden.

11.4.4 Anomaliendetektions-Pipeline

In der Anomaliendetektions-Pipeline werden Anomalien durch das GNN innerhalb von Live-Daten erkannt (Prozessschritt *Anomaliendetektion*). Diese Datenpakete werden basierend auf neuen Daten des CPPS erzeugt und durch die zuvor etablierte Daten-Pipeline verarbeitet. Nach einer erfolgreichen Erkennung einer Anomalie kann der Benutzer des Systems informiert sowie eine detaillierte Analyse und zusätzliches hinterlegtes Wissen zur Verfügung gestellt werden.

11.5 Graph Neural Network

11.5.1 Modellentwicklung

Wie oben beschrieben, kann ein CPPS als ein Graph repräsentiert werden, welcher aus drei Teilen besteht: (a) Knoten, welche die CPS und Produkte repräsentieren, (b) Kanten, welche die Interaktionen und Wechselbeziehungen zwischen den Knoten darstellen, sowie (c) einem Kontextknoten, welcher den globalen Zustand des CPPS beinhaltet. Die Datenvektoren von jedem Graphteil können zu Datensätzen kombiniert werden. Das GNN kann anschließend die transformierten und segmentierten Graphdaten verarbeiten. Das GNN-Konzept ist in Abb. 11.7 dargestellt.

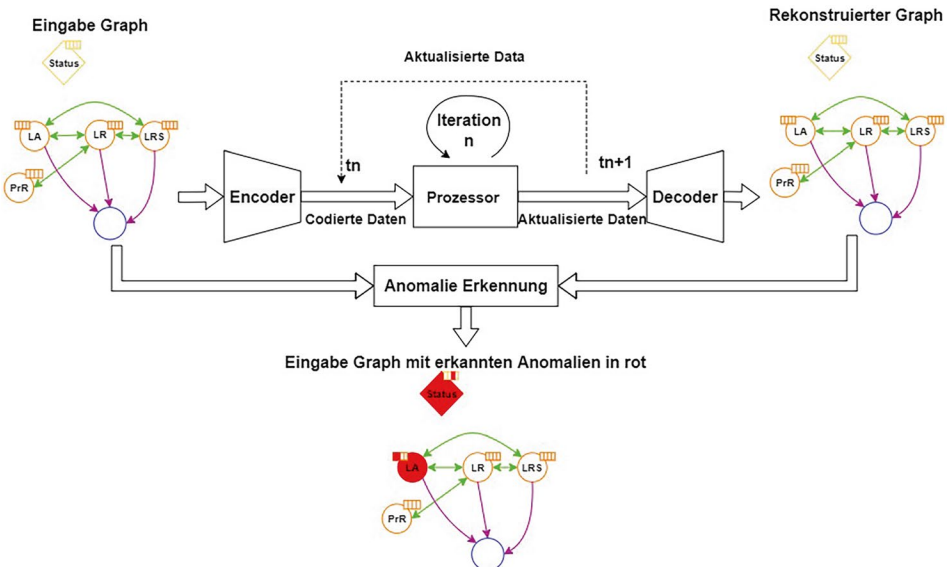


Abb. 11.7 GNN-Autoencoder-Konzept. (Angepasst von Goetz, C. und Humm, B. (2023b))

Das GNN besteht aus drei Komponenten: dem Encoder, dem Prozessor und dem Decoder (Battaglia et al. 2018). Innerhalb jeder Komponente werden die verschiedenen Datensätze des Graphen separat bearbeitet, wofür im GNN mehrere Neural Networks (NN) genutzt werden. Graphdaten dienen als Eingabe des Encoders. Wie bei Autoencodern üblich, werden die Dimensionen der Eingabe verringert und wichtige Charakteristika innerhalb der Daten herausgefiltert. Die Encoder-Daten werden anschließend dem Prozessor übermittelt, welcher verschiedene Aktualisierungen und Aggregationen durchführt. Ein Aktualisierungsprozess wird im folgenden Abschnitt beschrieben. Der Aktualisierungsschritt kann als ein iteratives Verfahren betrachtet werden, das mit einer bestimmten Anzahl von Wiederholungen ausgeführt wird. Innerhalb jeder Iteration werden Daten durch den Graphen transportiert, wodurch Knoten und Kanten mit Charakteristika von verbundenen Knoten beeinflusst werden. Führt man mehrere Iterationen durch, werden so Charakteristika durch den gesamten Graphen transportiert und beeinflussen auch nicht direkt verbundene Knoten. Zuletzt werden innerhalb des Decoders die Dimensionen der Eingabedaten wiederhergestellt und der Graph basierend auf den verarbeiteten Daten rekonstruiert. Der anschließende Prozess zur Anomaliedetektion wird im Abschnitt *Anomaliedetektion* erläutert.

11.5.2 Aktualisierungsprozess

Der Aktualisierungsprozess kann als eine Sequenz von verschiedenen Schritten beschrieben werden (Ferludin et al. 2022). Jeder Graphenteil wird dabei separat aktualisiert. Eine Möglichkeit, die hier näher erläutert werden soll, ist in Abb. 11.8 dargestellt. Dabei werden zuerst die einzelnen Kanten basierend auf dem Sendeknoten nT und Empfängerknoten nR , dem Kontextknoten c und dem eigenen Zustand e zu e' aktualisiert. Für die Aktualisierung des Knotens nR zu nR' werden die direkt anliegenden Kanten $e1'$, $e2'$, $e3'$, der Kontextknoten c und der eigene Zustand verwendet. Zuletzt wird der Kontext-

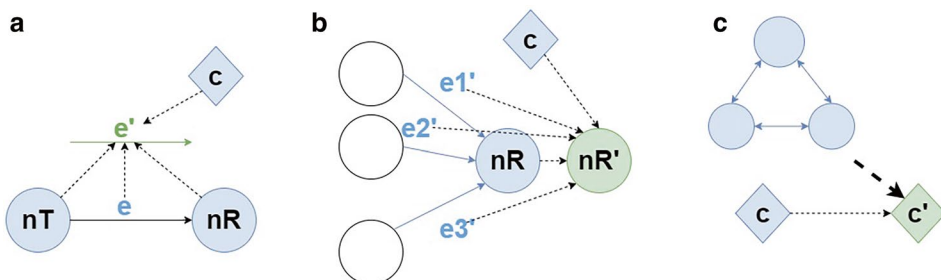


Abb. 11.8 Aktualisierungsprozess **a)** Aktualisierung Kanten, **b)** Aktualisierung Knoten, **c)** Aktualisierung Kontext. Grün: Aktualisiertes Graphenteil, blau: Graphenteile verwendet für die Aktualisierung. (Angepasst von Goetz, C. und Humm, B. (2023b))

knoten c anhand aller Knoten und Kanten zu c' aktualisiert. Diese Routine wird für jede Kante sowie jeden Knoten durchgeführt. In Bezug auf ein CPPS werden jedes CPS, die Interaktionen und Abhängigkeiten der CPS und Produkte sowie der globale Status verwendet, um den Graphen iterativ zu aktualisieren.

11.5.3 Anomaliedetektion

Wie in Abb. 11.7 dargestellt ist die Ausgabe des GNN der rekonstruierte Graph. Es ist zu beachten, dass das GNN mit normalen Prozessdaten trainiert wurde, also zu einer Zeit, in der der Produktionsprozess der Anlage fehlerfrei lief. Durch den Abgleich der Eingabe, die aktuell gemessenen Daten, mit der Ausgabe, die rekonstruierten Daten, kann ein Rekonstruktionsfehler ermittelt werden. Dabei beruht das Prinzip der AD durch Rekonstruktion darauf, dass die Rekonstruktion sich der Eingabe annähert und somit der Rekonstruktionsfehler gering ist. Im Falle einer Anomalie steigt der Rekonstruktionsfehler, da das GNN anhand des fehlerfreien Prozesses trainiert wurde und somit anomale Daten nicht korrekt rekonstruiert werden. Ist der Trainingsprozess abgeschlossen, wird anschließend der Rekonstruktionsfehler jedes im Training verwendeten Graphen aufgezeichnet. Durch die Aggregation dieser Rekonstruktionsfehler können sinnvolle Grenzwerte definiert werden. Überschreitet ein Rekonstruktionsfehler diesen Grenzwert, so wird eine Anomalie angenommen. Durch die Definition verschiedener Grenzwerte für die einzelnen Graphenteile können Anomalien innerhalb des Graphen zugeordnet werden. Basierend auf dem Aktualisierungsvorgang und der Struktur findet auch eine Verschleppung der Anomalie innerhalb des Graphen statt. Abhängig von der Ausprägung der Anomalie innerhalb jedes Teils des Graphen können auch Kanten, Knoten oder der Kontextknoten durch die Anomalie beeinflusst werden. Dies kann z. B. eine starke Anomalie innerhalb eines CPS sein, welche sich auch auf die angrenzenden Knoten und deren Kanten auswirkt.

11.6 Prototypische Implementierung

Um das Integrationskonzept zu evaluieren, wurde ein Prototyp für das CPPS aus Abb. 11.2 erstellt (Abb. 11.9). Bei der Implementierung eines Prototyps müssen Anforderungen an die jeweilige Pipeline und die Eigenschaften des CPPS berücksichtigt werden. Vor allem die Extraktion von Informationen aus *externen Wissensquellen* ist stark abhängig von der Verfügbarkeit, der Art und dem Format der jeweiligen Daten. Als Programmiersprache wurde Python Version 3.10 verwendet. Der Prototyp wurde auf einer *zentralen Recheneinheit*, einem PC mit einem Intel i9-12900 K Prozessor und einer NVIDIA Geforce RTX3080Ti GPU, entwickelt und getestet.

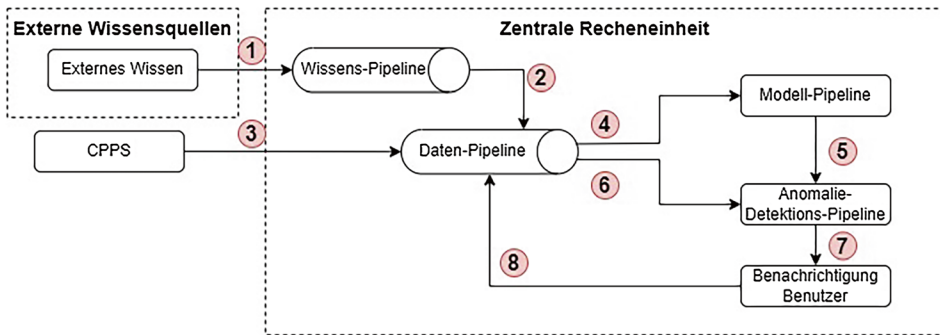


Abb. 11.9 Überblick Prototypenimplementierung mit nummeriertem Datenfluss

11.6.1 Datenfluss

Abb. 11.9 zeigt den Ablauf der verschiedenen Prozessschritte des Prototypen. Im folgenden Abschnitt werden die einzelnen Punkte des Datenflusses, die in der Abbildung markiert sind, kurz beschrieben.

ad 1: Strukturierte, semistrukturierte und unstrukturierte Daten in unterschiedlichen Dateiformaten, wie beispielsweise Bilder (JPEG, PNG), Dokumente (PDF, DOCX), Video (MP4, MOV) oder Datenbanken (Relational, Dokument, Zeitreihen), stehen als externes Wissen über das CPPS zur Verfügung und werden an die Wissens-Pipeline übergeben.

ad 2: Das erstellte Graphschema, ein Proto Buffer Text File,³ wird von der Wissens-Pipeline an die Daten-Pipeline für die Segmentierung der CPPS-Daten übertragen.

ad 3: 48 verschiedenen Prozess- und Statusdaten in Form von Zeitreihen werden von den CPPS durch ein OPC-DA⁴-Server-Client-Modell an die Daten-Pipeline geliefert.

ad 4: Von der Daten-Pipeline werden die gesammelten Trainingsdaten, segmentierte Graphdatenpakete im TFRecord⁵-Format, an die Modell-Pipeline gesendet.

ad 5: Nach Abschluss des Trainings des Modells wird das Modell an die Anomaliedetektions-Pipeline übergeben.

ad 6: Live-Daten des CPPS in Form von TFRecord-Graphdatenpakete werden von der Daten-Pipeline an die Anomaliedetektions-Pipeline übertragen.

³ <https://protobuf.dev> – Ein sprach- und plattformneutrale Darstellung von strukturierte Daten in Textform.

⁴ <https://opcfoundation.org/about/opc-technologies/opc-classic>

⁵ https://www.tensorflow.org/tutorials/load_data/tfrecord

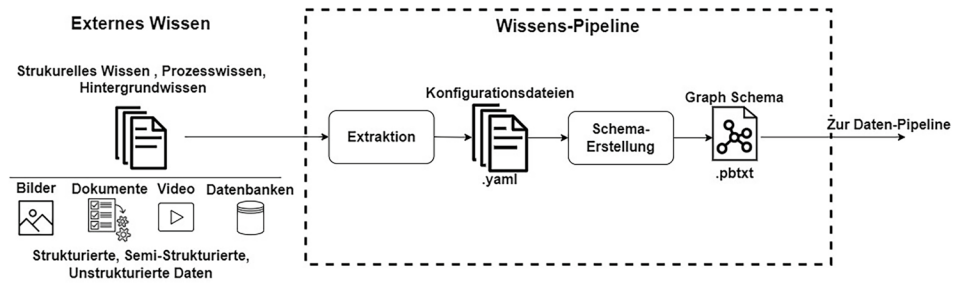


Abb. 11.10 Wissens-Pipeline des Prototyps

ad 7: Bei einer Detektion wird der Benutzer benachrichtigt und die Ergebnisse in Form von JSON Files⁶ zur Verfügung gestellt. Diese können zusammen mit den aktuellen Daten des CPPS genutzt werden, um die erkannten Anomalien zu visualisieren.

ad 8: Die Ergebnisse der Anomaliedetektions-Pipeline werden in der Datenbank der Daten-Pipeline hinterlegt.

11.6.2 Externe Wissensquellen

Wie in Abb. 11.10 dargestellt, kann *externes Wissen* über das CPPS in unterschiedlichen Dateiformaten, wie beispielsweise Bild, PDF, Video oder in Datenbanken, sowie in strukturierter, semistrukturierter oder unstrukturierter Form verfügbar sein. Innerhalb des Prototyps wurde alles verfügbare externe Wissen über das CPPS und CPS gesammelt. Als strukturelles Wissen wurden ein Designdiagramm sowie eine CAD-Zeichnung verwendet. Dabei konnten anhand des Designdiagramms die Positionen der verschiedenen CPS innerhalb des CPPS ermittelt werden und die CAD-Zeichnung stellte eine Blaupause für das gesamte mechanische System dar. Als Prozesswissen wurden die Bewegungsabläufe der Roboter und eine detaillierte Beschreibung der verbauten Sensorik und ihre Auslösebedingungen verwendet. Zusätzlich wurde der Prozessablauf des Produktionsprozesses basierend auf PackML,⁷ einem industriellen Standard zur Steuerung von Produktionsmaschinen, zur Verfügung gestellt. Als Hintergrundwissen waren FAQs, Handbücher und Serviceberichte über die CPS und das CPPS als PDF-Dokumente sowie das komplette Steuerprogramm mit Variablendeklarationen verfügbar. Da es sich bei der Implementierung um einen Prototypen handelt, wurden innerhalb des experimentellen Aufbaus viele externe Informationen per Hand in die Konfigurationsdateien extrahiert.

⁶<https://www.json.org>

⁷<https://www.omac.org/packml>

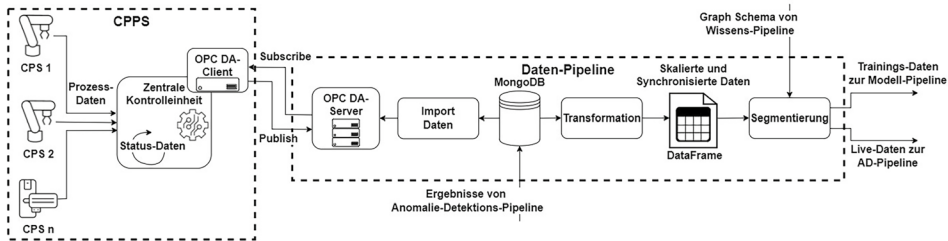


Abb. 11.11 Daten-Pipeline des Prototyps

11.6.3 Wissens-Pipeline

Um eine universelle Schnittstelle zwischen den unterschiedlichen Datentypen und der *Schemaerstellung* zu etablieren, wurde bei den Prototypen mit verschiedenen *Konfigurationsdateien* im YAML-Format⁸ gearbeitet. Diverse Konzepte zur Extraktion unterschiedlichster externer Daten können dadurch auch noch nachträglich in den Prototypen eingebunden werden. Dabei muss die *Extraktion* der Daten im Format der Konfigurationsdatei resultieren, um die Informationen innerhalb des Prototyps zu verwenden. Basierend auf diesen *Konfigurationsdateien* kann ein *Graphschema* erstellt werden. Innerhalb des Schemas werden die Datenvektoren mit Dimensionen und Datentypen für jeden Datensatz definiert. Zusätzlich müssen für die Kanten die Send- und Empfängerknoten angegeben werden. Solch ein Schema kann als *Proto Buffer Text File*, eine allgemein lesbare Form des Protocol-Buffers-Datenformats, gespeichert werden. Das Graphschema kann anschließend an die *Daten-Pipeline* für die *Segmentierung* übergeben werden.

11.6.4 Daten-Pipeline

Um die Daten des CPPS effektiv sammeln und speichern zu können, wurde für die Daten-Pipeline (Abb. 11.11) eine *MongoDB*⁹ integriert. MongoDB ist ein dokumentenorientiertes NoSQL-Datenbank-Managementsystem. Als Kommunikationsmodell, zwischen der Datenbank und dem CPPS, wurde ein *OPC-DA-Server-Client*-Modell basierend auf *Publish-Subscribe*-Routinen etabliert. Innerhalb der *Transformation* wurden abhängig der importierten Daten verschiedene Vorverarbeitungsroutinen mithilfe der Python-Bibliothek *Scikit-Learn*¹⁰ implementiert. Die Prozessdaten, verschiedene

⁸ <https://yaml.org>

⁹ <https://www.mongodb.com>

¹⁰ <https://scikit-learn.org>

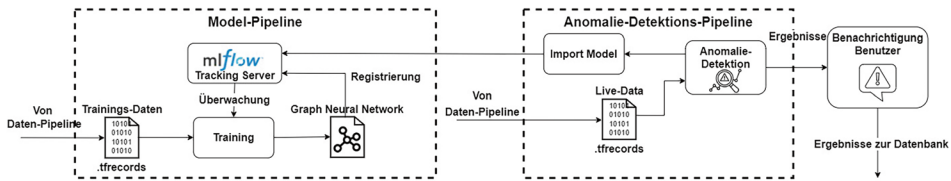


Abb. 11.12 Modell-Pipeline und Anomaliedetektions-Pipeline des Prototyps

Zeitreihen des CPS, wurden mit Methoden der Klasse *MinMaxScaler* in einen numerischen Bereich zwischen 0 und 1 skaliert. Anschließend wurde eine Synchronität mit den Statusdaten durch Verwendung von *Sliding Windows* realisiert. Für die Statusdaten wurden *OrdinalEncoder* verwendet, um die numerischen Bereiche zu reduzieren und gleichzeitig die Datenstruktur zu erhalten. Der daraus resultierende Pandas¹¹ *DataFrame* beinhaltet die synchronisierten und skalierten Daten des CPPS. In der *Segmentierung* werden anschließend die transformierten Daten basierend auf dem Graphschema in TFRecord-Graphdatenpakete, ein einfaches Format zur Speicherung von Binärdatensätzen, serialisiert.

11.6.5 Modell-Pipeline

Die Modell-Pipeline ist in Abb. 11.12 dargestellt. Für die Implementierung des GNN wurde die Bibliothek *TF-GNN*,¹² eine TensorFlow-GNN-Bibliothek (Ferland et al. 2022), gewählt. Die integrierten NNs innerhalb des GNN wurden basierend auf *TensorFlow*¹³ erstellt. Für das *Training* der verschiedenen Modelle werden die gesammelten *Trainingsdaten* verwendet. Die Überwachung und Verwaltung der experimentellen Resultate und Modelle wurde mithilfe eines *MLFlow*¹⁴-Tracking-Servers etabliert. Innerhalb diesem werden die erstellten Modelle registriert und können anschließend in der *Anomaliedetektions-Pipeline* importiert werden.

11.6.6 Anomaliedetektions-Pipeline

Die *Live-Daten* werden anschließend durch das importierte Modell innerhalb der *Anomaliedetektion* verarbeitet. Bei einer Detektion wird der Benutzer benachrichtigt und

¹¹ <https://pandas.pydata.org/>

¹² <https://github.com/tensorflow/gnn>

¹³ <https://www.tensorflow.org/>

¹⁴ <https://mlflow.org/>

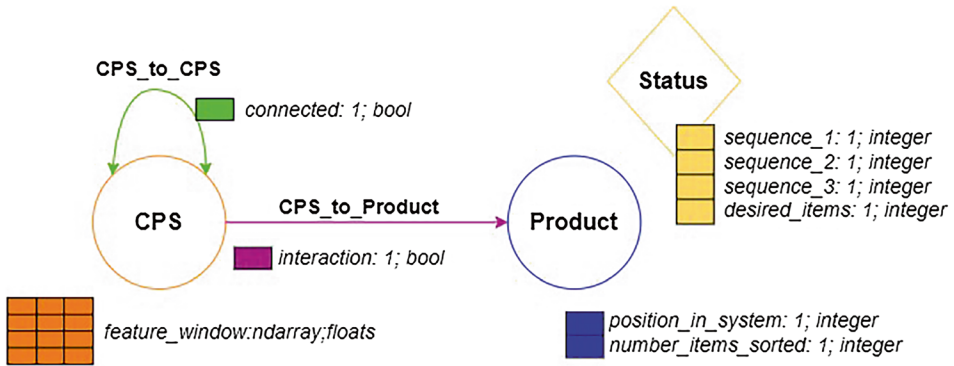


Abb. 11.13 Graphschema des Rotary Table Dispenser System; jede Komponente ist definiert mit Name:Dimension;Datentyp. (Nach Goetz, C. und Humm, B. (2023b))

eine aktuelle Aufnahme der CPPS-Daten mit den Ergebnissen dem Benutzer zur Verfügung gestellt. Diese werden zusätzlich an die Datenbank geliefert und dort für spätere Analysen hinterlegt.

11.6.7 Entwicklung des Graphschemas

Basierend auf den gesammelten Informationen konnte ein Graphschema, wie in Abb. 11.13 ersichtlich, erstellt werden. Dabei dient das Schema als eine Blaupause des CPPS, anhand dessen die Graphen generiert werden. Dafür werden sowohl die verschiedenen Typen von Knoten und Kanten sowie deren Variablen, inklusive Datentyp und Dimensionierung, deklariert.

Zuerst wurde der CPS-Knoten durch die strukturellen Informationen, das Kontrollprogramm und die verfügbaren Datenstrukturen definiert. Dabei wurde ein Designdiagramm der Anlage verwendet, um die Anzahl der CPS-Knoten zu bestimmen und das Kontrollprogramm aus Abb. 11.14 für die Dimensionierung der Datenvektoren mit Datentypen. Die Datenvektoren bestehen dabei aus mehrdimensionalen Arrays in Form von Sliding Windows der Prozessdaten, wie z. B. die aktuelle Position, Drehmoment und Geschwindigkeit, deklariert als *feature_window*.

Der Produktknoten wurde durch die Informationen des Produktionsprozesses generiert. Die Variable *position_in_system*, welche die aktuelle Position der Container innerhalb des Systems definiert, konnte mithilfe der Beschreibung der Aktivierungsreihenfolge der Sensorik erstellt werden. Die Definition der Anzahl an bereits platzierten Objekten in den Container wird durch die Variable *number_items_sorted* angegeben. Dabei ist jede Variable nur durch einen Integer-Wert definiert.

Die zwei Kantensets wurden anhand der Bewegungsmuster der Roboter in Kombination mit der CAD-Zeichnung des mechanischen Systems (Abb. 11.15) und den Infor-

<pre> (* Motion *) LimitVelocity : LREAL; LimitAccel : LREAL; LimitDecel : LREAL; LimitJerk : LREAL; Velocity : LREAL; Acceleration : LREAL; Deceleration : LREAL; (* EmptyCupVelo : LREAL; </pre>		
MotionWorks IEC	YASKAWA <small>Yaskawa America, Inc. 2121 Norman Drive South Westwego, IL 60085 USA</small>	Sheet number 6

Abb. 11.14 Ausschnitt des Kontrollprogramms mit Variablendeklarationen und Datentypen einer Achse

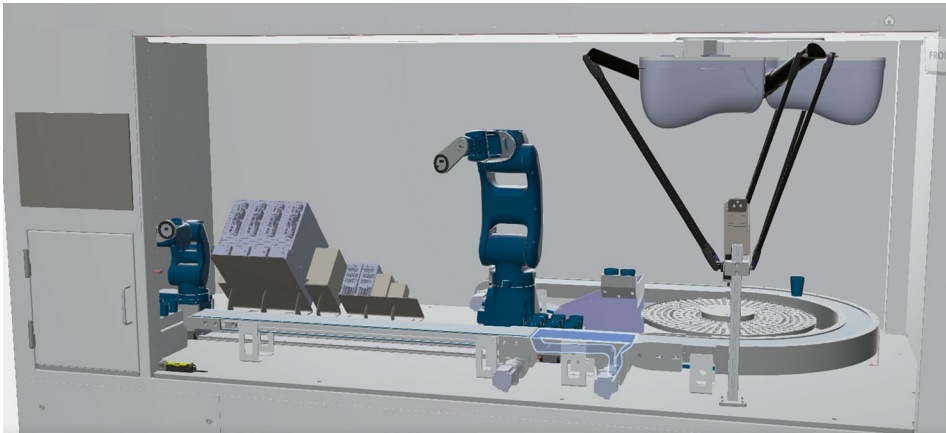


Abb. 11.15 CAD-Zeichnung Rotary Table Dispenser System

mationen des Produktionsprozesses definiert. Die CAD-Zeichnung bestimmt dabei den Kontaktradius jedes CPS innerhalb des CPPS. Die Kante zwischen den CPS-Knoten wird durch *CPS_to_CPS* angegeben und beinhaltet die boolesche Variable *Connected*. *CPS_to_Produkt* definiert die Interaktion zwischen CPS und Produkt mit der booleschen Variablen *Interaction*.

Der Statusknoten enthält Informationen über die Sequenzen des Produktionsprozesses und des Kontrollprogramms. Dies beinhaltet den jeweiligen aktuellen Status der PackML-Sequenz jedes Sektors als Integer, *sequence_1*, *sequence_2*, *sequence_3* und die gewünschte Anzahl von Produkten *desired_items* in jedem Container. Ein Ausschnitt dieser PackML-Sequenz ist in Abb. 11.16 dargestellt.

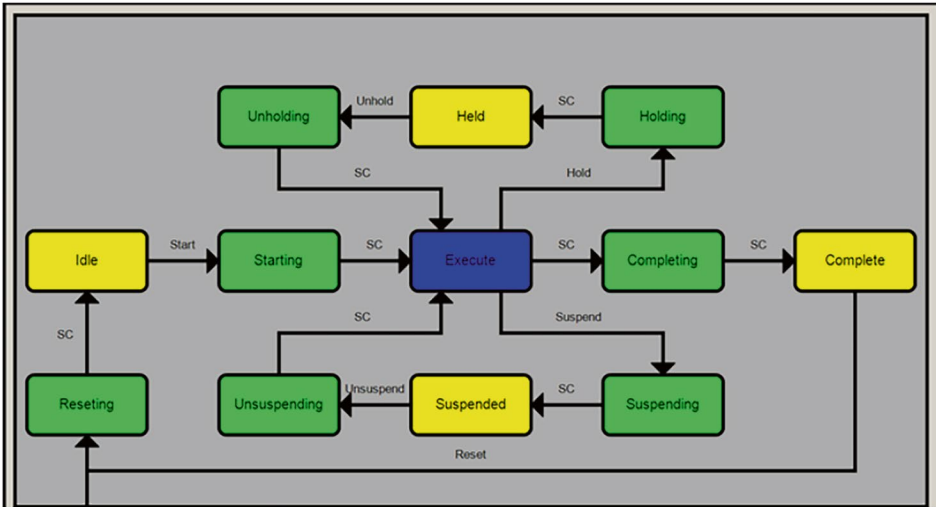


Abb. 11.16 Ausschnitt PackML-Sequenz

11.6.8 Graphmodellierung

Durch die Kombination der Status- und Prozessdaten mit dem erstellten Graphschema kann abschließend der komplette Graph des CPPS erstellt und externes Wissen hinterlegt werden. Dabei verändern sich die Datenvektoren der einzelnen Knoten sowie die Kanten abhängig vom aktuellen Prozess.

11.6.9 Modellkonfiguration

Wie im Abschnitt Graph Neural Network beschrieben, besteht ein GNN aus mehreren NN abhängig von dem CPPS. Innerhalb des Experiments wurden mehrschichtige Perzeptron-Modelle (MLP) als NN verwendet. Das dadurch resultierende GNN besteht aus 15 NN, verteilt auf die Graphteile, die verschiedenen Knoten und Kanten.

11.6.10 Modelltraining

Der Aktualisierungsprozess wurde so adaptiert, dass der Einfluss des Kontextknotens im Fall einer Kontextanomalie minimiert wurde. Dafür wurde der Kontextknoten nicht in die Aktualisierung der Kanten und anderen Knoten mit einbezogen. Als Verlustwert wurde der Rekonstruktionsfehler der jeweiligen Graphteile verwendet. Es wurde ein Adam-Optimierer eingesetzt und die Hyperparameter manuell angepasst.

11.7 Evaluierung

Um den implementierten Prototyp zu evaluieren, wurden verschiedene Tests an dem in Abb. 11.1 beschriebenen CPPS, dem Rotary Table Dispenser System, durchgeführt.

11.7.1 Experimenteller Aufbau

Das System besteht dabei aus verschiedenen CPS, einer zentralen Kontrolleinheit und einer Recheneinheit, die zusammen den Produktionsprozess realisieren. Während des Prozesses werden die Prozessdaten jedes einzelnen CPS mit einer Abtastrate von 2 ms und die Statusdaten des CPPS bei einer langsameren Zykluszeit von 20 ms aufgezeichnet. Insgesamt wurden 33 Prozessdatenzeitreihen und 15 Statusdatenzeitreihen aufgezeichnet, was für einen kompletten Produktionszyklus in ungefähr 11.767.000 individuellen Datenpunkten resultiert.

11.7.2 Data Recording

Um das Konzept zu testen, wurde der normale Prozess gezielt beeinflusst, um so realistische Anomalien zu erzeugen. Die entstandenen anomalen Bereiche innerhalb der Daten wurden manuell klassifiziert, um die Performanz des Modells evaluieren zu können. 5 Fehlerfälle wurden erstellt, drei davon innerhalb eines CPS, eine globale Anomalie und eine Anomalie im Produkt.

Reibung: Als Erstes wurde Reibung simuliert, indem eine manuelle Kraft gegen die Drehrichtung des Förderbandes aufgebracht wurde. Dies führte zu einem erhöhten Drehmoment des jeweiligen CPS. In einem realen Szenario kann dies durch den Verschleiß einer mechanischen Komponente oder ein gebrochenes Bauteil verursacht werden.

Vibration: Eine undefinierte Vibration innerhalb des CPS wurde durch manuelle Stöße an den mechanischen Komponenten des Drehtisches erzeugt. Dies führte zu einem undefinierten, kurzzeitigen Rauschen in den Drehmomentwerten des CPS. Diese Anomalie kann in industriellen Anlagen durch lose Verbindungen oder ein gebrochenes Lager verursacht werden.

Kollision: Kollisionen wurden simuliert, indem Objekte in die Bewegungsrichtungen der CPS platziert wurden, was zu starken Ausschlägen in den Drehmomentdaten führte. Ein reales Szenario wäre hier die Verursachung durch äußere Einflüsse oder Prozessfehler.

Globale Anomalie: Der Zustand des Prozessablaufs wurde während des Betriebs zufällig verändert. Dies kann z. B. durch eine absichtliche Einwirkung von außen, wie die Veränderung des Prozessablaufs durch eine manuelle Eingabe in das Steuerprogramm, verursacht werden. Dabei kann der Gesamtprozess gestört werden, um das System absichtlich zu schädigen.

Produktanomalie: Die Aufnahme von Produkten wurde manuell beeinflusst, sodass die Container nicht korrekt befüllt wurden. Dies kann z. B. durch eine verschlissene Saugglocke verursacht werden. Auch wurde die Position der Container im laufenden Betrieb manuell verändert, was beispielsweise durch einen fehlerhaften Produktionsschritt resultieren kann und zu einer undefinierten Position der Container innerhalb des Systems führt.

11.7.3 Experimentelle Resultate

Anomaliedetektion: Die experimentellen Ergebnisse für jeden Fehlerfall sind in Tab. 11.1 zu sehen. Die Ergebnisse wurden anhand der Erkennung der zuvor manuell klassifizierten anomalen Bereiche evaluiert, wobei auch die Auswirkungen der einzelnen Anomalien auf verbundene Komponenten innerhalb des Graphen berücksichtigt wurden. Dies kann beispielsweise eine starke Anomalie innerhalb des Förderbandes sein, welche sich auch auf die Container auf dem Band auswirkt. In solch einem Fall wurden auch die Container zu dem Zeitpunkt der Anomalie als abnormal klassifiziert. Wie aus den Versuchsergebnissen hervorgeht, liegt der *F1-Score* zwischen 0,67 und 0,82. Während die Anzahl der nicht erkannten abnormalen Bereiche relativ gering ist, ist die Anzahl der Fehlalarme hoch. Dies ist auf die starke Varianz der Zeitreihen, die schwankenden Korrelationen der Signale und die hohe Anzahl der verwendeten Modelle in den GNN zurückzuführen. Im Abschn. 11.9 werden Möglichkeiten zur Steigerung der Leistung analysiert.

Abb. 11.17 zeigt ein Beispiel für die Erkennung einer globalen Anomalie. Der Graph des CPPS (Abb. 11.17a) wurde zum Zeitpunkt der Anomalie erstellt. Abb. 11.17b zeigt Zeitreihendaten verschiedener CPS-Knoten und Abb. 11.17c die der Produktknoten des Testdatensatzes. Das rote Feld in b und c kennzeichnet den manuell klassifizierten abnormalen Bereich. Abb. 11.17 stellt nochmals die Kontextzeitreihe von Abb. 11.17c (oberste Zeitreihe) zur besseren Visualisierung dar. Die roten Markierungen in der Zeitreihe definieren die Überschreitung des Schwellenwertes des Rekonstruktionsfehlers und somit die Erkennung einer Anomalie (Abb. 11.18). Nach der Erkennung können, basierend auf dem zuvor extrahierten Wissen, Informationen zusammen mit der Erkennung dem Bediener zur Verfügung gestellt werden. Dadurch erhält der Anwender zusätzliche Informationen, um die Anomalie analysieren und richtig darauf reagieren zu können.

Abhängigkeits- und Interaktionserkennung: Durch die separate Rekonstruktion der Graphenteile können Anomalien innerhalb der Knoten und Kante erkannt werden. Dabei können abhängig vom gewählten Aktualisierungsprozess und den Adjazenzmatrizen Anomalien innerhalb eines Graphenteiles andere Teile beeinflussen. Bei einer korrekten Modellierung des Graphen und Trainings des GNN können so auch Anomalien in den Interaktionen des Systems identifiziert werden. Ein Beispiel dafür ist in Abb. 11.19 dargestellt. Dabei wurde eine starke Anomalie im Förderband verursacht, während gleichzeitig eine Interaktion zwischen Förderband und Logistikroboter stattgefunden hatte. Innerhalb des Graphen ist diese Interaktion als Kante zwischen den beiden CPS-

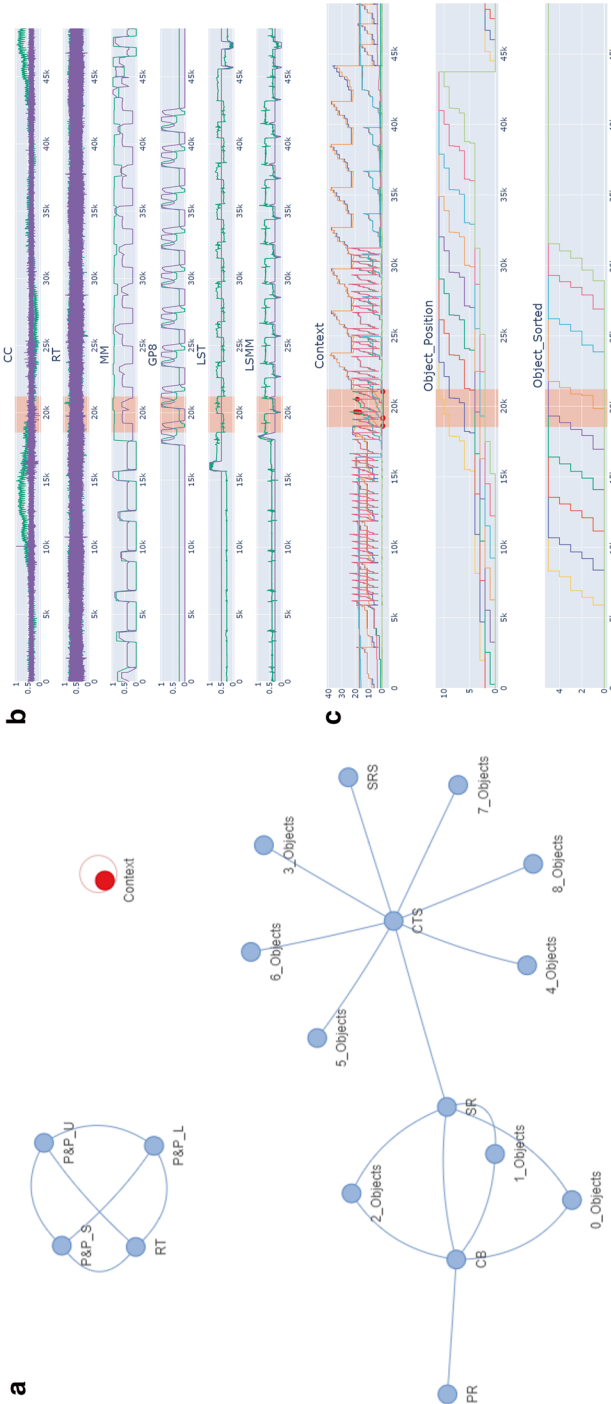


Abb. 11.17 Beispiel der Anomaliedetektion: **a)** Graphdarstellung, **b)** Zeitreihen verschiedener CPS, **c)** Zeitreihen Kontextknoten, Position der Produkte, Anzahl sortierter Objekte in jedem Produkt

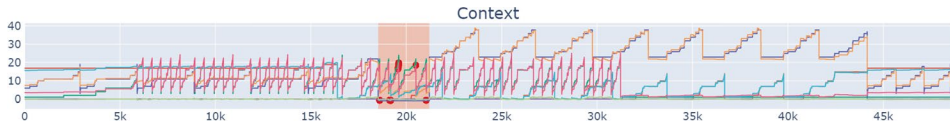


Abb. 11.18 Zeitreihe Kontextknoten; rote Markierungen identifizieren erkannte Anomalien

Knoten modelliert. Dementsprechend beeinflusst die Anomalie nicht nur das Förderband, sondern gleichzeitig auch die Kante und den Logistikroboter.

Auch Anomalien in den Interaktionen zwischen den CPS-Knoten und Produktknoten können erkannt werden. Tritt beispielsweise eine starke Anomalie im Förderband auf, werden die Container, welche sich zur Zeit der Anomalie auf dem Band befinden, auch als abnormal identifiziert. Dies ermöglicht es dem Bediener, diese Container und deren Inhalt später auf mögliche Schäden zu überprüfen. Aufgrund des gewählten Aktualisierungsprozesses des Kontextknotens wird dieser nur dann beeinflusst, wenn mehrere Anomalien gleichzeitig innerhalb des Graphen auftreten und somit ein ganzer Sektor des CPPS beeinflusst ist. Ein Beispiel dafür ist die Abweichung aller CPS eines Sektors innerhalb der Produktion.

11.8 Empfehlungen

Ausgehend von den Erkenntnissen, die bei der Umsetzung des Prototyps gewonnen wurden, und den Ergebnissen der Evaluierung lassen sich die folgenden Empfehlungen für die Umsetzung einer graphenbasierenden Anomalieerkennung geben.

- **Anomaliedetektion:** Externes Wissen bietet die Möglichkeit, bestehende oder neue AD zu verbessern. Eine umfassende Anomaliedetektion ist nur mithilfe von externem Wissen möglich. Daher sollte das Einbeziehen von externem Wissen bei der Entwicklung einer AD berücksichtigt werden.
- **Eigenschaften des Systems:** Bei der Integration müssen die Eigenschaften des CPPS berücksichtigt werden. Dabei werden beispielsweise besondere Anforderungen an Prozessorleistung und Speicherfähigkeit gestellt.
- **Verfügbarkeit der Daten:** Die Verfügbarkeit der Daten spielt eine wichtige Rolle. Sollten nicht ausreichend oder nur ungenügende externe Daten zur Verfügung stehen, kann eine Integration basierend auf externem Wissen kaum realisiert werden.
- **Industrielle Anforderungen:** Bei der Integration innerhalb eines industriellen Umfeldes müssen besondere Anforderungen wie Echtzeitfähigkeit, Performance und der universelle Einsatz berücksichtigt und gegebenenfalls erfüllt werden. Eine Evaluierung dieser Anforderungen ist vor der Konzeption essenziell.

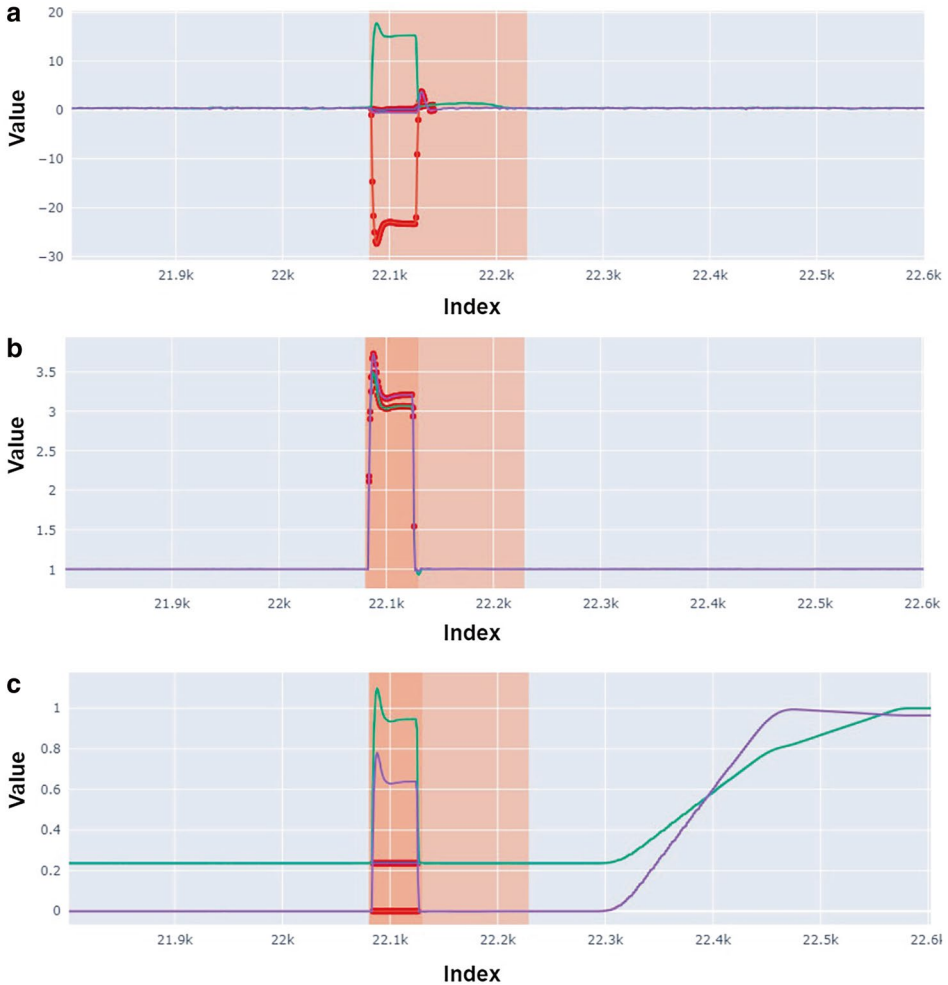


Abb. 11.19 Abhängigkeitsbeispiel: **a)** Anomalie innerhalb des Förderbandes, **b)** beeinflusste Kante zwischen Förderband und Roboter, **c)** Auswirkung der Anomalie innerhalb des Roboters (Goetz, C. und Humm, B. 2023b)

Tab. 11.1 Auswertung der Testergebnisse

Fehlerfall	TP	TN	FP	FN	Precision	Recall	F1-Score
Reibung	68	2577	37	10	0,64	0,87	0,74
Vibration	74	2653	42	12	0,63	0,86	0,73
Kollision	82	2470	22	12	0,78	0,87	0,82
Globale Anomalie	32	1270	10	4	0,88	0,70	0,82
Produktanomalie	26	1470	20	5	0,56	0,83	0,67

- **Verhältnismäßigkeit:** Durch die Einbindung von externem Wissen und die Verwendung komplizierter Techniken steigt auch die Gesamtkomplexität der AD. Für bestimmte Anwendungen sind daher einfache Techniken der AD alleine basierend auf internen Daten ausreichend und daher vorzuziehen.

11.9 Fazit

Durch die Kombination von externem Wissen und Machine Learning kann eine umfassende Anomaliedetektion für komplexe CPPS und deren Prozesse realisiert werden. Dabei muss das externe Wissen zuerst in Kombination mit den zur Verfügung stehenden Daten des Systems gebracht werden. Eine Möglichkeit, diese Kombination zu realisieren, sind Graphen. Anhand von Graphen lassen sich komplexe Interaktionen und Abhängigkeiten sowie strukturelle Eigenschaften mit Status- und Prozessdaten des Systems verbinden. Durch Graph Neural Networks können Techniken aus dem Deep Learning für die Anomaliedetektion auf komplexe Graphstrukturen angewendet werden. Eine Realisierungsmöglichkeit wurde in diesem Kapitel vorgestellt. Auch wenn der erreichte F1-Score noch geringer ist als die erzielten Werte unseres vorherigen Konzeptes (Goetz und Humm 2023a, b, c; Goetz, C. und Humm, B. G. 2023a), welches nur die Daten der CPS separat betrachtet hatte, lassen sich mit dem GNN zusätzliche Anomalien in den Abhängigkeiten und Interaktionen erkennen. Des Weiteren erlaubt das GNN Anwendern Hinweise zur Lokalisierung von Fehlern zu geben.

Wie geht es weiter? Durch die offene Struktur kann das GNN mit einer Vielzahl von Techniken verändert, erweitert und gegebenenfalls verbessert werden. Dies kann beispielsweise der Austausch der verwendeten MLP mit Convolutional Neural Networks oder der Einsatz von Recurrent Neural Networks innerhalb des Aktualisierungsprozesses sein. Auch kann das beschriebene Prinzip der Rekonstruktion zu einer Vorhersage des nächsten Zustandes des Graphen geändert werden, was abhängig vom Anwendungsfall zu einer Verbesserung führen kann. Auch kann weiteres externes Wissen über das CPPS in die Modellierung des Graphen eingebunden werden, um so die Anomaliedetektion möglicherweise zu verbessern. Aktuell finden enorme Entwicklungen und Verbesserungen im Bereich von Graph Neural Networks statt, weshalb die Anzahl an erfolgreichen Anwendungen stetig steigt. Für die Anomaliedetektion und vor allem im industriellen Bereich dagegen lassen sich noch kaum Lösungen finden. Das hier beschriebene Konzept bietet einen ersten Ansatz für eine vollständige interaktions- und abhängigkeitsbewusste Anomaliedetektion für CPPS basierend auf Graph Neural Networks.

Literatur

Ali, N., Hussain, M. & Hong, J.-E., 2022. SafeSoCPS: a composite safety analysis approach for system of cyber-physical systems. *Sensors*, Volume 22, p. 4474.

- Battaglia, P. W. et al., 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.
- Cai, H., Zheng, V. W. & Chang, K. C.-C., 2018. A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, Volume 30, pp. 1616–1637.
- Deng, A. & Hooi, B., 2021. Graph Neural Network-Based Anomaly Detection in Multivariate Time Series. *Proceedings of the AAAI Conference on Artificial Intelligence*, May, Volume 35, pp. 4027–4035.
- Derrow-Pinion, A. et al., 2021. *Eta prediction with graph neural networks in google maps*. s.l., s.n., p. 3767–3776.
- Eiteneuer, B., Hranisavljevic, N. & Niggemann, O., 2019. *Dimensionality reduction and anomaly detection for cpps data using autoencoder*. s.l., s.n., p. 1286–1292.
- Ferludin, O. et al., 2022. TF-GNN: graph neural networks in TensorFlow. *arXiv preprint arXiv:2207.03522*.
- Goetz, C. & Humm, B., 2023. Decentralized Real-Time Anomaly Detection in Cyber-Physical Production Systems under Industry Constraints. *Sensors*, Volume 23, p. 4207.
- Goetz, C. & Humm, B. G., 2023a. Unsupervised process anomaly detection under industry constraints in cyber-physical systems using convolutional autoencoder. In: *Computational Intelligence for Engineering and Management Applications: Select Proceedings of CIEMA 2022*. s.l.:Springer, p. 659–674.
- Goetz, C. & Humm, B. G., 2023b. Unsupervised Correlation- and Interaction-Aware Anomaly Detection for Cyber-Physical Production Systems based on Graph Neural Networks. In: Elsevier Procedia Computer Science: Select Proceedings of ISM 2023.
- Jazdi, N., 2014. *Cyber physical systems in the context of Industry 4.0*. s.l., s.n., pp. 1–4.
- Ji, S. et al., 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems*, Volume 33, p. 494–514.
- Marwedel, P., 2021. *Embedded system design: embedded systems foundations of cyber-physical systems, and the internet of things*. s.l.:Springer Nature.
- Ma, X. et al., 2021. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Transactions on Knowledge and Data Engineering*.
- Monostori, L., 2014. Cyber-physical production systems: Roots, expectations and R&D challenges. *Procedia Cirp*, Volume 17, p. 9–13.
- Müller, T., Jazdi, N., Schmidt, J.-P. & Weyrich, M., 2021. Cyber-physical production systems: enhancement with a self-organized reconfiguration management. *Procedia CIRP*, Volume 99, p. 549–554.
- Rajkumar, R., Lee, I., Sha, L. & Stankovic, J., 2010. *Cyber-physical systems: The next computing revolution*. s.l., s.n., pp. 731–736.
- Scarselli, F. et al., 2008. The graph neural network model. *IEEE transactions on neural networks*, Volume 20, p. 61–80.
- van Harmelen & ten Teije, 2019 “A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems”, Frank van Harmelen, Annette ten Teije, *Journal of Web Engineering*, Vol. 18 1–3, 97–124. <https://doi.org/10.13052/jwe1540-9589.18133>.
- Wang, Q., Mao, Z., Wang, B. & Guo, L., 2017. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Transactions on Knowledge and Data Engineering*, Volume 29, pp. 2724–2743.
- Wu, Z. et al., 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, Volume 32, p. 4–24.
- Zhang, Z., Cui, P. & Zhu, W., 2020. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, Volume 34, p. 249–270.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Vorhersage von Sportergebnissen mittels probabilistischer Programmierung

12

Christian Herta  und Thomas Hoppe 

12.1 Einleitung

Kernaussagen

1. Bayes'sche Modellierung/Statistik erlaubt den natürlichen Umgang mit Unsicherheit.
2. Domänenspezifisches Wissen in Form von Abhängigkeiten zwischen Zufallsvariablen schränkt den Suchraum von probabilistischen grafischen Modellen ein.
3. Gerichtete grafische Modelle, die mit Bayes'schen Verfahren gelernt werden, können als probabilistische Programme implementiert werden.
4. Der Satz von Bayes ermöglicht es, die Modellvariablen und damit die Modelle inkrementell an neu erhobene Daten anzupassen.
5. Die Modellierung mit probabilistischen grafischen Modellen ist ein inhärent hybrides KI-Verfahren.
6. Probabilistische grafische Modelle ermöglichen die Ableitung nichtquantifizierbarer Größen aus Daten.
7. Der Ausgang von Mannschaftsspielen basiert auf nichtquantifizierbaren Größen, wie den Fähigkeiten der Spielern, der Zusammensetzung, den Spielstärken und dem Zusammenspiel der Mannschaften, den Entscheidungen der Schiedsrichter als auch einem Zufallsfaktor, landläufig als „Glück“ resp. „Pech“, bezeichnet.

C. Herta (✉) · T. Hoppe
Hochschule für Technik und Wirtschaft (HTW) Berlin, Berlin, Deutschland
E-Mail: christian.herta@htw-berlin.de

T. Hoppe
E-Mail: thomas.hoppe@htw-berlin.de

Prognosesysteme können die Entscheidungsfindung in vielen Bereichen von Wirtschaft und Technik unterstützen. Allerdings können Ereignisse in der Regel nicht exakt vorhergesagt werden. Bestenfalls können Wahrscheinlichkeiten für ihr Eintreten angegeben werden. Das bedeutet, dass ein Modell der Welt in der Lage sein muss, mit Unsicherheiten umzugehen und mit Wahrscheinlichkeiten zu operieren, um Schlussfolgerungen ziehen zu können. Die Bayes'sche Sichtweise der Wahrscheinlichkeitstheorie erlaubt es, Vorwissen wie Domänenwissen auf natürliche Weise mit Daten zu kombinieren, um Wahrscheinlichkeitsaussagen zu erhalten. Darüber hinaus unterstützt sie die kontinuierliche Anpassung von Modellvariablen und -parametern, wenn neue Daten verfügbar werden. Durch die Nutzung von Expertenwissen benötigen diese Verfahren weniger Daten als generische maschinelle Lernmodelle.

Einführungen in die Bayes'sche Sichtweise der Wahrscheinlichkeitstheorie beginnen traditionell ausführlich mit den mathematischen Grundlagen. Diese werden meist anhand einfacher Standardbeispiele erläutert. Die Frage, wie reale, mit Unsicherheiten behaftete Probleme mit Wahrscheinlichkeiten modelliert werden können und wie Domänenexpertise in die Modellierung einfließt, wird dagegen oft nur am Rande behandelt. Im Gegensatz dazu wird in diesem Kapitel der Schwerpunkt auf die Modellierung Bayes'scher Prognosesysteme anhand eines realitätsnahen Beispiels gelegt. Auf die mathematischen Grundlagen wird nur im unbedingt notwendigen Umfang eingegangen.¹

Komplexere Modellierungen können mithilfe von probabilistischen grafischen Modellen – die wir im Folgenden als gerichtete grafische Modelle bezeichnen – realisiert werden. In diesen Netzen werden statistische Abhängigkeiten zwischen verschiedenen Variablen mithilfe von Fachwissen modelliert. In diesem Kapitel wird das Vorgehen bei der Modellierung anhand von Beispielen aus dem Sportbereich zur Prognose von Spielergebnissen aufgezeigt, da dieses Gebiet für die meisten Leser leicht verständlich sein dürfte. Die Implementation eines solchen Modells in Software erfolgt mithilfe einer Programmiersprache der probabilistischen Programmierung und auf der Basis von Statistiken in Tabellenform, Zeitreihen der Mannschaftsentwicklung und Graphen zur Beschreibung von Abhängigkeiten zwischen Zufallsvariablen.

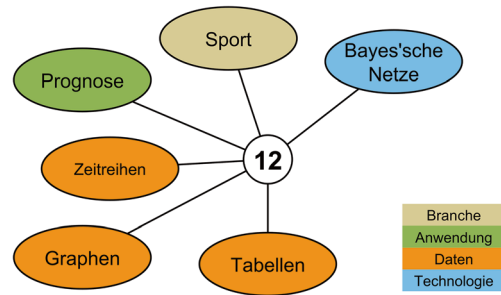
Abb. 12.1 zeigt, wie sich dieses Kapitel in Bezug auf die in diesem Buch verwendeten Kriterien zur Beschreibung hybrider KI-Anwendungen einordnet.

12.1.1 Vorgehen bei der Modellbildung

Probabilistische grafische Modelle und probabilistische Programme erlauben es, basierend auf dem Satz von Bayes, Probleme als Zufallsprozesse zu modellieren, die unterschiedliche Zufallsvariablen kombinieren, um beobachtete Daten zu erklären.

¹Für interessierte Leser und Leserinnen sei an dieser Stelle auf (Koller und Friedmann 2009) und (Barber 2012) verwiesen, um mehr über die mathematischen Grundlagen zu erfahren.

Abb. 12.1 Thematische Einordnung des Kap. 12



In der traditionellen, frequentistischen Statistik werden Wahrscheinlichkeitsverteilungen und deren Parameter als konstant, aber unbekannt – quasi als gottgegeben – betrachtet. Daten, die erhoben oder gemessen werden, werden als zufällig aus diesen Verteilungen gezogen betrachtet. Aussagen mittels statistischer Hypothesentests sind nur bzgl. der „Extremheit“ der Daten in Bezug auf die Verteilungen möglich.² Insbesondere sind in der frequentistischen Sichtweise keine Aussagen über die Wahrscheinlichkeit der angenommenen Verteilungen möglich.

Die Bayes'sche Sichtweise betrachtet Verteilungen und deren Parameter als Zufallsvariablen, die ein Modell für die Daten darstellen. Bayes'sche Methoden verwenden anfängliche Wahrscheinlichkeitsverteilungen und A-priori-Werte (Prior) ihrer Parameter, um die Unsicherheit über Zufallsvariablen zu repräsentieren, und passen die Parameterverteilungen basierend auf den beobachteten Daten an. Basis hierfür ist der Satz von Bayes, der es darüber hinaus erlaubt, die Wahrscheinlichkeit zu ermitteln, wie gut die Verteilungen die Daten beschreiben. Die Verteilungen und deren Parameter werden hierbei nur als Hypothesen über die Daten betrachtet, sodass die gleichen Daten mehrfach genutzt werden können, um die Wahrscheinlichkeiten unterschiedlicher Verteilungshypothesen zu ermitteln. Die Bayes'sche Sichtweise entspricht damit eher einem Modellbildungs- beziehungsweise Modellkonstruktionsprozess, bei dem Verteilungen Hypothesen über die Welt darstellen, deren Variablen an die realen Daten angepasst werden.

Grundlage für die Erfolge des maschinellen Lernens mit modernen Architekturen neuronaler Netze sind riesige Datenmengen, die, wie in Kap. 9 dargestellt, nicht immer verfügbar sind bzw. erst im Laufe der Anwendung langsam gesammelt werden. Einen Ausweg aus diesem Dilemma bieten gerichtete grafische Modelle, die gegenüber vielen anderen Verfahren des maschinellen Lernens eine wesentliche zusätzliche Eigenschaft aufweisen: die explizite Darstellung von Abhängigkeiten zwischen Zufallsvariablen. Damit kann domänenspezifisches Wissen genutzt werden, um den Einfluss der Variablen aufeinander einzuschränken. Die Kombination mit dem Bayes'schen Ansatz, alle

²Wurden die Daten einmal für einen Hypothesentest verwendet, gelten sie als „verbrannt“. Für weitere Tests sind neue Daten zu erheben oder Korrekturfaktoren zu verwenden.

Variablen durch Wahrscheinlichkeitsverteilungen zu beschreiben, erlaubt eine weitere Reduktion der Datenmenge, da weiteres Vorwissen die A-priori-Verteilungen bestimmt. Diese Bayes'schen Modelle sind dann mit einer entsprechenden Unsicherheit behaftet. Wie unsicher das Modell ist, d. h., wie zuverlässig die Vorhersage ist, kann direkt aus den Verteilungen abgelesen werden. Durch zusätzliche Daten oder weiteres Domänenwissen kann die Unsicherheit reduziert und damit die Vorhersagegüte der Modelle erhöht werden. Die Modellierung für eine Anwendung mit gerichteten grafischen Modellen ist somit ein inhärent hybrider KI-Prozess.

12.1.2 Anwendungsfall: Vorhersage von Fußballergebnissen

Wie in anderen Bereichen des Sports hat die Informatik auch in Fußballvereinen, Fußballstadien und Wettbüros Einzug gehalten, sei es, um den Wert und die Stärke einer Mannschaft durch gezielteres Training zu steigern, die Bewegungen und Aktionen einzelner Spieler zu erfassen, ganze Spiele statistisch auszuwerten oder Wettquoten zu ermitteln.

Die ökonomische Bedeutung des Profifußballs wurde von McKinsey im Jahr 2020 in der Studie „Unternehmen Bundesliga“ untersucht (McKinsey und Company 2020). Danach liegt die Gesamtwertschöpfung bei über 11 Mrd. EUR und es werden ca. 127.000 Menschen auf Vollzeitbasis hierdurch beschäftigt. Hierbei kommt dem Kaderwertmanagement eine große Bedeutung zu. Einerseits, da der Einkauf starker Spieler nicht nur die Gewinnchancen und damit den Wert des Kaders erhöht, sondern andererseits auch, da der Spielertransfer durch Verleih und Verkauf eine wesentliche Einnahmequelle für die Vereine darstellt. Einfach formuliert, durch die Optimierung der Spielstärke einer Mannschaft kann der Kaderwert optimiert werden. Da der Kaderwert mit dem sportlichen Erfolg korreliert (Abb. 12.10 in McKinsey und Company 2020), kann er stellvertretend für die – nicht direkt messbare – Spielstärke verwendet werden. Sofern eine Mannschaft nicht nur unter monetären Entscheidungen gemanagt wird, kann die Spielstärke der Mannschaft für kaderwertbasierte Managemententscheidungen herangezogen werden.

Laut dem Deutschen Sportwettenverband betrug das Umsatzvolumen des deutschen Sportwettenmarkts im Jahr 2021 9,4 Mrd. EUR (Deutscher Sportwettenverband 2021). Hiervon nehmen lt. Franke (2021) Wetten auf Fussballergebnisse mit 55 % den größten Marktanteil ein. „Auf Platz 2, wenngleich mit großem Abstand, folgt Tennis“ (Franke 2021) mit 17 %. Die Vorhersage von Spielergebnissen ist daher auch für Wettbüros unumgänglich, um Wettquoten festzulegen und damit ihre eigenen Gewinne zu optimieren.

Im Rahmen des Sportjournalismus wird der Ausgang von Fußballspielen – zumeist von Weltmeisterschaftsspielen – oft durch tierische WM-Orakel³ vorhergesagt. Auf alle

³[Tierische WM-Orakel und ihre Erfolge | ZooRoyal Magazin](#) (letzter Zugriff 8.6.2023).

möglichen Begegnungen ausgelegt, wären solche Tierorakel kaum noch zu managen. Ein KI-Orakel für Wettende stellt dagegen eine automatisierbare, kommerziell nutzbare Entscheidungshilfe dar, mit der die Wettenden ihre eigenen Gewinnchancen u. U. erhöhen könnten.

Nicht nur bedingt durch diese Einsatzgebiete stellt die Vorhersage von Fußballergebnissen ein interessantes Anwendungsgebiet für hybride KI-Verfahren dar, sondern auch durch die Verfügbarkeit und Menge an Daten, die über die Spiele erhoben werden, und durch die allgemein leicht verständlichen Abhängigkeiten, die den Ausgang der Spiele beeinflussen.

Neben diesen praktischen Anwendungsbereichen stellt die Vorhersage von Sportergebnissen ein interessantes Beispiel für die Lehre dar, um Studierenden die Grundlagen der Modellierung von probabilistischen Lösungen beizubringen, da ihnen der Sportkontext aus ihrer Erfahrung geläufig sein sollte.

Mannschaftsspiele sind hier von besonderem Interesse, da ihre Ausgänge sowohl durch die Fähigkeiten der Spieler, die Zusammensetzung und das Zusammenspiel der Mannschaft, die Entscheidungen der Schiedsrichter als auch durch einen Zufallsfaktor, landläufig als „Glück“ resp. „Pech“ bezeichnet, bestimmt werden. Diese unterschiedlichen Faktoren bilden einerseits eine Herausforderung für die Vorhersage von Mannschaftsspielen, andererseits ermöglichen sie es, unterschiedliche Formen von Domänenwissen zu berücksichtigen.

Darüber hinaus bilden Mannschaftsspiele ein gut abgrenzbares System, welches kaum Änderungen und nur wenigen externen Einflussfaktoren unterliegt, dass durch die periodische Wiederholung unter annähernd gleichen Bedingungen große Mengen an Daten produziert werden (Aoki et al. 2017).

Aoki et al. (2017) analysierten unterschiedliche Mannschaftsspiele, wie Basketball, Fußball, Handball und Volleyball, um die Faktoren „skill“ (Können) und „luck“ (Glück) als relative Gewichtungen in einem probabilistisch grafischen Modell (PGM) abzubilden. Sie zeigten, dass die Glückskomponente einen wesentlichen Einfluss auf den Ausgang von Spielen hat und dass komplexe, featurebasierte Modelle einfache Vorhersagemodelle nur schwer übertreffen können. Ihr Modell basiert auf der Entflechtung von Skill und Luck. Hierzu verwenden sie einen „skill coefficient“ für Mannschaften. Dieser Koeffizient repräsentiert die Abweichung des Ligaergebnisses einer Mannschaft am Ende einer Spielsaison von einem Ligaergebnis, welches allein vom Zufall abhängen würde.⁴

Einen Überblick über weitere wissenschaftliche Arbeiten zur Vorhersage von Spielergebnissen unterschiedlicher Sportarten geben Wunderlich und Memmert (2020).

⁴Dieser Skill-Koeffizient kann auch negative Werte annehmen. Dies erscheint etwas verwunderlich, kann aber lt. Aoki et al. (2017) durch einen – nicht näher erläuterten – „Versatzmechanismus während der Saison“ oder auf Absprachen zwischen den Teams erklärt werden.

12.1.3 Übersicht über das Vorsageverfahren von Sportergebnissen

Mit Boxology haben van Harmelen und ten Teije (2019) und Witschel et al. (2021), wie im Kapitel Einleitung dargestellt, einen Ansatz zur vereinheitlichten Beschreibung von maschinellen Lernverfahren und wissensbasierten Systemen vorgestellt. Gerichtete probabilistische grafische Modelle können als hybrides KI-Verfahren angesehen werden, da sie die symbolische Repräsentation von Abhängigkeiten mit dem Erlernen von Wahrscheinlichkeitsverteilungen kombinieren und es ermöglichen, unscharfe Inferenzen zu ziehen. Das Training, die Vorhersage von Spielergebnissen und die Evaluation grafischer Modelle können als Boxology-Grafik wie in Abb. 12.2 dargestellt werden.

In das Training eines probabilistischen Modells zur Vorhersage von Spielergebnissen fließen sowohl die symbolisch durch einen Graphen beschreibbaren Abhängigkeiten zwischen Zufallsvariablen, Annahmen über A-priori-Verteilungen, Statistiken über historische Spielausgänge (hier Fußball) und Ergebnisse der Spiele der aktuellen Saison ein. Im Rahmen der Anwendung des Modells bzw. zu dessen Weiterentwicklung wird das probabilistische Modell genutzt, um Vorhersagen aktueller Spiele zu treffen (z. B. welche Mannschaft gewinnen wird, wie groß die Tordifferenz ist oder wie das Ergebnis lautet) bzw. die Güte des Modells anhand der aktuellen Spiele zu beurteilen.

12.2 Einführendes Beispiel: Siegvorhersage im Tennis

Die Entwicklung eines Ansatzes zur Vorhersage von Spielergebnissen bildet, neben der praktischen Anwendung, einen hinreichend komplexen Kontext zur Vermittlung der Vorgehensweise zur Modellierung gerichteter grafischer Modelle.

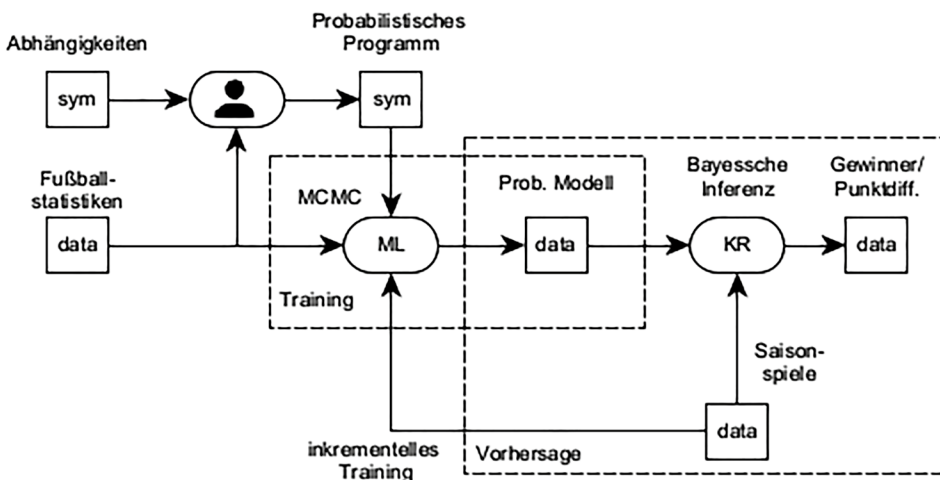
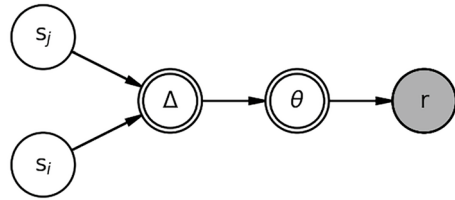


Abb. 12.2 Boxology-Modell des hybriden KI-Systems

Abb. 12.3 Einfaches Modell zur Vorhersage von Spielergebnissen r auf der Basis der Spielstärken s zweier Spieler⁵



Zuerst soll ein Spiel modelliert werden, bei dem es nur einen Sieger und einen Verlierer gibt, wie z. B. beim Tennis. Als Ergebnis wird nur betrachtet, wer von beiden Spielern gewonnen hat. In einer einfachen Modellierung kann jedem Spieler eine Spielstärke s_i resp. s_j zugewiesen werden. Das Ergebnis eines Spiels ist in der Modellierung von den Spielstärken bzw. der Differenz Δ beider Spielstärken abhängig. Mit dieser Differenz $\Delta = s_i - s_j$ kann die Wahrscheinlichkeit θ , dass Spieler i gewinnt, bestimmt werden. Mit dieser Wahrscheinlichkeit kann das Spielergebnis r vorhergesagt werden. Die Berechnung von θ kann mit der logistischen Funktion durchgeführt werden:

$$\theta = \frac{1}{1 + e^{-\Delta}}$$

Diese logistische Funktion bildet dabei die Differenz Δ auf einen Wert zwischen 0 und 1 ab, sodass die Interpretation als Wahrscheinlichkeit möglich wird.

Spielen die Sportler mehrfach gegeneinander, kann das Ergebnis unterschiedlich ausfallen. Das Ergebnis ist eine Zufallsvariable. Die Spielstärke der beiden Sportler kann dabei nicht direkt gemessen bzw. bestimmt werden. Die Spielstärke ist eine sogenannte verborgene Variable (auch als latente Variable bezeichnet), von der das Spielergebnis abhängt. Grafisch kann man diese Abhängigkeiten zwischen den Variablen als gerichtetes grafisches Modell darstellen:

Für die Ergebnisse r (Result) liegen Daten vor, d. h. viele historische Spielergebnisse für eine Menge von vielen Spielern. Hier verwenden wir die Ergebnisse der ATP-Turniere des Jahres 2012. Das Modell in Abb. 12.3 ist also nur ein Ausschnitt aus einem umfassenderen Graphen, der alle Spieler und Spiele berücksichtigt.

Die Spielstärken der Spieler werden in der Bayes'schen Sichtweise als Wahrscheinlichkeitsverteilungen modelliert (Abb. 12.4). Daher werden Δ und θ (deterministisch) auch zu Wahrscheinlichkeitsverteilungen (Abb. 12.5). Die Breite (Varianz) der Verteilung von θ entspricht der Unsicherheit, mit der man θ aus den Daten schätzen kann. Je mehr Daten man hat, desto schmaler wird die Verteilung.

⁵Einfache Knoten stellen Zufallsvariablen dar, graue Knoten beobachtbare Variablen und doppelt umrandete Knoten stellen deterministische Variablen dar, die aus Zufallsvariablen, Konstanten oder anderen deterministischen Variablen direkt deterministisch berechnet werden können.

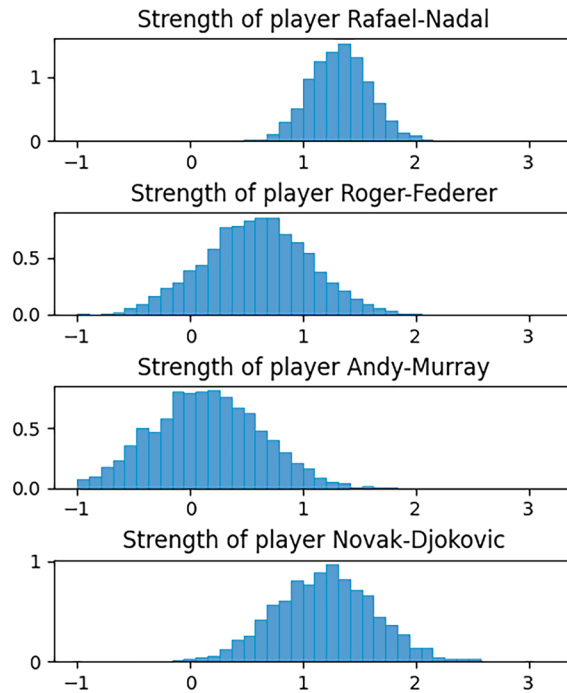


Abb. 12.4 Wahrscheinlichkeitsverteilungen der Spielstärken unterschiedlicher Tennisspieler (berechnet mit PyMC⁶)

Aus diesen Verteilungen kann bereits abgeleitet werden, dass Rafael Nadal und Novak Djokovic im Jahr 2012 stärkere Spieler waren als Roger Federer und Andy Murray. Die kleinere Breite der Spielstärkenverteilung von Rafael Nadal deutet darauf hin, dass er eine konstantere Spielstärke hatte als die anderen drei Spieler.

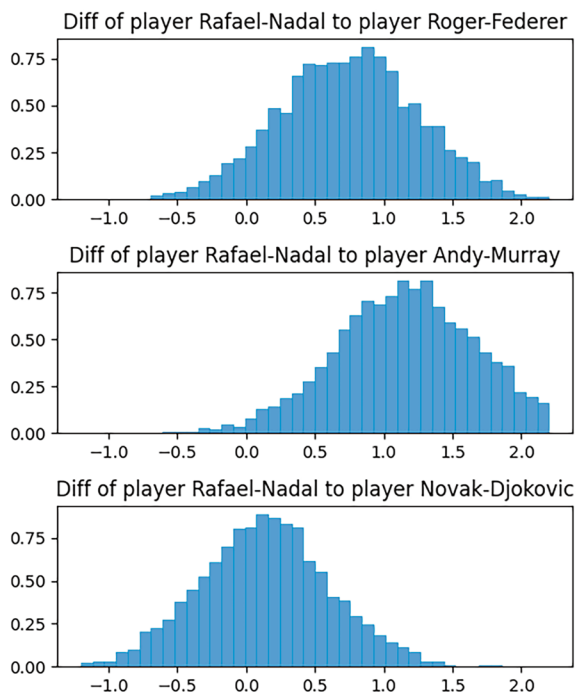
Aus den Wahrscheinlichkeiten der Spielstärke können darüber hinaus Prognosen generiert werden. Zwar lassen sich die konkreten Ergebnisse eines Spiels nicht vorhergesagen, es kann aber vorhergesagt werden, welcher der beiden Spieler gewinnt und mit welcher Wahrscheinlichkeit. Dazu wird zuerst aus den Spielstärkenverteilungen jeweils eine Verteilung für θ ermittelt, wie ein Spieler in unterschiedlichen Partien gegen andere Gegner abschneidet:

Um einen Wahrscheinlichkeitswert, dass Spieler 1 gewinnt, zu ermitteln, wird hieraus ein mit der Verteilung gewichteter Mittelwert von θ berechnet. Dies entspricht der Wahrscheinlichkeit, dass Spieler i gegen Spieler j gewinnt.

Hieraus kann einerseits die durchschnittliche Wahrscheinlichkeit, dass Rafael Nadal gegen Roger Federer gewinnt, mit 0,67 ermittelt werden, und dass er auch gegen Andy

⁶<https://www.pymc.io> (letzter Zugriff 8.6.2023).

Abb. 12.5 Verteilungen der Wahrscheinlichkeit θ , dass Novak Djokovic gegen andere Spieler gewinnt

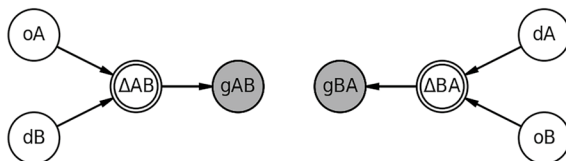


Murray gewinnt mit 0,75. Gegen Novak Djokovic jedoch gleichen seine Chancen einem Münzwurf mit einer Wahrscheinlichkeit von 0,53. Andererseits kann aus diesen Verteilungen auch gleichzeitig ein Credibility-Intervall⁷ über die Sicherheit der Aussagen abgeleitet werden.

Zusammengefasst haben wir in diesem Abschnitt gesehen, dass schwer objektivierbare abstrakte Größen – wie die Spielstärke von Spielern –, die den Ausgang eines Spiels als „Differenz ihrer Spielstärken“ bestimmen, als latente Zufallsvariable modelliert werden können. Der Unterschied in den Spielstärken ist eine Zufallsvariable, von der der Spielausgang abhängt resp. die Wahrscheinlichkeit, dass ein bestimmter Spieler gewinnt. Historische Daten über die Ausgänge einer Vielzahl von Spielen erlauben es, die bedingten Wahrscheinlichkeiten der latenten Spielstärken in Form von Wahrscheinlichkeitsverteilungen zu bestimmen.

⁷Credibility Intervalle sind das bayes'sche Analogon zu den Konfidenzintervallen der traditionellen frequentistischen Statistik.

Abb. 12.6 Modellierung der Angriffs- (o für offensiv) und der Abwehrstärke (d für defensiv) zweier Mannschaften



12.3 Prognose von Fußballbundesligaspielen

12.3.1 Grundlegende unabhängige Modelle

Im Vergleich zur Vorhersage von Tennisspielen, bei denen nur Sieg oder Niederlage möglich sind, stellt die Vorhersage von Fußballergebnissen die nächste Schwierigkeitsstufe dar. Neben Sieg und Niederlage kann es auch zu einem Unentschieden kommen. Das obige Modell müsste entsprechend erweitert werden. Außerdem hat der Spielort, ob es sich um ein Heim- oder Auswärtsspiel handelt, einen Einfluss auf das Spielergebnis. Um die Vorhersagen noch detaillierter zu machen, kann die Anzahl der von den beiden Mannschaften geschossenen Tore vorhergesagt werden (Abb. 12.6).

Hierzu wird eine geeignete Wahrscheinlichkeitsverteilung benötigt, um die geschossenen Tore zu modellieren. Die Poisson-Verteilung ist eine diskrete Wahrscheinlichkeitsverteilung, die angewandt wird, um die Wahrscheinlichkeit der Häufigkeit von Ereignissen in einem festen Zeitintervall zu beschreiben. Für unser Beispiel bedeutet dies: Ein Fußballspiel dauert knapp über 90 min und die Ereignisse sind die Tore, die die beiden Mannschaften schießen. Die Poisson-Verteilung hat einen Parameter $\lambda > 0$, der angibt, wie viele Ereignisse (Tore) in einem Zeitintervall im Durchschnitt zu erwarten sind. Darüber hinaus können die weiteren Annahmen der Poisson-Verteilung als erfüllt betrachtet werden.⁸

Die Stärke von Mannschaften kann beispielsweise über zwei latente Variablen modelliert werden: eine für die Abwehr- und die andere für die Angriffsstärke. Die Differenz zwischen der Angriffsstärke des Teams A und der Abwehr des Teams B ergibt den Parameter $\lambda = \Delta AB = oA - dB$. Diese Differenz ist der Erwartungswert der Poisson-Verteilung, wie viele Tore im Durchschnitt Team A gegen Team B erzielt hat. Die Anzahl der Tore, die Team B schießt, wird analog modelliert.⁹ gAB sind hierbei die Tore, die Mannschaft A gegen Mannschaft B erzielt. Die Verteilungen der Angriffs- und Abwehrstärke unterschiedlicher Mannschaften werden in Abb. 12.7 dargestellt.

⁸Die Poisson-Verteilung wird daher üblicherweise für die Tor-Modellierung in der Literatur verwendet, z. B. bei (Karlis und Ntzoufras 2003) und (Baio und Blangiardo 2010).

⁹Da die Poisson-Verteilung nur für positive Lambda definiert ist, muss sie bei einer negativen Differenz auf null gesetzt werden. Alternativ könnte die Differenz über eine Exponentialfunktion in den positiven Bereich gebracht werden.

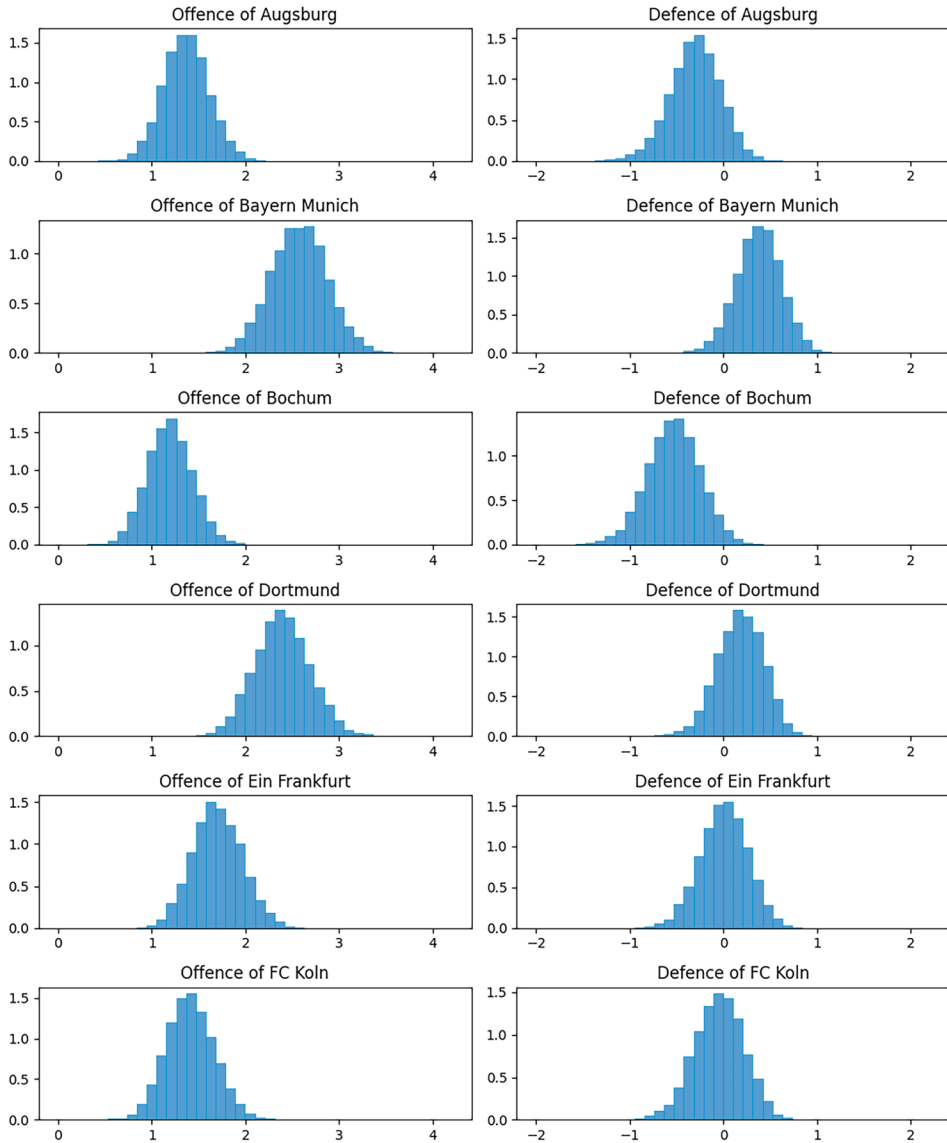


Abb. 12.7 Angriffs- und Verteidigungsspielstärken (s) einiger Bundesligamannschaften

Aus den Angriffs- und Abwehrstärken können Wahrscheinlichkeitsvorhersagen für die geschossenen Tore gemacht werden, wenn zwei Mannschaften gegeneinander spielen. Aus den Verteilungen der Angriffs- und Abwehrstärken ergeben sich zwei Verteilungen für λ : $\lambda = \Delta AB$ und $\lambda = \Delta BA$. Aus diesen können Torverteilungen berechnet werden.

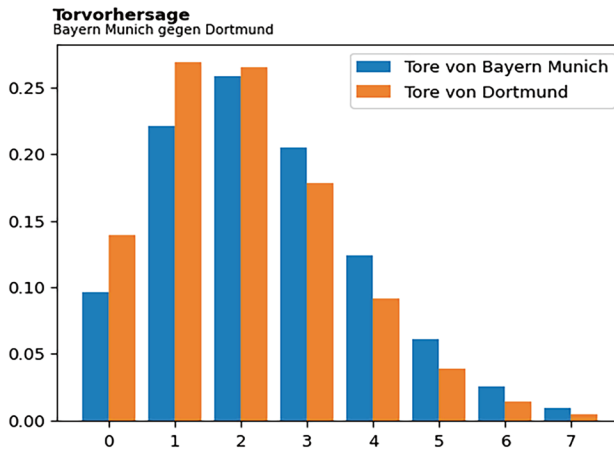


Abb. 12.8 Aus den Spielstärken der Mannschaften ermittelte, erwartete Torverteilungen für ein Spiel Bayern München gegen Borussia Dortmund

Abb. 12.8 zeigt exemplarisch eine Torverteilung, die mit den berechneten Spielstärken der Mannschaften Bayern München und Borussia Dortmund erwartet werden kann, wenn sie gegeneinander spielen. Beispielsweise liegt die Wahrscheinlichkeit, dass Dortmund kein Tor schießt bei ca. 14 % und dass Dortmund genau ein Tor schießt bei ca. 27 % etc.

12.3.2 Zusammengeführtes Modell

Bei obigem Modell wurden die geschossenen Tore beider Mannschaften vereinfachender Weise als statistisch unabhängig betrachtet. Es konnte jedoch empirisch gezeigt werden, dass die beiden Poisson-Variablen Heim- und Auswärtstore korreliert sind (Karlis und Ntzoufras 2003). Hierfür wurden in der Literatur verschiedene Lösungen entwickelt. Karlis und Ntzoufras (2003) verwendeten eine bivariate Poisson-Verteilung zur Modellierung, die zu einer komplizierten Formulierung mit Kovarianzen führt. Alternativ dazu wurde von Baio und Blangiardo (2010) ein hierarchisches Bayes'sches Modell vorgeschlagen, das auch die Poisson-Variablen durch eine Hierarchie verknüpft. Dieser generische Ansatz führt aber zum Problem des Overshrinkage (Baio und Blangiardo 2010; Abb. 12.9).

Hier wird dagegen eine andere Modellierung vorgeschlagen, die die spezielle Struktur der Anwendung berücksichtigt und somit zusätzlich eine bessere Interpretierbarkeit ermöglicht:

Nimmt man an, dass je nach Aufstellung und Trainervorgabe eine Mannschaft mal defensiver oder offensiver ausgerichtet ist, kann man für die Teams eine Spielstärke – ana-

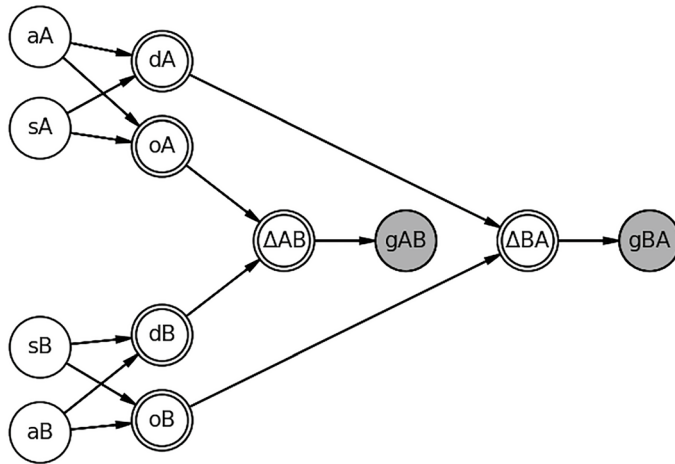


Abb. 12.9 Um Spielstärken (s) und Abweichungen (a) erweiterte Modellierung

log zum Tennis – modellieren und die Offensiv- und Defensivstärke als Abweichung von der Spielstärke berechnen:

Offensivstärke = Spielstärke + Abweichung

Defensivstärke = Spielstärke – Abweichung

beispielsweise für Team A:

$$oA = sA + aA$$

$$dA = sA - aA$$

Für ein Spiel und Team werden dieselbe Spielstärke und Abweichung zur Berechnung der Offensiv- und Defensivstärke verwendet. Das heißt, für ein Spiel wächst bei größerer Abweichung die Offensivstärke genauso, wie die Defensivstärke für das Team fällt. Aus den Offensiv- und Defensivstärken werden die Torverteilungen dann genauso wie beim ersten Modell bestimmt.

Diese alternative Modellierung hat genauso viele nichtdeterministische Variablen, wie das Modell mit den unabhängigen Offensiv- und Defensivstärken.

12.3.3 Erweiterungen des Modells

Das Modell aus Abb. 12.8 kann erweitert werden, um z. B. einen Heimvorteil abzubilden. Darunter versteht man, dass Mannschaften oft zu Hause erfolgreicher sind als auswärts. Im Modell kann man dies berücksichtigen, indem man z. B. zur Spielstärke der Heimmannschaft eine Heimvorteilvariable h addiert (Abb. 12.10):

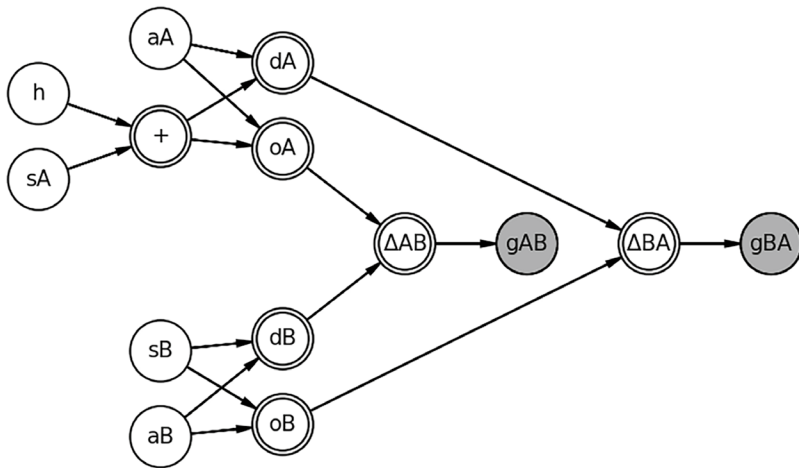


Abb. 12.10 Um Heimvorteil (h) erweiterte Modellierung

Den Heimvorteil könnte man pro Team modellieren. Um jedoch die Anzahl der freien Variablen klein zu halten, wird der Heimvorteil global modelliert, d. h., wir gehen davon aus, dass dieser für alle Teams gleich ist.

Außerdem kann man noch zusätzlich den Gewinner bzw. ein Unentschieden direkt vorhersagen, wie bei TOTO-Sportwetten. Dabei wird die Wahrscheinlichkeit für die drei möglichen Ausgänge (Heimsieg = 1, Unentschieden = 0 oder Auswärtssieg = 2) als multinomiale Zufallsvariable modelliert.

Die Wahrscheinlichkeit für Ausgang i lässt sich mithilfe von Softmax für das Dreiklassenproblem berechnen (siehe z. B. Barber 2012):

$$P(r_i) = \frac{e^{w_i \Delta_{AB}}}{\sum_j e^{w_j \Delta_{AB}}}$$

mit der Differenz der Spielstärken Δ_{AB} und den gelernten Gewichten w_i bzw. w_j . Der Index i bzw. j kann die drei Ausgangswerte 0, 1 und 2 annehmen. Somit ergibt sich das Modell in Abb. 12.11 und die entsprechenden Ergebnisse in 12.12:

Der durchschnittliche globale Heimvorteil (entsprechend Abb. 12.13) liegt bei ca. 0,3. Dies entspricht 0,3 Toren, die im Durchschnitt von einer Mannschaft im eigenen Stadion mehr als auswärts geschossen werden.

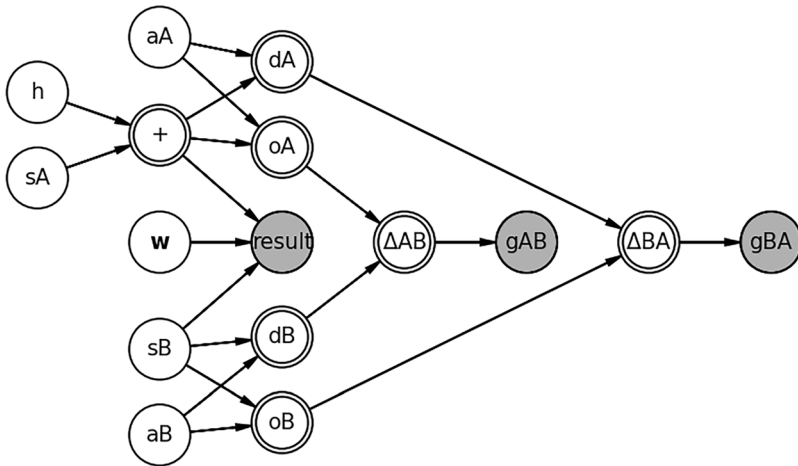


Abb. 12.11 Um die Vorhersage des Spielausgangs (Gewinnen bzw. Unentschieden) erweiterte Modellierung¹⁰

12.3.4 Alternative Modellierungen des Heimvorteils

Der Heimvorteil lässt sich aber auch anders modellieren. Welche Modellierung man wählt, hängt davon ab, an welcher Interpretation man interessiert ist.

Will man erfahren, wie viele Bruchteile von Toren (Bruchteile) die Heimmannschaft mehr erzielt als die Auswärtsmannschaft, dann addiert man den Heimvorteil zu der Spielstärke der Heimmannschaft, wie oben beschrieben. Dies wird als ein additives Modell bezeichnet. Die Spielstärke liegt auf einer Torskala, da der Parameter λ der erwarteten Ereignishäufigkeit (hier Tore) entspricht.

Will man dagegen Aussagen über den relativen Stärkezuwachs treffen, dann kann man ein multiplikatives Modell wählen. Hier wird die Heimspielstärke mit einem Faktor multipliziert. Eine Auswertung der Bundesligasaison 2022/2023 ergibt einen Faktor von rd. 1,2. Dies entspricht einem Spielstärkenzuwachs von ca. 20 %.¹²

¹⁰In der Variablen *result* wird eine nominale Variable (0: unentschieden, 1: Heimmannschaft A gewinnt, 2: Auswärtsmannschaft B gewinnt) beobachtet bzw. vorhergesagt.

¹²In einer Analyse von neun untersuchten aufeinanderfolgenden Spielsaisons der brasilianischen A Fußball Liga haben (Aoki et al., 2017) ermittelt, dass die Wahrscheinlichkeit für den Gewinn eines Auswärtsspiels im Vergleich zu Heimspielen um etwa die Hälfte geringer (0,25 vs. 0,5) ist. Ob dies auch für europäische Ligen oder deutsche Bundesliga bestätigt werden kann, wäre noch zu untersuchen.

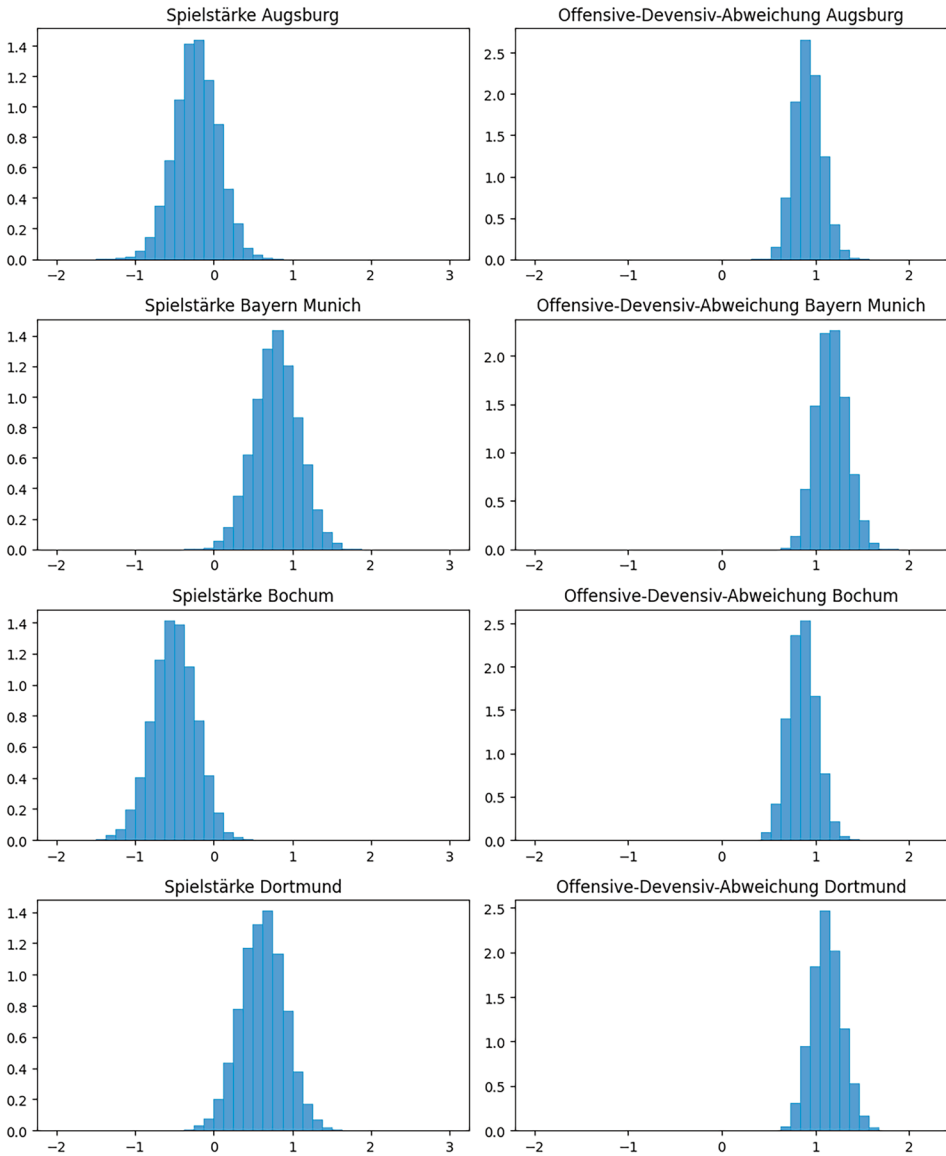
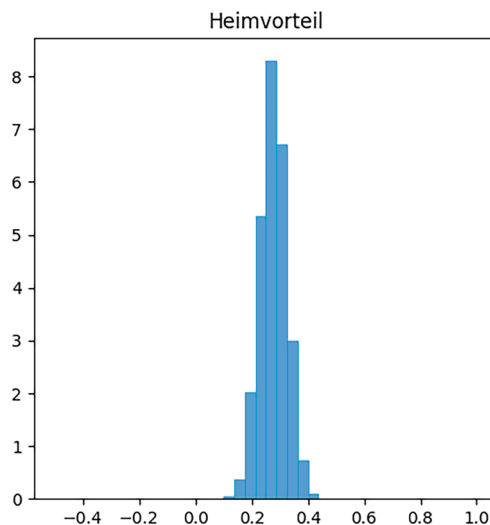


Abb. 12.12 Verteilung der Spielstärken (links) und Offensive-Defensiv-Abweichungen (rechte Spalte) der ersten vier Mannschaften der Bundesligasaison 2022/2023 unter Berücksichtigung aller Spiele

Abb. 12.13 Verteilung des additiven Heimvorteils der Bundesligasaison 2022/2023 unter Berücksichtigung aller Spiele der Saison



12.3.5 Vergleich mit der Bundesligatabelle

Ein Vergleich der durchschnittlichen Spielstärken mit der finalen Platzierung der Vereine in der Bundesliga zeigt, dass bereits bei einer einzigen Simulation der Saison 2022/2023 die Endplatzierung der Vereine in der Bundesliga relativ gut vorausgesagt werden könnte (Abb. 12.14).

Die Abweichungen in beiden Rankings hängen einerseits davon ab, dass bei Punktegleichstand zweier Vereine in der Tabelle auch die Tordifferenzen zur Ermittlung der Platzierung herangezogen werden. Andererseits wurde die Spielstärke über lediglich eine Simulation der Saison ermittelt, sodass der Zufallsfaktor der Simulation noch einen großen Einfluss auf die Spielstärkenverteilung hat. Um diesen Faktor zu minimieren, müssten weitere Simulationen durchgeführt werden.

12.4 Erweiterungen des Modells

Im vorangegangenen Abschnitt wurde die Spielstärke innerhalb einer Saison konstant angenommen. Natürlich kann sich die Spielstärke auch innerhalb einer Saison ändern, z. B. durch unterschiedliche Mannschaftsaufstellungen, durch englische Wochen, durch Mehrfachbelastungen einzelner Spieler oder durch Trainerwechsel.

Abb. 12.14 Vergleich berechneter Spielstärken mit dem Tabellenplatz der Bundesligaspielsaison 2022/2023

	Team	Spielstärke	Tabellenplatz
0	Bayern Munich	0.794984	1
1	Dortmund	0.615320	2
2	RB Leipzig	0.301446	3
3	Freiburg	0.221585	5
4	Union Berlin	0.207174	4
5	Ein Frankfurt	0.122252	7
6	Wolfsburg	0.082214	8
7	Leverkusen	0.080493	6
8	M'gladbach	-0.014473	10
9	Mainz	-0.039895	9
10	FC Koln	-0.079917	11
11	Stuttgart	-0.190931	16
12	Werder Bremen	-0.206765	13
13	Hoffenheim	-0.210545	12
14	Augsburg	-0.222828	15
15	Hertha	-0.423233	18
16	Bochum	-0.506890	14
17	Schalke 04	-0.577294	17

12.4.1 Spielstärke als Zeitreihe

Die Spielstärken von Mannschaften können sich mit der Zeit verändern. Eine Möglichkeit, dies zu modellieren, ist, die Spielstärke als sogenannte Markov-Kette zu betrachten. Eine Markov-Kette beschreibt eine Folge von Zuständen, bei denen ein Folgezustand nur abhängig vom vorangegangenen Zustand ist (siehe z. B. Koller und Friedmann 2009). So kann sich die Spielstärke von Spieltag zu Spieltag (leicht) positiv oder negativ verändern. Dies ergibt ein sogenanntes Hidden-Markov-Modell, da die nicht direkt beobachtete (verborgene) Variable Spielstärke eine Markov-Kette bildet (Abb. 12.15). So kann eine Zunahme bzw. Abnahme der Spielstärke einer Mannschaft im Modell berücksichtigt werden. Grafisch lässt sich die folgendermaßen darstellen:

Für die Vorhersage bedeutet dies, dass die letzten Spiele stärker berücksichtigt werden als weiter zurückliegende Spiele, um die Entwicklung der Mannschaft innerhalb einer Saison bzw. sogar über mehrere Saisons abzubilden. Längere Pausen zwischen Spieltagen (z. B. die Winterpause) können durch eine vorgegebene Anzahl von Kettengliedern ohne Beobachtung (hier: Spielergebnisse) modelliert werden. Dadurch vergrößert sich

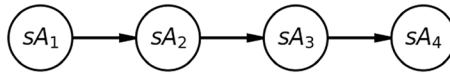


Abb. 12.15 Modell der Markov-Kette der über die Zeit variierenden Spielstärke einer Mannschaft¹¹

die Unsicherheit in den Spielstärkenverteilungen, d. h., die Verteilungen weisen eine größere Varianz auf, sie werden breiter.

12.4.2 Einbezug von Marktwert und Tabellenplatz der Vorsaison

Zu Beginn einer neuen Spielsaison liegen keine aktuellen Spielergebnisse für die einzelnen Mannschaften vor. Um mit nur wenigen Ergebnissen dennoch einigermaßen aussagekräftige Vorhersagen zu machen, kann man weitere Daten einfließen lassen.

Eine Möglichkeit ist es, die A-priori-Spielstärken für die einzelnen Teams aus historischen Daten zu berechnen. So können z. B. der Marktwert (mw) des Kaders und Ergebnisse der Vorsaison wie der Tabellenplatz (tp) in ein gewichtetes lineares Modell zur Berechnung der initialen A-priori-Spielstärke einfließen (Abb. 12.16).

Die Variablen des (Bayes'schen) linearen Modells w_0 , w_1 und w_2 wären für alle Teams gleich. Die starke Korrelation zwischen Kaderwert und erzielten Punkten, auf die die

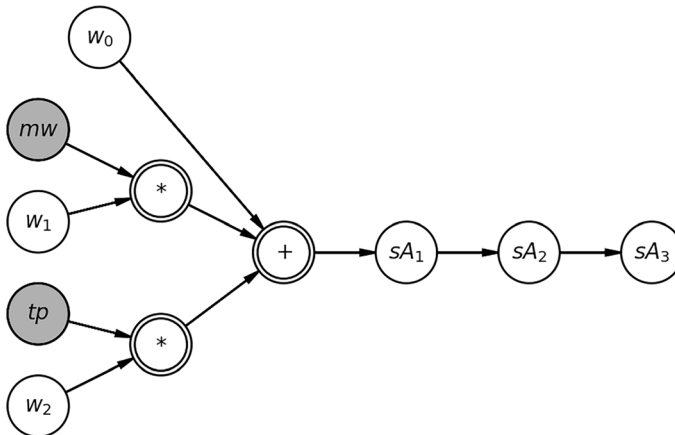


Abb. 12.16 Lineares Modell zur Berechnung der initialen A-priori-Spielstärke der Markov-Kette

¹¹ Der Rest des Modells kann dabei wie oben modelliert werden.

McKinsey-Studie (McKinsey und Company 2020) hinweist, untermauert diese Vorgehensweise.

12.4.3 Einbezug weiterer KPI und Daten aus Spielanalysen

Bis hier haben wir die Modellierung lediglich auf der Ebene der ganzen Mannschaft und des Spielausgangs modelliert. Hierbei wurden sowohl der Charakter der Mannschaft (Offensiv- vs. Defensivmannschaft, Kämpfer, Techniker, Schönspieler), die konkreten Mannschaftsaufstellungen als auch weitere statistische Daten über z. B. die Anzahl der Torschüsse, Ecken, der gespielten Pässe, angekommenen Pässe, Fehlpässe, Passquote, Ballbesitz, Zweikampfquote, Fouls, gelbe und rote Karten einzelner Spieler noch nicht berücksichtigt.

Hier bietet das Feld der Spielanalyse im Fußball (Memmert 2022) weitere Hinweise sowohl auf zusätzliche Abhängigkeiten („Die meisten Fouls werden von der Defensivmannschaft ausgeführt, die nicht im Ballbesitz ist“, Phatak 2022) als auch auf Software zur Positions- und Spielzuganalyse und Quellen für Eventdaten (Caparrós 2022).

Ob diese Ergänzungen und Erweiterungen jedoch zu einer qualitativen Verbesserung der Vorhersageergebnisse führen, ist noch offen. Im schlimmsten Fall können blinde Erweiterungen durch irrelevante oder nur marginal relevante Daten zu einer Erhöhung der Varianz der Vorhersagen führen. Um dies zu vermeiden, sind solche Erweiterungen durch automatisierte Evaluation zu rechtfertigen. Hierfür ist eine entsprechende Lösung jedoch erst noch zu entwickeln.

Aoki et al. (2017) weisen auf eine weitere – u. U. nur für die brasilianischen Ligen relevante – potenzielle Anwendung hin: dass es Fälle gibt, in denen „bedingt durch eine Form von Ausgleichsmechanismus während der Saison oder aufgrund einer Absprache zwischen den Teams“ durch Daten belegbare Irregularitäten identifizierbar gemacht werden können. Hieraus ein Verfahren zur Warnung vor manipulierten Spielen und damit für ein Versagen des Vorhersagesystems zu erarbeiten, ist jedoch noch eine Aufgabe für die Zukunft.

12.5 Technische Umsetzung

Die informationstechnische Umsetzung der Modelle erfolgt mittels probabilistischer Programmierung. Dabei werden die Modelle in entsprechenden Frameworks bzw. domänen-spezifischen Programmiersprachen formuliert. Beispiele hierfür sind TensorFlow Probability, Stan, Pyro und PyMC. Das Trainieren der Modelle erfolgt dabei mittels Blackbox-Solver, die auf verschiedenen Verfahren beruhen, beispielsweise Markov Chain Monte Carlo (MCMC) oder Variational Inference (VI). Beim Trainieren der Modelle werden die (ursprünglichen breiten) Wahrscheinlichkeitsverteilungen der Verteilungsparameter

```

model = pm.Model() // pymc model for winner prediction of tennis games
with model:
    # prior distribution for the skills of each player
    skill = pm.Normal("skill", mu=0, tau=3., shape=(nb_tennis_players,))
    # skill differences of players transformed with logistic function
    skill_div_p = [logistic(skill[player_1]-skill[player_2])
                   for (player_1, player_2) in tennis_games]
    # convert from python list into a (pytensor) vector
    skill_div_p_ = T.stack(skill_div_p)
    # observed prediction, the winner is always player1 (observation 1)
    prediction = pm.Bernoulli("prediction", observed=1, p=skill_div_p_)

```

Abb. 12.17 Einfaches Modell zur Tennissiegevorhersage

```

with model:
    trace = pm.sample(draws=2000, tune=200)

```

Abb. 12.18 Aufruf des Ziehens von 2000 Stichproben mittels MCMC

durch die Daten geschärft, wie oben beschrieben. Der Vorteil von VI gegenüber MCMC sind dabei die Skalierbarkeit auf größere Datenmengen und das schnellere Training.

Zuerst muss das Modell definiert werden. Dies wird im Folgenden beispielhaft an der einfachen Tennissiegevorhersage (siehe oben) für das Python-Framework PyMC gezeigt.¹³ Jedem Spieler wird eine normalverteilte Variable `skill` zugeordnet. Die alle zu einem Vektor `skill` zusammengefasst werden. Die Anzahl der Elemente entspricht dabei der Anzahl der Spieler. Als Ausgangsverteilung wird eine breite Gauß-Verteilung mit Mittelwert (`mu=0`) und Schärfe (`tau=3`) gewählt. In der Variablen `tennis_games` sind die Ergebnisse aller Spiele zusammengefasst. Diese werden gemäß dem Graphen der Abb. 12.3 mit der logistischen Funktion zu Differenzen transformiert. Aus diesen wird die beobachtete Variable `prediction` (Bernoulli-Trials) erzeugt. Der entsprechende Python-Code ist in Abb. 12.17 dargestellt.

Das Modell wird dann einfach mit dem MCMC-Blackbox-Solver trainiert (Abb. 12.18).

Die Ergebnisse (Verteilungen der verborgenen Variablen `skill`) werden in der Variablen `trace` gesammelt und können z. B. als Verteilungen, wie in Abb. 12.5 gezeigt, geplottet werden.

¹³Die programmiertechnische Umsetzungen der hier beschriebenen Modelle sind als jupyter notebooks unter folgender Web-Adresse verfügbar: <https://gitlab.com/deep.TEACHING/probabilistic-sport-result-prediction>

12.6 Zusammenfassung und Ausblick

Unter den Mannschaftssportspielen bildet der Fußball eine hervorgehobene Stellung nicht nur durch das große Interesse in breiten Schichten der Bevölkerung, durch seine Gesamtwertschöpfung und seinen Marktanteil im Bereich der Sportwetten, sondern auch durch die potenzielle Verfügbarkeit umfangreicher Spielstatistiken. Der Ausgang eines Spiels wird von vielen unterschiedlichen Faktoren beeinflusst, die teilweise voneinander abhängig sind und für die Modellierung durch ein gerichtetes grafisches Modell geeignet sind.

Am Beispiel der Vorhersage von Fußballspielen haben wir dargestellt, wie ein gerichtetes grafisches Modell schrittweise durch Berücksichtigung zusätzlicher Abhängigkeiten modelliert werden kann. Hierbei wurden unterschiedliche Modellierungsmuster, wie additive und multiplikative Modelle, Hidden-Markov-Modelle, angesprochen und anhand der Vorhersage des Ausgangs der Saison durch eine einzelne Simulation gezeigt, dass die Spielstärke bereits gut zu den Tabellenplätzen der Mannschaften korreliert, deren Vorhersage jedoch auf vielen Saisonsimulationen beruhen sollte, um Zufallseffekte zu minimieren.

Zusätzliche Modellerweiterungen stellen natürlich die Frage in den Raum, ob sie überhaupt notwendig sind, i. e., ob der damit verbundene Aufwand ihren Nutzen rechtfertigt. Ein automatisiertes Evaluationsverfahren zu implementieren, wäre ein erster Schritt, um sicherzustellen, dass mit Modellveränderungen kein Qualitätsverlust einhergeht. So lassen sich auch Modelle quantitativ vergleichen.

Der Zugriff auf die Spielstatistiken bildet derzeit noch die größte Hürde. Während es relativ leicht ist, auf den Ausgang der Spiele der aktuellen Saison zuzugreifen, erfordert der Zugriff auf Daten über einzelne Spieler, Statistiken über die Angriffs- und Verteidigungssituationen, historische Daten oder auch der Zugriff auf Fußballdaten anderer Länder derzeit noch größeren Rechercheaufwand und erfordert weitere Nutzungserlaubnisse.

Im Idealfall kann ein (hybrides) daten- und domänenwissengetriebenes Entscheidungsunterstützungssystem den Trainern bei der Vorbereitung auf spezielle Gegner helfen. Wenn die für den Spielausgang relevanten Eigenschaften identifiziert werden, können die Taktik und Aufstellung der Mannschaft entsprechend auf den Gegner angepasst werden, um die Wahrscheinlichkeit eines Sieges zu erhöhen.

12.7 Empfehlungen

Zusammenfassend können folgenden Empfehlungen gegeben werden:

- Viele Parameter der realen Welt lassen sich nicht exakt vorhersagen, sodass ihre Modellierung als Zufallsvariablen sinnvoll ist, um Aussagen über ihre Wahrscheinlichkeit, ihre Streuung und Glaubwürdigkeit abzuleiten.
- Gerichtete grafische Modelle als auch die probabilistische Modellierung erlauben es, Hintergrundwissen in die Modellierung einfließen zu lassen, um die Parameter des Modells bereits mit relativ wenigen Daten zu ermitteln.
- Parameter, die nicht mess- oder erhebbar sind, können als latente Variablen modelliert werden, sofern sie helfen, beobachtbare Daten zu erklären bzw. abzuleiten.
- Modellierungen inkrementell durchzuführen ist dann sinnvoll, wenn viele Zusammenhänge bereits bekannt sind und viele unterschiedliche Daten über die Zusammenhänge bereits erhoben wurden.
- Hierbei ist jedoch darauf zu achten, dass Modellerweiterungen auch einen Nutzen mit sich bringen; ihre Verbesserungen sind daher durch eine kontinuierliche Evaluation nachzuweisen.

12.8 Fazit

Im Vergleich zu Kap. 9 kann durch viele historische Daten und das empirische Wissen über Parameterabhängigkeiten im Anwendungsgebiet „Sport“ eine andere Vorgehensweise bei der Modellierung gewählt werden. Ausgehend von einem einfachen initialen Modell kann durch Hinzunahme weiterer Einflussfaktoren zusätzliches Hintergrundwissen über das Anwendungsgebiet in den Modellierungsprozess einfließen. Dieses Hintergrundwissen ist vielen Sportbegeisterten durch ihre Erfahrungen vertraut, auch wenn sie es formal nicht ausdrücken können.

Ob dieses Wissen sinnvoll in ein formales Modell integriert werden kann, hängt natürlich davon ab, ob entsprechende Daten für das Training des Modells zur Verfügung stehen. Durch seinen Stellenwert in der Gesellschaft ist dies insbesondere für Fußballspiele gegeben, für die potenziell aus einer Vielzahl von Statistiken geschöpft werden kann.

Hieraus ergibt sich eine andere Ausgangslage im Vergleich zu Kap. 9. Der Nutzen jeder Modellerweiterung lässt sich anhand der historischen Daten in Hinblick auf die Qualität der Vorhersageergebnisse überprüfen.

Literatur

- (Aoki et al. 2017) “Luck is Hard to Beat: The Difficulty of Sports Prediction”, Raquel Aoki, Assunção Renato Martins, Pedro O.S. Vaz de Melo, 2017, <https://doi.org/10.1145/3097983.3098045>, <https://arxiv.org/abs/1706.02447>
- (Baio & Blangiardo, 2010) “Bayesian hierarchical model for the prediction of football results.”, *Journal of Applied Statistics*, 37 (2) 253–264.

- (Barber, 2012) Bayesian Reasoning and Machine Learning, David Barber, Cambridge University Press, 2012
- (Caparrós, 2022) “Key Performance Indicators – KPIs”, Marc Garnica Caparrós, in (Memmert 2022), https://doi.org/10.1007/978-3-662-63444-8_19
- (Deutscher Sportwettenverband, 2021) „Markt“, Deutscher Sportwetten Verband, 2021. <https://dswv.de/markt/>
- (Franke, 2021) “Das Milliardengeschäft mit der Spielsucht”, Martin Franke, Frankfurter Allgemeine Zeitung, 4. Januar 2021. <https://www.faz.net/aktuell/wirtschaft/schneller-schlau/sportwetten-in-deutschland-das-milliardengeschaeft-mit-spielsucht-17125985.html>
- (Karlis & Ntzoufras, 2003), ‘Analysis of sports data by using bivariate Poisson models’, Journal of the Royal Statistical Society D 52, 381–393.
- (Koller & Friedmann, 2009) Probabilistic Graphical Models, Daphne Koller, Nir Friedman, The MIT Press, Cambridge Massachusetts, 2009
- (McKinsey & Company, 2020) “Unternehmen Bundesliga, Die ökonomische Bedeutung des Profifußballs in Deutschland”, McKinsey & Company, https://media.dfl.de/sites/2/2020/09/2020_Bundesliga_Studie_Deutsch.pdf
- (Memmert, 2022) “Spielanalyse im Sportspiel”, Daniel Memmert (Hrsg.), Springer Spektrum, 2022, <https://doi.org/10.1007/978-3-662-63444-8>
- (Phatak, 2022) “Normalisierung von KPIs basierend auf Ballbesitz”, Ashwin Phatak, in (Memmert 2022), https://doi.org/10.1007/978-3-662-63444-8_21
- (van Harmelen & ten Teije, 2019) “A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems”, Frank van Harmelen, Annette ten Teije, Journal of Web Engineering, Vol. 18 1–3, 97–124. <https://doi.org/10.13052/jwe1540-9589.18133>
- (Wunderlich & Memmert, 2020) “Forecasting the outcomes of sports events: A review” Fabian Wunderlich, Daniel Memmert, European Journal of Sport Science, 2020 <https://doi.org/10.1080/17461391.2020.1793002>
- (Witschel et al., 2021) “Visualization of patterns for hybrid learning and reasoning with human involvement”, Hans Friedrich Witschel, Charuta Pande, Andreas Martin, Emanuele Laurenzi, Knut Hinkelmann“, In: Dornberger, R. (eds) New Trends in Business Information Systems and Technology. Studies in Systems, Decision and Control, vol 294. Springer, Cham. <https://doi.org/10.1007>, https://www.researchgate.net/publication/342784361_Visualization_of_Patterns_for_Hybrid_Learning_and_Reasoning_with_Human_Involvement

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.



Teil IV

Kombination von Machine Learning und wissensbasierter KI



Hybrides wissensbasiertes Reasoning Für wissensintensive Prozesse am Beispiel von Notrufabfragen

13

Eric Rietzke , Carsten Maletzki , Lisa Grumbach 
und Ralph Bergmann 

Kernaussagen

1. Als wissensintensiver Prozess wird der Verlauf einer Notrufabfrage durch das in Entscheidungsfindungen eingebrachte Wissen der mitwirkenden Menschen dominiert (Vaculín et al. 2011).
2. Die Integration dieses Wissens in ein prozessführendes System kann für eine wertvolle Entscheidungsunterstützung ausgenutzt werden.
3. In Notrufabfragen können Entscheidungen auf Basis regelbasierten und erfahrungsbasierten Wissens getroffen werden.
4. Regelbasiertes Wissen ist formalisierbar und kann als explizites Expertenwissen zur Unterstützung von Entscheidungsfindungen eingebracht werden.

E. Rietzke (✉) · C. Maletzki · L. Grumbach · R. Bergmann
Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI), Trier, Deutschland
E-Mail: eric.rietzke@dfki.de

C. Maletzki
E-Mail: carsten.maletzki@dfki.de

L. Grumbach
E-Mail: lisa.grumbach@dfki.de

R. Bergmann
E-Mail: ralph.bergmann@dfki.de

E. Rietzke
LiveReader GmbH, Oberthal, Deutschland

R. Bergmann
Lehrstuhl für Wirtschaftsinformatik II, Universität Trier, Trier, Deutschland

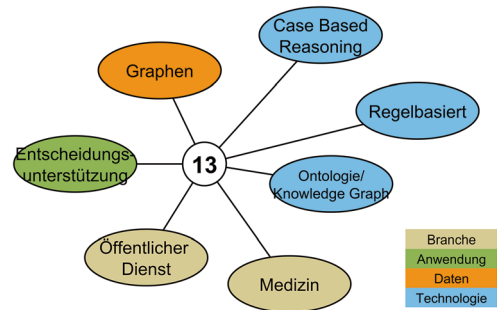
5. Erfahrungsbasiertes Wissen ist oft implizites Wissen, welches sich durch die Historie getroffener Entscheidungen manifestiert.
6. Regelbasiertes und fallbasiertes Schließen kann in einem hybriden Ansatz vereint werden, um Schwächen der jeweils anderen Methode teilweise auszugleichen.

13.1 Einleitung

Die Bewältigung von Naturkatastrophen, Pandemien und Großschadensereignissen erfordert vielfältige Maßnahmen, die zu einer Aktivierung von Feuerwehr, Rettungsdienst und anderen Behörden und Organisationen mit Sicherheitsaufgaben (BOS) führen. Die BOS reagieren hierbei oftmals auf Notrufe, die Bürgerinnen und Bürger bspw. unter der europaweiten Notrufnummer 112 absetzen. Entgegengenommen werden diese Notrufe durch Disponenten, die initial durch die Notsituation begleiten und eine angemessene Alarmierung von Einsatzkräften einleiten. Eine qualitativ hochwertige Notrufbearbeitung fordert dabei kommunikatorisches und analytisches Geschick (Hackstein 2017; Sudowe 2017). Im Kontext hochdynamischer Extremsituationen ist es zudem nötig, dass Disponenten die aktuelle Gesamtlage in Entscheidungsfindungen einbeziehen. Das Forschungsprojekt SPELL arbeitet daran, die Verfügbarkeit von Informationen über die aktuelle Lage zu erhöhen und andererseits die dadurch entstehende Informationsflut durch eine Auswertung mittels Dienste auf Basis künstlicher Intelligenz (KI) zu beherrschen (<https://spell-plattform.de/>). In diesem Kontext fokussiert sich dieses Kapitel auf die Herausforderung, wie diese entscheidungsrelevanten Erkenntnisse zur Entscheidungsunterstützung der Notrufabfrage beitragen können.

Das Forschungsfeld der KI wird heutzutage von datenbasierten Ansätzen und insbesondere künstlichen neuronalen Netzen dominiert. Künstliche neuronale Netze besitzen ein breites Anwendungsspektrum und werden dadurch sowohl bei Klassifikationsproblemen als auch zunehmend zur Generierung medialer Inhalte genutzt. Begrenzt wird dieses Spektrum vor allem durch Einschränkungen in der Erklärbarkeit von Ergebnissen und von einer oftmaligen Abhängigkeit von umfangreichen Datenmengen für maschinelle Lernverfahren. Ist das zu lernende Wissen bereits initial hinreichend bekannt oder können benötigte Lernmengen nicht oder nur mit unverhältnismäßig hohem Aufwand beschafft werden, eignen sich alternative KI-Methoden mitunter besser. Eine Methode, die üblicherweise mit deutlich reduzierten Lernmengen auskommt, ist das fallbasierte Schließen (engl.: Case-based Reasoning – CBR). Das fallbasierte Schließen baut im Kern auf einer Fallbasis aus historischen Fällen und ihren jeweiligen Lösungen auf. Das Ziel des fallbasierten Schließens ist es, anhand dieser Fallbasis eine geeignete Lösung für eine neue Situation abzuleiten. Hierbei liegt die Annahme zugrunde, dass sich ähnliche Probleme mit ähnlichen Lösungen bearbeiten lassen. Ganz ohne eine Lernmenge kann hingegen die Methode des regelbasierten Schließens (engl.: Rule-based Reasoning – RBR) auskommen. Das regelbasierte Schließen nutzt explizit formulierte Regeln als Grundlage, um über bekannte Sachverhalte neue Fakten abzuleiten. Gemein haben

Abb. 13.1 Thematische Einordnung des Kap. 13



hierbei beide Methoden eine inhärente Erklärbarkeit ihrer Schlussfolgerungen. Beim regelbasierten Schließen kann allerdings nur das explizit in Form von Regeln hinterlegte Wissen einen Beitrag leisten. Dies kann zu enormen Aufwänden bei Aufbau und Wartung des Regelwissens führen. Das fallbasierte Schließen auf der anderen Seite kann bei sehr wenigen oder keinen Fällen je nach Problem nur wenige spezifische oder sogar keine Lösungsvorschläge generieren.

Da Notrufende mitunter hochsensible private Informationen teilen, ist die Nutzung von Notfalldaten zur Bildung von Lernmengen hierzulande an enorme Aufwände geknüpft. Mit dem Blick darauf, dass Disponenten mitunter kritische Entscheidungen treffen müssen, ist zudem die Erklärbarkeit von Ergebnissen eine wichtige Eigenschaft. Wird aufgrund dieser Eigenschaften das regelbasierte Schließen angewandt, muss aufgrund des hohen Erstellungs- und Wartungsaufwandes von Regelwissen davon ausgegangen werden, dass Regeln die Anforderungen zur Bewältigung komplexer Aufgabenstellungen eher unvollständig abdecken. Allerdings könnte parallel hierzu sukzessive eine für das fallbasierte Schließen nötige Fallbasis aufgebaut werden. Das auf dieser Grundlage ermöglichte fallbasierte Schließen soll das unvollständige Regelwissen ergänzen und so Lücken des initialen Regelwerks schließen. Das vorliegende Kapitel führt in diesem Zusammenhang eine prozessorientierte und auf Knowledge-Graphen basierende Integration von regelbasiertem und fallbasiertem Schließen ein. Abb. 13.1 ordnet dieses Kapitel thematisch ein bezüglich der Branche, Anwendung, Daten sowie Technologien.

Abb. 13.2 zeigt das Boxology-Modell¹ des hybriden KI-Systems. Das in diesem Kapitel eingeführte hybride wissensbasierte Reasoning strebt eine parallele Ausführung von KI-Methoden zur Generierung entscheidungsrelevanter Hinweise an. Da diese Hinweise mitunter divergieren können, müssen sie im Rahmen eines Schrittes zur Zusammenführung (Reconciliation) aufeinander abgestimmt werden, bevor sie einem Disponenten als Entscheidungsgrundlage präsentiert werden.

¹ Für eine Einführung in die Boxology siehe Kap. 1.

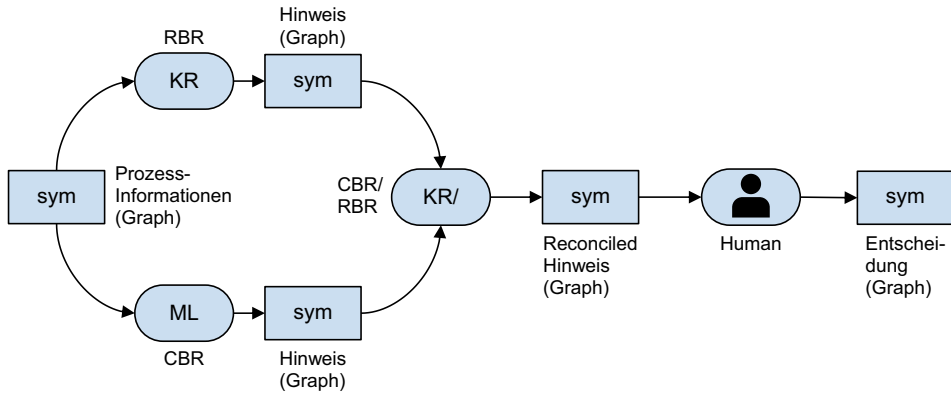


Abb. 13.2 Boxology-Modell des hybriden KI-Systems

Nachfolgend wird zunächst in die relevanten Grundlagen zur Notrufabfrage und dem hybriden wissensbasierten Reasoning eingeführt. Hiernach wird zunächst der methodische Ansatz definiert, unter dem hybrides wissensbasiertes Reasoning eigenes Wissen in den Prozess zur Unterstützung von Entscheidungsfindungen einbringen kann. In diesem Zusammenhang wird eine mögliche Softwarearchitektur zur Realisierung einer prozessorientierten Unterstützung in Entscheidungsfindungen vorgestellt. Anschließend wird definiert, wie diese Architektur in der Praxis implementiert wurde.

13.2 Notrufabfrage und hybrides wissensbasiertes Reasoning in Integrierten Leitstellen

Dieser Abschnitt dient der vertiefenden Analyse von Notrufabfragen und der Verdeutlichung des Mehrwertes, den hybrides wissensbasiertes Reasoning für Disponenten leisten kann. Zudem erfolgt eine Betrachtung der Notrufabfrage aus der übergeordneten Perspektive ihrer Prozesskategorie der wissensintensiven Prozesse. Dies soll es einem interessierten Leser erleichtern, analoge Problemstellungen zu identifizieren, auf die sich die in diesem Kapitel beschriebene Methodik übertragen lässt.

In Deutschland arbeiten Disponenten der Notrufnummer 112 üblicherweise in Integrierten Leitstellen, wo sie sowohl feuerwehrtechnische als auch medizinische Notrufe entgegennehmen. Notrufe, die dem Zuständigkeitsbereich der Polizei zuzuordnen sind, werden üblicherweise in räumlich und organisatorisch getrennten Leitstellen bearbeitet. Diese Separierung spiegelt sich auch in der Entscheidung wider, für polizeiliche Angelegenheiten die gesonderte Notrufnummer 110 anzubieten. Der Anwendungsfall dieses Kapitels beschränkt sich auf Notrufe, die bei der 112 eingehen. Notrufe im medizinischen Umfeld folgen laut Møller et al. einer iterativen Prozedur, dessen exakte Ausprägung von kontextuellen Einflüssen im Umfeld von Notrufenden und Disponenten

abhängt (Møller et al. 2021). Im Rahmen dieser iterativen Prozedur (Abb. 13.3) erheben Disponenten Informationen von Notrufenden durch das Stellen von Fragen, deren Antworten zur Entwicklung einer Vorstellung von den Geschehnissen am Ort des Notfalls beitragen. Diese Vorstellung, welche auch Mental Picture genannt wird, dient als Grundlage zur Ableitung von Entscheidungen und ggf. infolgedessen zur Identifikation von zu erledigenden Aufgaben. Eine solche Aufgabe kann z. B. die Alarmierung eines Rettungswagens umfassen. Die iterative Prozedur als Ganzes wird durch Dialoge zur Eröffnung und zum Abschluss des Gesprächs gerahmt, welche vor allem zum Abgleich der individuellen Erwartungen der Gesprächsteilnehmer dienen. Kontextuell beeinflusst wird die iterative Prozedur aufseiten der Notrufenden vor allem durch ihren individuellen Notrufgrund und ihre Fähigkeit, das aktuelle Problem wahrnehmen und beschreiben zu können. Einflüsse aus dem Umfeld von Disponenten hingegen entstehen vor allem durch den Umfang der ihnen zur Verfügung stehenden Informationen und das Wissen, das zur Interpretation beim Aufbau des mentalen Bildes benötigt wird. Wenngleich Møller et al. das vorgestellte Modell mit dem Fokus auf medizinische Notrufe entwickelten, so erwies es sich in der Arbeit der Autoren in Expertenworkshops auch als geeignet für eine Übertragung auf den feuerwehrtechnischen Bereich.

Bei der Bearbeitung von 112-Notrufen stehen Disponenten vor nennenswerten Herausforderungen. Hierbei müssen sie sich unter Zeitdruck ein ausreichend präzises Mental Picture von möglicherweise lebensbedrohlichen Situationen und den Geschehnissen am Notfallort verschaffen. Die hierfür benötigten Informationen müssen sie von Anrufern in Ausnahmesituationen erfragen, die möglicherweise unter einer enormen emotionalen Belastung stehen. Dabei kann der beschriebene Zeitdruck zu einem Konflikt zwischen einer schnellen und präzisen Entscheidung führen. Eine schnelle, aber oberflächliche Bearbeitung eines Notrufs kann zu einem Fehleinsatz führen, bei dem Einsatzkräfte für eine Belanglosigkeit ausrücken. Da die alarmierten Einsatzkräfte für zeitgleich

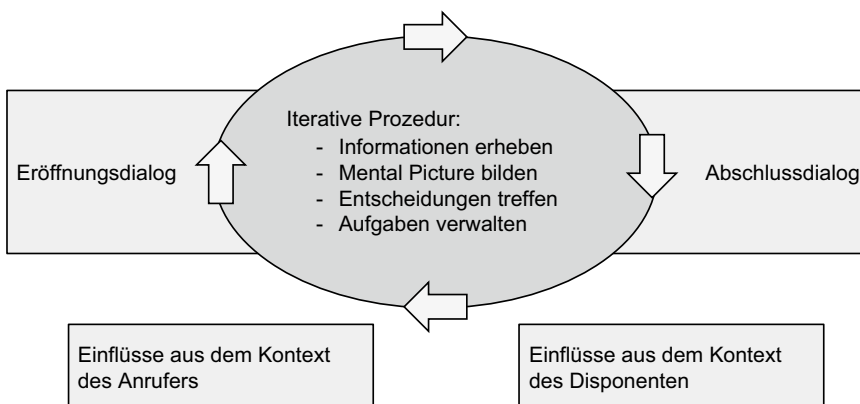


Abb. 13.3 Modell der Notrufbearbeitung (in Anlehnung an Møller et al. 2021)

stattfindende kritische Notfälle dann nicht mehr zur Verfügung stehen, können Fehlalarmierungen zu Lücken in der Patientenversorgung führen.

Der Prozess der Notrufabfrage lässt sich der Kategorie der wissensintensiven Prozesse (engl. Knowledge-intensive Processes – KiPs) zuordnen. Die erfolgreiche Ausführung derartiger Prozesse besitzt eine starke Abhängigkeit von dem Wissen der prozessausführenden Personen (Wissensarbeiter; Vaculín et al. 2011). Während Wissensarbeiter in vielfältigen Branchen zu finden sind, wie zum Beispiel im Finanz-, Ingenieur- und Gesundheitswesen, liegt der Fokus ihrer Tätigkeit vor allem auf der Generierung, Verteilung und Anwendung von Wissen (Davenport 2005). In diesem Zusammenhang stellt die Unterstützung von Wissensarbeitern durch angemessene prozessführende Systeme ein bislang nicht abschließend gelöstes Problem dar. Besonders relevant wird diese Thematik bei der Betrachtung, dass Wissensarbeiter einen enormen Einfluss auf den wirtschaftlichen Wert ihrer Organisationen besitzen (Davenport 2005). Allerdings ist der Entwurf eines wirkungsvollen Prozesssystems zur Unterstützung von Wissensarbeitern an einige herausfordernde Anforderungen (Di Ciccio et al. 2015) geknüpft. Diese Anforderungen leiten sich aus den wesentlichen Charakteristika (Di Ciccio et al. 2015) von KiPs ab. KiPs sind wissensgetrieben, kollaborationsorientiert, unvorhersehbar, sich entwickelnd, zielorientiert, ereignisgetrieben, regelgetrieben und nicht wiederholbar. Damit sich das eingebrachte Wissen der Wissensarbeiter nutzbringend entfalten kann, fokussieren wir im Rahmen unserer Arbeit (Rietzke et al. 2021) im Besonderen drei Anforderungen:

- **Flexibilität:** Anwender benötigen die Freiheitsgrade, um abseits von vordefinierten Ablaufschemas (Kontrollfluss) agieren zu können.
- **Adaptivität:** KiPs müssen sich zur Laufzeit auf wechselnde Prozessziele ausrichten können.
- **Datengetriebenheit:** Die verfügbaren Daten in einem KiP sind die Grundlage für die fortlaufende Prozessausführung.

Wir betrachten diese Anforderungen als zentrale Voraussetzungen für ein Prozesssystem zur Unterstützung von KiPs. Das hierbei in KiPs eingebrachte Wissen unterteilen wir in zwei Kategorien. Explizites Wissen ist Wissen, das durch Gesetze und Richtlinien bereits vorliegt oder in einer solchen Form zum Ausdruck gebracht und letztlich durch Regeln formalisiert werden kann. Implizites Wissen ist Wissen, welches oftmals durch Erfahrungswissen vorliegt, einen wichtigen Prozessbeitrag leisten kann, aber zugleich oft unsichtbar einwirkt und kaum in Regeln formuliert werden kann. Am Beispiel der Notrufabfrage lässt sich in diesem Hinblick feststellen, dass Disponenten umfassende Regelwerke anwenden müssen. Dies umfasst zum einen das Wissen über allgemeine Regelwerke zur Patientenversorgung, medizinische Sachverhalte und leitstellenspezifische Richtlinien. Am Beispiel medizinischer Sachverhalte definiert der ärztliche Leiter des Rettungsdienstes, welche Symptome als medizinische Notfälle einzustufen sind und eine Entsendung festgelegter Einsatzkräfte bedürfen. Aufgrund der Vielzahl komple-

xer Notfälle mit jeweils sehr individuellem Charakter deckt dieses explizite Regelwerk jedoch nur einen Teil des Wissens ab, das von Disponenten zur Entscheidungsfindung eingebracht werden muss. Zusätzlich wird Erfahrungswissen eingebracht, welches als implizites Wissen vorliegt und durch die Historie von Notrufvorgängen gesammelt wurde. Ein Beispiel eines kritischen Notfalls in der Stadtmitte verdeutlicht dies: Ein Patient zeigt deutliche Hinweise auf einen Herzinfarkt – dies bedeutet, dass er binnen einer fest vorgegebenen Zeit durch den Rettungsdienst erreicht und umgehend in eine Spezialklinik überführt werden muss. Die Regeln sehen den Einsatz eines Rettungshubschraubers vor. Mit der Erfahrung der Disponenten aus einem früheren Fall wird jedoch berücksichtigt, dass dieser nicht vor Ort in der Stadtmitte landen kann. Deshalb stellt ein Rettungswagen die schnellere und bessere Option dar.

Heutzutage herrscht ein allgemeiner Konsens dahin gehend, dass Notrufrisponenten von einem Entscheidungsunterstützungssystem begleitet werden sollten, um die Patientensicherheit durch eine gleichmäßigere Anwendung von Wissen zu erhöhen (Baumann 2017; Mayr 2020). Alle verfügbaren Systeme basieren hierbei im Wesentlichen auf demselben methodischen Ansatz eines Entscheidungsbaumes, um Wissen zur Entscheidungsunterstützung in den Prozess einzubringen. Problematisch ist hierbei jedoch, dass diese Entscheidungsbäume einen fest einzuhaltenden Ablauf erfordern. Insbesondere erfahrene Disponenten sehen die daraus resultierende Form der Unterstützung als zu starr an (Baumann 2017). Zudem hat dieser Ansatz eine grundsätzliche Schwäche: Ein Notruf darf nur eine begrenzte Zeit dauern, weshalb ein Entscheidungsbaum nur eine beschränkte Tiefe aufweisen kann. Diese Beschränkung wird der Komplexität der Realität nicht gerecht, weshalb die Entscheidungsbäume nur die wesentlichsten und kritischsten Pfade eines Notrufs abdecken und die weniger kritischen Notrufe weitestgehend ignoriert werden. Infolgedessen bleiben laut Umfragen (Rietzke 2021) weit über 90 % der Notrufe ohne eine adäquate Unterstützung durch ein begleitendes System. Dies resultiert darin, dass Diagnosen nicht präzise bzw. vollständig gestellt werden können und somit eine darauf abgestimmte Maßnahme verzögert wird.

Abb. 13.4 zeigt einen Prototypen zur Umsetzung von hybridem wissensbasierten Reasoning zur Unterstützung von Notrufabfragen. In der dargestellten Situation klagt ein Patient über Atemnot und berichtet über einen Allergenkontakt. Dies aktiviert im Regelwissen des Systems einen deutlichen Hinweis auf eine Anaphylaxie. Das System informiert den Disponenten hierüber und erfragt, ob auch dieser von einer Anaphylaxie ausgehen möchte. Falls dem so ist, muss dem Patienten schnellstmöglich ein adrenalinhaltiges Medikament verabreicht werden. Durch die Atemnot befindet sich der Patient bereits in einem lebensbedrohlichen Zustand, sodass auch ohne das Feststehen einer Verdachtsdiagnose die Entsendung eines schnellstmöglichen Rettungsmittels vorgeschlagen wird. Regelbasiert wird hierbei identifiziert, dass die Entsendung eines Rettungshelikopters (RTH) derzeit die beste Option ist, um den Einsatzort zu erreichen. Hierbei wird anhand der Fallbasis des Systems zudem identifiziert, dass ein RTH bei nahe gelegenen Einsätzen in der Vergangenheit aufgrund mangelnder Landemöglichkeiten zur nächsten Feuerwache entsandt wurde. In derartigen Fällen wurde zusätzlich zum RTH

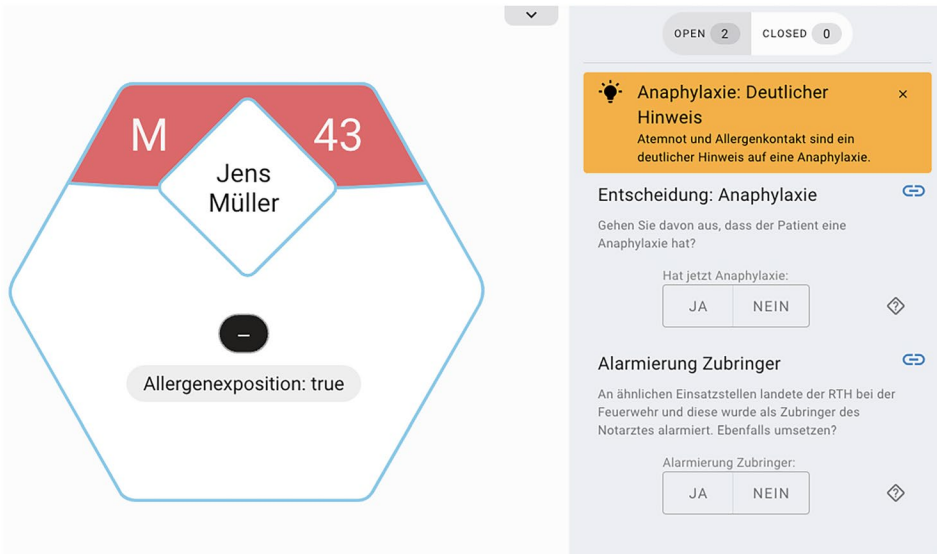


Abb. 13.4 Beispielszenario für hybrides wissensbasiertes Reasoning

noch ein Einsatzfahrzeug der Feuerwehr alarmiert, um den Notarzt des RTH schnellstmöglich vom Landeplatz zum Einsatzort zu bringen. Das System schlägt diese Maßnahme vor und falls der Disponent diese annimmt, führt das System den Disponenten durch die Alarmierung der konkreten Rettungsmittel.

13.3 Technische Grundlagen des hybriden wissensbasierten Reasoning

Regelbasiertes Schließen, engl. Rule-based Reasoning (RBR; Buchanan und Duda 1983), ist eine Methode der symbolischen KI. Die angewandte Wissensbasis ergibt sich hier aus explizit modelliertem Wissen, den Fakten und darauf aufbauenden zu definierenden logischen Regeln. Während der Laufzeit werden die Regeln auf die vorhandenen Informationen angewandt, um mögliche Schlussfolgerungen zu ziehen und somit neues Wissen abzuleiten. Regelbasierte Ansätze eignen sich vor allem für generelles Wissen und sind statisch, da ein Lernen nur durch Ergänzen oder Anpassen von Regeln möglich ist. Gleichzeitig sind die Ergebnisse nachvollziehbar und erklärbar durch die explizite Anwendung der Regeln. Auf der anderen Seite ist ein hoher initialer Modellierungs- und Wartungsaufwand notwendig, um die Wissensbasis aufzubauen.

Fallbasiertes Schließen, engl. Case-based Reasoning (CBR), ist ein Problemlösungsansatz, der auf der Annahme basiert, dass ähnliche Probleme ähnliche Lösungen erfordern (Aamodt und Plaza 1994; Bergmann et al. 2021). Dabei werden Erfahrungen aus früheren Situationen auf aktuelle Probleme übertragen, d. h., das Erfahrungswissen wird wiederverwendet. Als instanzbasierter Ansatz kann im CBR bereits eine kleine Menge an Daten, also spezifisches Wissen, zum erfolgreichen Einsatz und hohen Nutzen führen. Die dabei verwendete Wissensbasis ergibt sich implizit aus den protokollierten Daten, den *Fällen*. Während der Problemlösung wird nach einem oder mehreren ähnlichen Fällen, die auf Basis eines definierten Ähnlichkeitsmaßes berechnet werden, gesucht und deren Lösung evtl. angepasst und auf das aktuell betrachtete Problem übertragen. Die vorgeschlagene Lösung wird im nächsten Schritt entweder übernommen und angewandt oder nochmals überarbeitet. Das aktuell betrachtete Problem kann mit der entsprechenden Lösung wiederum in die Wissensbasis aufgenommen werden und somit die nutzbaren Erfahrungen stetig ergänzen und zu einem dynamischen, lernenden System führen.

Hybride künstliche Intelligenz beschreibt die Kombination von unterschiedlichen Verfahren aus dem Bereich der KI. In dem vorgestellten Ansatz fokussieren wir auf regelbasierte und fallbasierte Ansätze. Diese können auf verschiedene Art und Weise kombiniert werden. Laut der Klassifikation nach Prentzas und Hatzilygeroudis (2007) werden drei unterschiedliche Integrationsmöglichkeiten unterschieden, je nach dem Zeitpunkt und der Art und Weise, wie regelbasiertes und fallbasiertes Schließen auf das Problem angewandt werden. Beide Ansätze können parallel, sequenziell oder integriert zur Lösung herangezogen werden. Die integrierte Anwendung nutzt eine der beiden Methoden als Teil der anderen Methode, z. B. könnte die Adaption innerhalb einer fallbasierten Methode regelbasiert erfolgen. Bei der sequenziellen Verarbeitung wird das Ergebnis der einen Methode von der anderen Methode weiterverarbeitet, so könnte ein regelbasiertes Ergebnis genutzt werden, um nach einem ähnlichen Fall zu suchen.

Im hier vorgeschlagenen Konzept kommt die dritte Integrationsart zur Anwendung, die beide Methoden parallel auf die gleiche Eingabe und zum gleichen Zweck anwendet (Abb. 13.5). Besonderen Fokus muss hierbei auf die sogenannte Reconciliation gelegt werden, da sich die beiden Ergebnisse unterscheiden oder sogar widersprechen können. (z. B. Gates et al. 2019). Um Ergebniskonflikte aufzulösen oder zur Entscheidungsunterstützung entsprechend aufzubereiten bzw. zu erklären, existieren verschiedene Methoden, wie z. B. die Berechnung des Mittelwertes der Ähnlichkeit, probabilistisches Schließen oder auch die regelbasierte Abstimmung der Ergebnisse (Prentzas und Hatzilygeroudis 2007).

Einige Ansätze existieren, die auch die Reconciliation-orientierte Kombination von CBR und RBR umsetzen. Golding und Rosenbloom nutzen einen fallbasierten Ansatz, um unvollständige Regeln zu ergänzen, um die korrekte Aussprache von Begriffen abzuleiten (Golding und Rosenbloom 1996). Im Versicherungskontext wurde eine hybride KI entwickelt, die fallbasiert explizit Ausnahmen der Regeln abbildet (Lee 2002). Eine regelbasierte Reconciliation wurde von Bruninghaus (1994) umgesetzt, um die Inter-

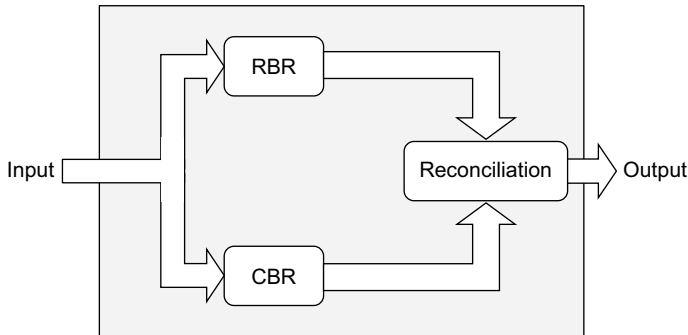


Abb. 13.5 Reconciliation-oriented Model nach Prentzas und Hatzilygeroudis (2007)

pretation und Auslegung von Gesetzestexten zu unterstützen. Auch in der medizinischen Domäne wurden verschiedene Kombinationen von CBR und RBR untersucht. In der Arbeit von Phuong et al. wird eine hybride KI genutzt, um die Diagnose von Lungenerkrankungen zu unterstützen (Phuong et al. 2001). Der hybride Ansatz von Saraiva et al. nutzt Regeln, um die Ähnlichkeitsberechnung zu spezifizieren, indem die Attribute der Fälle je nach Ausprägung gewichtet werden. Diese integrierte Kombination von RBR und CBR soll Krebsdiagnosen verbessern (Saraiva et al. 2015).

13.4 Softwarearchitektur und methodischer Ansatz

In diesem Abschnitt beschreiben wir einen hybriden Ansatz, der darauf abzielt, regelbasiertes und fallbasiertes Wissen in ein prozessausführendes Informationssystem einzubringen, um eine nutzenstiftende Kollaboration zwischen Mensch und Maschine zu ermöglichen. Hierzu wird das explizite Wissen in Form von Regeln durch das Verfahren des regelbasierten Schließens operationalisiert, während das Erfahrungswissen durch das Verfahren des fallbasierten Schließens adressiert wird. Beide Verfahren nutzen hierbei das Informationsmodell des Prozesssystems.

Aufgrund der Problematik mit den heutzutage angewendeten Entscheidungsbäumen zur Einbringung von Wissen in die Notrufabfrage wird nachfolgend zunächst ein alternativer methodischer Ansatz eingeführt. Dieser methodische Ansatz stellt die Grundlage für die Unterstützung von Disponenten durch ein hybrides wissensbasiertes Reasoning vor. Wenngleich der Ansatz anhand der Notrufabfrage hergeleitet wird, ist er aufgrund seiner generischen Ausrichtung grundsätzlich auch auf andere wissensintensive Prozesse übertragbar. Im Anschluss an die Einführung des methodischen Ansatzes wird dargelegt, wie die Softwarearchitektur des Entscheidungsunterstützungssystems ausgestaltet sein kann, um nach der angestrebten Methodik explizites sowie implizites Wissen zur Entscheidungsfindung in den Prozess einzubringen.

Das Ziel der in diesem Abschnitt beschriebenen Anwendung von hybridem wissensbasierten Reasoning ist die Unterstützung von Entscheidungsfindungen in wissensintensiven Prozessen. Hierfür soll das System regelbasiertes und fallbasiertes Wissen in die Ausführung der Prozesse einbringen. Die Rolle des Systems lässt sich hierbei durch die eines Co-Piloten charakterisieren, der einem Entscheidungsträger wichtige Hinweise für eine eigene unabhängige Entscheidung gibt. Eine gelungene Entscheidungsunterstützung reduziert die menschliche Variabilität in wissensintensiven Entscheidungen und harmonisiert divergierende menschliche Wissensgrundlagen. Hierfür orientiert sich der methodische Ansatz des vorgestellten Systems an dem in der Notrufabfrage beobachteten Konzept von Mental Pictures (Møller et al. 2021; Blandford und Wong 2004). Laut Møller et al. (2021) entstehen Mental Pictures durch die Anwendung des Wissens eines Disponenten zur Interpretation verfügbarer Informationen, um hiervon geeignete Entscheidungen abzuleiten. Analog hierzu soll das vorgestellte System das durch KI-Methoden bereitgestellte Wissen nutzen, um verfügbare Informationen zu interpretieren. Dies erfolgt mit dem Ziel der Ableitung einer Entscheidungsunterstützung. Durch die Co-Piloten-Rolle des Systems besteht diese Entscheidungsunterstützung aus einer Menge von Hinweisen, die nachfolgend künstliches Mental Picture genannt werden. Abb. 13.6 verdeutlicht, wie dieses künstliche Mental Picture im vorgestellten System zustande kommt, und zeigt, inwiefern es in Kombination mit dem menschlichen Mental Picture Einfluss auf Entscheidungsfindungen in wissensintensiven Prozessen nimmt. Dabei entspricht die Struktur des methodischen Ansatzes einem Zyklus, bei dem eine Entscheidung eine Erweiterung des initialen Informationsstandes darstellt, der wiederum eine Aktualisierung der Mental Pictures und infolgedessen konsekutiven Entscheidungsfindungen ermöglicht. Der Vorteil dieses Ansatzes gegenüber Entscheidungsbäumen liegt darin, dass er – anstatt von einem festen Ablauf auszugehen – auf den aktuellen Informationsstand blickt und auf dieser Grundlage adaptiv identifiziert, wie mit dem Prozess fortgefahren werden soll.

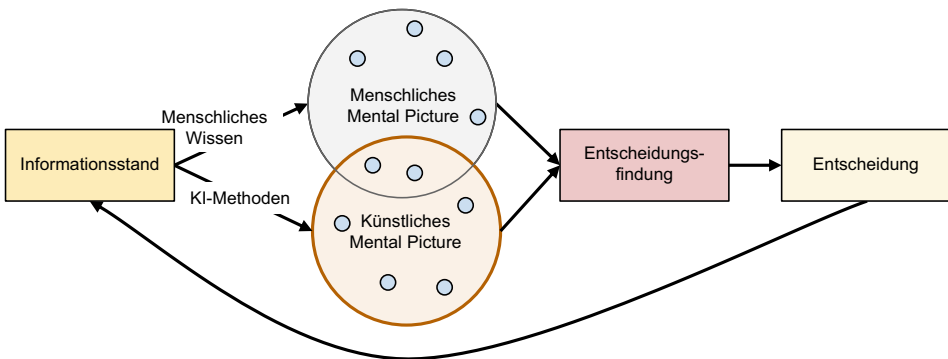


Abb. 13.6 Methodischer Ansatz des vorgestellten Systems

Der beschriebene methodische Ansatz des vorgestellten Systems bezweckt eine Entscheidungsfindung unter der Einflussnahme des künstlichen Mental Picture. Die Entscheidungsfindung muss hierbei mit dem Umstand umgehen, dass sich Teile von menschlichen und künstlichen Mental Pictures inhaltlich überlappen oder abweichen – je nachdem, ob das jeweils eingebrachte Wissen zu derselben Einschätzung der Situation führt. Kommt es zu einer Abweichung, muss diese vor einer Entscheidung abgewogen werden. In diesem Kontext obliegt es letztlich dem Menschen, eine Entscheidung zu treffen. Hierfür gibt es generell drei Optionen: (1) Hinweis genauer untersuchen, (2) Hinweis als Grundlage für weitere Entscheidungen annehmen oder (3) Hinweis ablehnen. Die Entscheidung für eine genauere Untersuchung (1) führt zu der Anforderung, dass das System vorschlagen sollte, was unternommen werden muss, um Informationen zur Verdeutlichung des Hinweises zu erheben. Im Gegensatz dazu bedeutet die Annahme des Hinweises als Grundlage für weitere Entscheidungen (2), dass dieser bereits als deutlich genug angesehen wird, um weitere darauf aufbauende Entscheidungen zu treffen. Dies kann zum Beispiel bei der Notrufabfrage zur Einleitung einer Notfallreaktion führen. Eine Ablehnung (3) hingegen bedeutet, dass einem identifizierten Hinweis im weiteren Verlauf nicht aktiv nachgegangen werden soll. Unabhängig davon, welche Entscheidung aufgrund eines Hinweises getroffen wird, stellt diese eine Erweiterung des initialen Informationsstandes dar, die den weiteren Prozessverlauf beeinflusst.

Hybrides wissensbasiertes Reasoning führt im Kontext des vorgestellten methodischen Ansatzes zwei KI-Methoden zusammen, um ein künstliches Mental Picture zu generieren. Dieses künstliche Mental Picture soll einen Einfluss auf Entscheidungsfindungen in wissensintensiven Prozessen nehmen. Da der Ansatz eine parallele Anwendung der KI-Methoden anstrebt, ist es möglich, dass entstehende Hinweise im künstlichen Mental Picture durch unterschiedliche Wissensgrundlagen nur einseitig geschlussfolgert werden oder widersprüchlich sind. Dies wird in Abb. 13.7 genauer dargestellt.

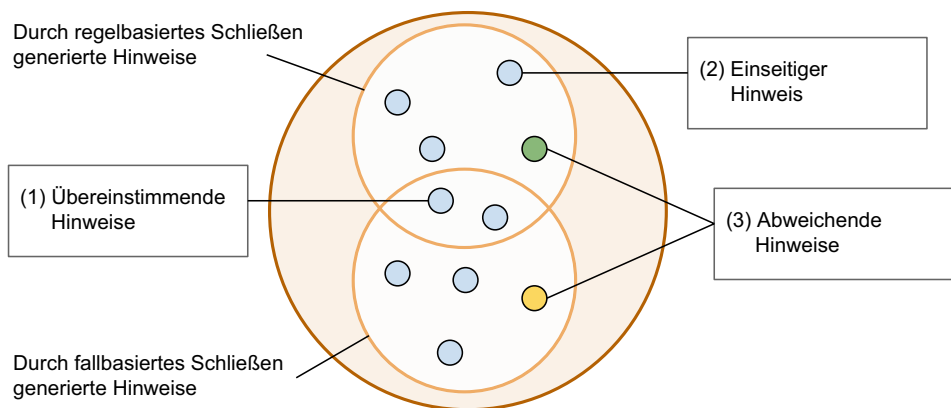


Abb. 13.7 Aufteilung von künstlichen Mental Pictures zur Entscheidungsunterstützung

Im Hinblick auf eine möglichst hochwertige Entscheidungsunterstützung sollten diese Zustände aufgelöst werden.

Im Hinblick auf eine Entscheidungsunterstützung unproblematisch sind übereinstimmende Hinweise (1), bei denen sowohl das fallbasierte als auch das regelbasierte Schließen zu demselben Hinweis gekommen sind. Dieser Hinweis kann ohne eine weitere Analyse in die Entscheidungsfindung eingehen. Wurde hingegen ein Hinweis nur einseitig durch eine KI-Methode (2) identifiziert, deckt dessen eingebrachtes Wissen eine Wissenslücke der jeweils anderen KI-Methode ab. Kommen fallbasiertes und regelbasiertes Schließen zu abweichenden Hinweisen (3), bedarf es in jedem Fall einer Analyse der identifizierten Hinweise unter Anwendung einer Strategie zur Auflösung des Konfliktes. Erst das Ergebnis dieser Auflösung sollte in die Entscheidungsfindung eingehen. Ist es nicht möglich, den Konflikt aufzulösen, sollte dieser derart aufbereitet werden, dass er durch einen Menschen im Rahmen der Entscheidungsfindung aufgelöst werden kann.

Die Umsetzung einer Reconciliation bei abweichenden Hinweisen kann durch eine weitere Anwendung des regelbasierten und fallbasierten Schließens erfolgen. Eine regelbasierte Auflösung schreibt hierbei fest vor, welcher Hinweis unter welchen Umständen wie gehandhabt werden soll. In Bezug auf Notrufe könnte eine solche Regel einen Konflikt nach der Maßgabe auflösen, dass stets der Hinweis bevorzugt wird, der die Patientensicherheit erhöht – selbst wenn dies eine Fehldisposition zur Folge haben könnte. Fallbasiert könnte eine solche Situation dadurch aufgelöst werden, dass derjenige Hinweis priorisiert wird, der in der Vergangenheit in ähnlichen Fällen zu einer positiv bewerteten Entscheidung geführt hat. Ein konkreter Mechanismus hierfür wurde bereits skizziert (Maletzki et al. 2023).

Im Kontext des methodischen Ansatzes des vorgestellten Systems führt eine regelbasiert oder fallbasiert umgesetzte Reconciliation zu einer Erweiterung des künstlichen Mental Picture. Dies wird in Abb. 13.8 am Beispiel einer regelbasierten Auflösung eines Konfliktes verdeutlicht. Die hierbei gezeigten Bereiche 1–3 fassen dabei noch einmal zusammen, wie das Ergebnis der Reconciliation zu den ursprünglichen Schlussfolgerungen steht. Liegt das Ergebnis hierbei im Bereich 1 oder 2, wurde bei der Reconciliation eine Auffassung angenommen, die einem zuvor ermittelten Schluss entspricht. Hierbei kann ein Konflikt aufgelöst worden sein oder, im Falle eines einseitigen Hinweises, dieser für eine Entscheidungsunterstützung angenommen worden sein. Liegt das Ergebnis im Bereich Nr. 3, wurde im Rahmen der Reconciliation eine neue Form von Hinweis gebildet. Dieser Hinweis beschreibt die unklare Situation in einer Weise, die es einem Menschen ermöglicht, anhand des eigenen Mental Picture eine Auflösung vorzunehmen.

Im eingeführten methodischen Ansatz sind verfügbare Informationen die Grundlage, um Hinweise zu ermitteln, auf die durch Entscheidungen im Rahmen des wissensintensiven Prozesses reagiert werden muss. Getroffene Entscheidungen sind wiederum Informationen, die Auswirkungen auf den Prozess und künftige Entscheidungen haben. Der Informationsstand des Prozesses stellt in diesem Zusammenhang sowohl den Aus-

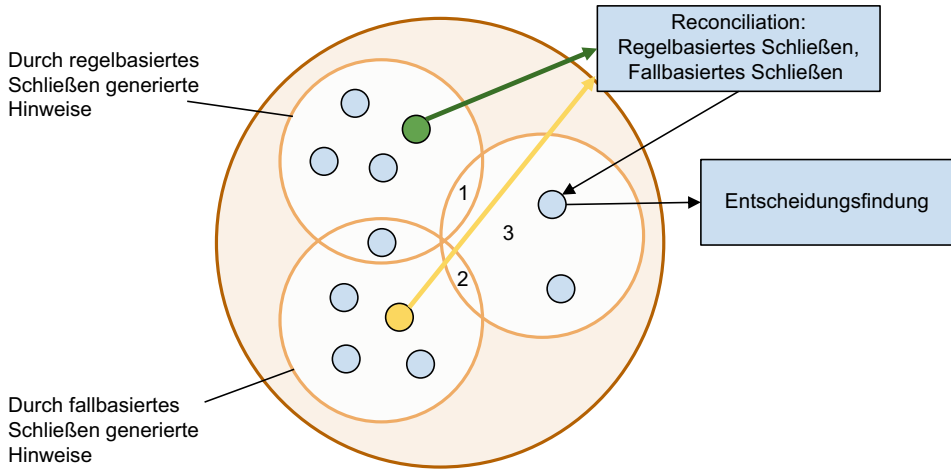
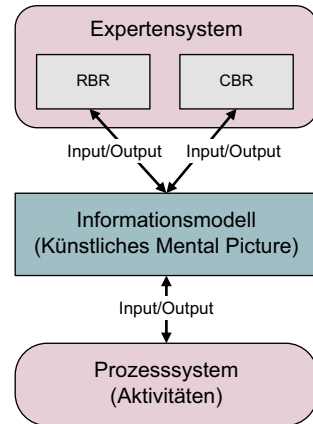


Abb. 13.8 Erweiterung des künstlichen Mental Picture durch Reconciliation

gangs- als auch Zielpunkt von Entscheidungsfindungen dar. Die Art und Weise, nach der bei einer Anwendung dieses Ansatzes Entscheidungen getroffen werden, lässt sich durch diesen Fokus auf verfügbare Informationen als datengetrieben einordnen. Bei einer datengetriebenen Sichtweise auf Prozesse bestimmt die Verfügbarkeit von Informationen, welche Aktivitäten ausgeführt werden können und sollen (Rietzke et al. 2018; Maletzki et al. 2019). Prozesssysteme, die einem datengetriebenen Paradigma folgen, bauen oftmals auf eine enge Verzahnung zwischen den verwalteten Prozessmodellen und den aktuell verfügbaren Informationen. Dies kann durch die Abbildung von Input- und Output-Beziehungen zwischen den Aktivitäten eines Prozesses und den im Prozess behandelten Informationen umgesetzt werden. Mit dem Blick auf eine Softwarearchitektur zur Umsetzung des methodischen Ansatzes ist es empfehlenswert, dies zu übernehmen. Abb. 13.9 zeigt in diesem Zusammenhang eine Skizze der Softwarearchitektur des in diesem Beitrag vorgestellten Systems. Hierbei wird auf einem datengetriebenen Prozesssystem aufgebaut, dessen Aktivitäten über Input- und Output-Beziehungen mit einem Informationsmodell zur Haltung des aktuellen Informationsstandes verknüpft sind. Hierbei beinhaltet der Informationsstand alle prozessrelevanten Informationen. Nach dem methodischen Ansatz bestehen künstliche Mental Pictures ebenfalls aus prozessrelevanten Informationen, die Eingang in die durch das Prozesssystem verwalteten Aktivitäten für Entscheidungsfindungen nehmen sollten. Nur so können die durch das fallbasierte und regelbasierte Schließen ermittelten Hinweise einen Einfluss auf Entscheidungsfindungen haben. Deshalb muss die Architektur zudem die Möglichkeit schaffen, die durch KI-Methoden ermittelten Hinweise im Informationsmodell zu ergänzen. Andererseits müssen KI-Methoden auch mit aktuell verfügbaren Informationen versorgt werden, um die Hinweise generieren zu können. Analog zum Prozesssystem wird hierfür

Abb. 13.9 Architektur des Systems



ein generisches Expertensystem für hybride KI integriert. Dieses Expertensystem verwaltet die Bindung der KI-Methoden an das Informationsmodell und steuert ihre Ausführung.

Dieser Abschnitt führte den methodischen Ansatz und die hieraus abgeleitete Softwarearchitektur für das in diesem Kapitel vorgestellte System zur Unterstützung von Entscheidungsfindungen in wissensintensiven Prozessen ein. Durch den methodischen Ansatz basiert die Unterstützung von Entscheidungsfindungen auf der Dualität von menschlichen und künstlichen Mental Pictures. Unstimmigkeiten im künstlichen Mental Picture, die auf dieser Grundlage im Hinblick auf eine Entscheidungsfindung entstehen, werden im Rahmen einer Reconciliation adressiert. Die Reconciliation führt hierbei zu einer Erweiterung des künstlichen Mental Picture. Des Weiteren wurde in diesem Abschnitt die grobe Architektur des vorgestellten Systems skizziert, welches jedes seiner Module über ein Informationsmodell integriert. Dadurch wird es möglich, dass die über das Expertensystem integrierten KI-Methoden einen Einfluss auf Aktivitäten zur Entscheidungsfindung im Prozesssystem nehmen können.

13.5 Technologien und Implementierung

Eine Implementierung der vorgestellten Softwarearchitektur impliziert die Notwendigkeit eines engen Beziehungsgeflechtes zwischen Aktivitäten, KI-Methoden und dem Informationsstand. Als Technologie zur Realisierung dieses Beziehungsgeflechtes bietet sich die Anwendung von Wissensgraphen im Kontext weiterer semantischer Technologien an. Um das Schema des Wissensgraphen zu spezifizieren, wird auf das Ontology- and Data-Driven Business Process Model (ODD-BP; Rietzke et al. 2021) zurückgegriffen. ODD-BP wurde mit dem Fokus auf die Integration von Daten, Wissen und Prozessen in einer einheitlichen semantischen Wissensbasis zur Unterstützung wissens-

intensiver Prozesse entwickelt. ODD-BP verfolgt in diesem Zusammenhang einen datenzentrierten Ansatz zur Prozessmodellierung und -ausführung. Für die datenzentrierte Steuerung einer Prozessausführung stellt ODD-BP zudem eine auf OWL/SWRL basierende Inferenzierungsmethodik bereit (Maletzki et al. 2019). Das von der vorgestellten Softwarearchitektur referenzierte Informationsmodell besteht bei ODD-BP aus Dataobjects zur Repräsentation von prozessrelevanten Entitäten und Attributen, die die Eigenschaften dieser Entitäten genauer durch entsprechende Werte beschreiben (Abb. 13.10). Am Beispiel der Notrufabfrage könnte ein Patient durch ein Dataobject repräsentiert werden, während seine Vitalwerte, Symptome und sonstigen Personendaten als Attribute abgebildet werden. Allgemein knüpfen in ODD-BP an den Attributen des Informationsmodells die Aktivitäten (nachfolgend: Task) der Prozessmodelle über Input- und Output-Relationen (Input: *required_by*; Output: *delivers*) an. Tasks können hierbei entweder manuell durch einen Menschen (User Task) oder automatisch durch ein System (System Task) ausgeführt werden.

ODD-BP und die vorgestellte Architektur wurde im Rahmen der Arbeit der Autoren bereits durch eine Implementierung umgesetzt. Das hierbei entstandene Software-Produkt trägt den Namen Notitia. Notitia wendet ODD-BP derart an, dass Integrationen von KI-Methoden über Implementierungen von System Tasks erfolgen können. Dies ermöglicht ein skriptbasiertes Reasoning, bei dem Regeln nativ definiert werden können. Fallbasiertes Schließen wird in diesem Zusammenhang über den skriptbasierten Aufruf von entsprechenden Diensten umgesetzt. In diesem Zusammenhang kann auf verschiedene Frameworks für das fallbasierte Schließen mit unterschiedlichen Schwerpunkten zurückgegriffen werden (Schultheis et al. 2023). ProCAKE (Bergmann et al. 2019) ist hierbei ein domänenunabhängiges Framework für strukturelles und prozessorientiertes fallbasiertes Schließen. Integriert in das vorgestellte hybride System besteht das Datenmodell aus einem Teilbereich des gesamten Informationsmodells. Somit können die Fälle aus den abgeschlossenen historischen Prozessinstanzen abgeleitet werden

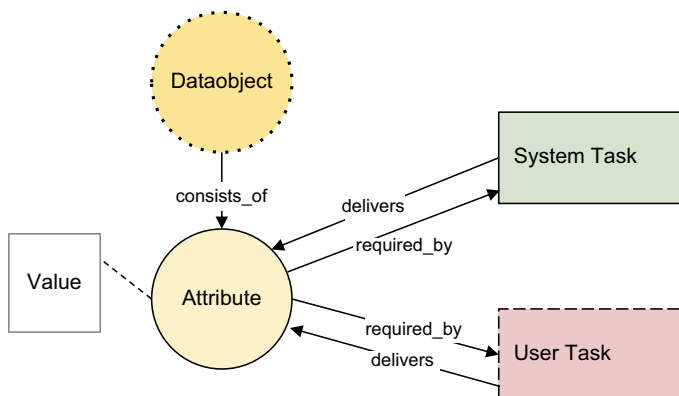


Abb. 13.10 Zentrale Konzepte in ODD-BP (Rietzke 2021)

und als Grundlage die Fallbasis bilden. Zum Berechnen ähnlicher Fälle existieren unterschiedliche Ähnlichkeitsmaße, die u. a. die Struktur des Wissensgraphen und somit die Semantik ausnutzen können, und verschiedene Suchalgorithmen. ProCAKE eignet sich somit zur Umsetzung einer System Task für die Notrufabfrage, die Attribute im Sinne von ODD-BP als Eingabe benutzt, um einen entsprechenden Output auch in Form von Attributen von Dataobjects zu generieren.

Notitia stellt zur Modellierung von Informationsmodell, Prozessen und Wissen entsprechende Editoren bereit. In diesem Zusammenhang sieht Notitia die Modellierung eines domänenspezifischen Informationsmodells wie auch von Prozessen unterschiedlicher Typen vor. Dabei umfasst Notitia auch Module zur Steuerung einer Prozessausführung. Damit anhand dieser Module ein mit dem methodischen Ansatz im Einklang stehendes Verhalten realisiert werden kann, bedarf es eines Modellierungsschemas, das die Anwendung von ODD-BP vorschreibt. Das Modellierungsschema muss dabei die vom methodischen Ansatz vorgegebenen Konzepte des künstlichen Mental Picture, Reconciliation und Entscheidungsfindung umsetzen. Abb. 13.11 zeigt einen Entwurf für dieses Modellierungsschema. Hierbei wird der initiale Informationsstand durch Dataobjects und Attribute realisiert. Zur Generierung des künstlichen Mental Picture besitzen die Attribute Input-Relationen zu entsprechenden System Tasks zur Integration von KI-Methoden. Als Output liefern diese System Tasks Attribute, welche Hinweise auf bestimmte entscheidungsrelevante Sachverhalte repräsentieren. Aus Gründen der Übersichtlichkeit wird in Abb. 13.11 nicht dargestellt, dass diese Output-Hinweise auch zu entsprechenden Dataobjects zugeordnet sind. Liefern die KI-Methoden einen Output, der zu im Konflikt stehenden Hinweisen führt, werden die jeweiligen Attribute mit einer weiteren System Task zur Umsetzung einer Reconciliation verknüpft. Das Output dieser System Task stellt nach dem methodischen Ansatz eine Erweiterung des künstlichen Mental Picture dar und wird ebenfalls durch ein Attribut im Informationsmodell abgebildet. Dieses Attribut geht schließlich in eine User Task zur Realisierung der Entscheidungsfindung ein, bei der ein Wissensarbeiter über den identifizierten Hinweis unterrichtet wird. Die Entscheidung des Wissensarbeiters als Reaktion auf einen erhaltenen Hinweis wird anschließend ebenfalls durch ein Attribut im Informationsmodell

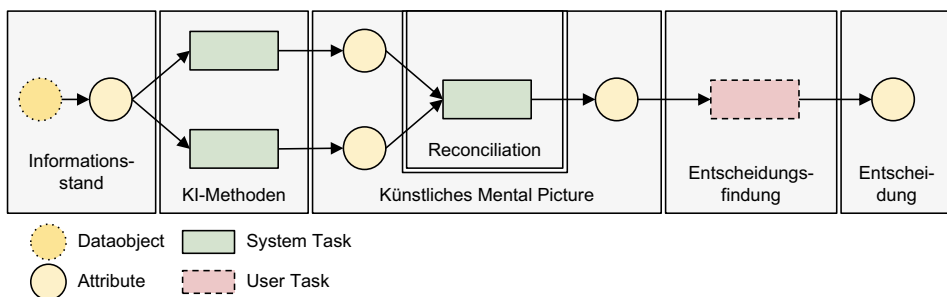


Abb. 13.11 Umsetzung des methodischen Ansatzes mit ODD-BP

abgebildet. Dieses Attribut kann dann mit weiteren Tasks verknüpft werden, die je nach getroffener Entscheidung bspw. relevante fehlende Informationen erheben oder konsequente Entscheidungsfindungen adressieren.

Nachfolgend wird anhand eines praktischen Beispiels skizziert, wie das vorgestellte Entwurfsmuster zur Umsetzung des methodischen Ansatzes in ODD-BP angewendet werden kann. Abb. 13.12 beschreibt hierfür einen beispielhaften Zustand einer durch ODD-BP repräsentierten Notrufabfrage. In diesem Prozess klagt ein Anrufer über intensiven Husten, was der Disponent durch die Ergänzung eines Boolean am entsprechenden Attribut des Patienten vermerkt. Der Wert dieses Attributes (zusammen mit den sonstigen relevanten Attributen) geht in die beiden System Tasks zur Integration des regelbasierten und fallbasierten Schließens ein. In diesem Zusammenhang erkennt das regelbasierte Schließen, dass ein Hinweis auf eine akute Covid-19-Erkrankung vorliegt. Im Gegensatz hierzu ermittelt der integrierte Service zur Umsetzung von fallbasiertem Schließen einen deutlichen Hinweis auf eine Covid-19-Erkrankung. Dieser deutliche Hinweis entsteht, da in der Fallbasis ein sehr ähnlicher Fall gefunden wurde, bei dem eine Covid-19-Erkrankung vorlag. Die beiden Attribute zur Repräsentation der erkannten Hinweise gehen als Teil des künstlichen Mental Picture in eine System Task zur Umsetzung der Reconciliation ein. Am Beispiel ist diese System Task durch einen regelbasierten Ansatz umgesetzt, der in der Lage ist, den vorliegenden Konflikt durch unterschiedlich deutliche Hinweise auf Covid-19 aufzulösen. Um die Patientensicherheit zu maximieren, sieht die hierbei implementierte Regel vor, dass der deutliche Hinweis zu priorisieren ist und dass dieser die Grundlage für eine Entscheidungsfindung darstellen soll. Entsprechend geht der bereinigte deutliche Hinweis auf Covid-19 in eine User Task zur Entscheidungsfindung ein. Die im Rahmen dieser User Task getroffene Entscheidung stellt schließlich den Output der Task dar. Am gezeigten Beispiel hat der Disponent durch eine Annahme des vorgeschlagenen Hinweises reagiert. Durch weitere, nicht in Abb. 13.12 dargestellte

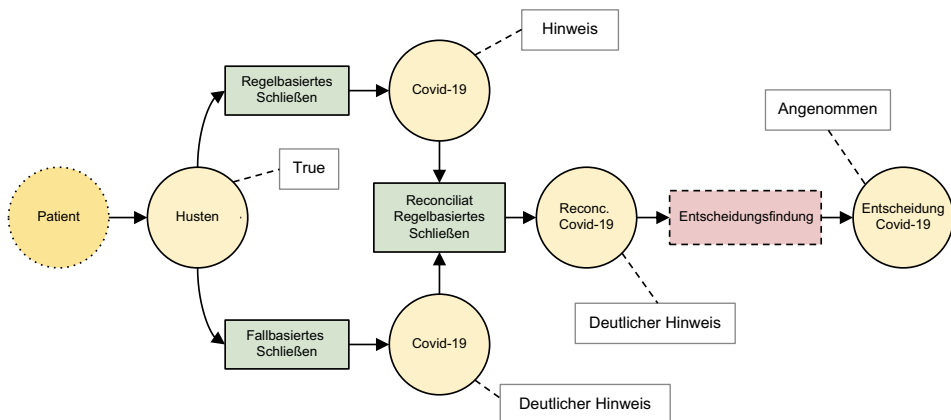


Abb. 13.12 Implementierung des methodischen Ansatzes für die Notrufabfrage

Verknüpfungen, kann diese Entscheidung die Grundlage für konsekutive Entscheidungen darstellen. Hierbei ist es beispielsweise möglich, dass weitere System Tasks zur Integration von KI-Methoden Hinweise darauf schlussfolgern, inwiefern der vorliegende Fall einer Hospitalisierung und einer entsprechenden Zuteilung von Notfallressourcen bedarf.

13.6 Empfehlungen

Um ein hybrides wissensbasiertes Reasoning für wissensintensive Prozesse effektiv anwenden und nutzen zu können, empfiehlt sich:

1. Aufbau eines von allen Verfahren gemeinsam zu nutzenden Informationsmodells, um die Differenzen im Mental Picture der beiden KI-Methoden zu minimieren,
2. Identifikation von formalisierbaren Regeln wie Gesetze, Richtlinien und Regeln zum Aufbau einer Grundlage durch das explizite Wissen,
3. Definition und Aufbau einer geeigneten Fallbasis, um implizites Wissen abzubilden, Entwicklung effektiver Ähnlichkeitsmaße und Adaptionenverfahren,
4. Sicherstellen einer Output-Kompatibilität, indem beide Verfahren auf dem identischen Informationsmodell ihren Output abbilden,
5. Ausarbeitung einer geeigneten Strategie zur Reconciliation, abhängig von den jeweiligen zu priorisierenden Charakteristiken, wie z. B. Patientensicherheit, Kostenminimierung, Zeitoptimierung,
6. Aufbau einer aussagekräftigen Visualisierung des Informationsstandes, damit auch das Mental Picture zwischen Mensch und Maschine bestmöglich für die Entscheidungsfindung angeglichen werden kann.

13.7 Fazit

Der vorgeschlagene hybride Ansatz wurde wie beschrieben im Kontext einer Notrufabfrage im Rahmen des Forschungsprojektes SPELL (gefördert durch das BMWK – FKZ: 01MK21005A) realisiert und wird derzeit weitreichend evaluiert. Dabei ist die Notrufabfrage lediglich ein Stellvertreter für einen wissensintensiven Prozess, der mit den klassischen Ansätzen der kontrollflussorientierten Geschäftsprozessmodellierung und -ausführung nur unzureichend umgesetzt werden kann. Grundsätzlich ist die hybride Methodik domänenunabhängig und eignet sich für KiPs, deren Wissensanteil aus expliziten Regeln und impliziten Erfahrungen zusammengesetzt ist. Da die gewählte Implementierung über skriptbasierte System Tasks jeglichen Dienst über eine entsprechende Schnittstelle ansprechen kann, lässt sich der Ansatz jedoch auch leicht auf weitere KI-Methoden ausweiten.

Mit dem Modellieren des regelbasierten Wissensanteils kann initial mit dem Aufbau einer Prozessumgebung begonnen werden, welche den bekannten und formalisierbaren expliziten Wissensanteil adressiert. Dabei wird jedoch grundlegend nicht von einer vollständigen Abbildbarkeit des Prozesswissens ausgegangen. Mit dem Aufbau einer Fallbasis und einer Strategie zur Bestimmung von ähnlichen abgeschlossenen Fällen wird der Grundstein gelegt, das nicht sichtbare implizite Erfahrungswissen im Prozesskontext nutzbar zu machen. Mit dem Ausführen der ersten Prozessinstanzen beginnt dann der Aufbau historischer Fälle, an den ein parallel laufender Lernprozess angeknüpft werden kann. Im Zuge eines Review-Prozesses können dann positive und negative Fälle identifiziert werden. Bei positiven Fällen kann dann die Fallbasis erweitert werden, sodass der Anteil des impliziten Erfahrungswissens steigt. Negative Fälle hingegen sind der Ausgangspunkt, um das explizite Regelwissen zu überarbeiten oder zu erweitern.

Im Zuge unserer weiteren Forschungsarbeiten werden wir weitere Domänen und Anwendungsfälle mit diesem hybriden KI-Verfahren adressieren, womit wir unsere Erwartungen an den domänenunabhängigen Charakter dieses methodischen Ansatzes weiter belegen möchten.

Literatur

- Aamodt, A. and Plaza, E. (1994) 'Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches', *AI Commun.*, 7(1), pp. 39–59.
- Baumann, A. (2017) 'Leitlinien oder Protokolle zur Notrufabfrage', in A. Hackstein, H. Sudowe (Eds.), *Handbuch Leitstelle*, Volume 2, pp. 189–193.
- Bergmann, R., Grumbach, L., Malburg, L. and Zeyen, C. (2019) 'ProCAKE: A Process-Oriented Case-Based Reasoning Framework'. ICCBR Workshops, pp. 156–161.
- Bergmann, R., Minor, M., Bach, K., Althoff, K.-D., and Munoz-Avila, H. (2021) 'Fallbasiertes Schließen', in Görz, G., Schmid, U., & Braun, T. (Eds.) 'Handbuch der Künstlichen Intelligenz', Volume 6, De Gruyter, Oldenburg.
- Blandford, A. and Wong B. L. W. (2004) 'Situation awareness in emergency medical dispatch', *International Journal of Human-Computer Studies*, 61(4), pp. 421–452, <https://doi.org/10.1016/j.ijhcs.2003.12.012>.
- Bruninghaus, S. (1994) 'DANIEL: Integrating case-based and rule-based reasoning in law', in Proceedings of the Twelfth National Conference on Artificial Intelligence, Menlo Park, CA: AAAI-Press, Student abstract.
- Buchanan, B. G. and Duda, R. O. (1983) 'Principles of Rule-Based Expert Systems', in: *Advances in Computers*, 22, Elsevier.
- Davenport, T. H. (2005) 'Thinking for a Living: How to Get Better Performances And Results from Knowledge Workers', Harvard Business School Press, Boston.
- Di Ciccio, C., Marrella, A., Russo, A. (2015) 'Knowledge-Intensive Processes: Characteristics, Requirements and Analysis of Contemporary Approaches', *Journal on Data Semantics*, 4, pp. 29–57, <https://doi.org/10.1007/s13740-014-0038-4>.
- Gates, L., Kisby, C. and Leake, D. (2019) 'CBR Confidence as a Basis for Confidence in Black Box Systems', in Proceedings of the International Conference on Case-Based Reasoning (ICCBR 2019), pp. 95–109.

- Golding, A. R. and Rosenbloom, P. S. (1996) 'Improving accuracy by combining rule-based and case-based reasoning', *Artificial Intelligence*, 87(1–2), pp. 215–254.
- Hackstein, A. (2017) 'Standardisierte Notrufabfrage', in A. Hackstein, H. Sudowe (Eds.), *Handbuch Leitstelle*, Volume 2, pp. 181–189.
- Lee M.-R. (2002). 'An Exception Handling of Rule-Based Reasoning Using Case-Based Reasoning', *J. Intell. Robotic Syst.*, 35(3), pp. 327–338.
- Maletzki, C., Rietzke, E., Grumbach, L., Bergmann, R. and Kuhn, N. (2019) 'Utilizing Ontology-Based Reasoning to Support the Execution of Knowledge-Intensive Processes', *Business Process Management Workshops*, pp. 32–44.
- Maletzki, C., Grumbach, L., Rietzke, E. and Bergmann, R. (2023) 'Towards Hybrid Intelligent Support Systems for Emergency Call Handling', in Martin, A., Hinkelmann, K., Fill, H.-G., Gerber, A., Lenat, D., Stolle, R., van Harmelen, F. (Eds.) *Proceedings of the AAAI 2023 Spring Symposium on Challenges Requiring the Combination of Machine Learning and Knowledge Engineering (AAAI-MAKE 2023)*.
- Mayr, B. (2020) 'Strukturierte bzw. standardisierte Notrufabfrage: Leisten die Systeme tatsächlich, was sie vorgeben zu leisten?', *Notfall und Rettungsmedizin*, 23, pp. 505–512.
- Møller, T. P., Jensen, H. G., Viereck, S., Lippert, F. and Østergaard, D. (2021) 'Medical dispatchers' perception of the interaction with the caller during emergency calls - a qualitative study'. *Scandinavian Journal of Trauma, Resuscitation and Emergency Medicine*, 29, <https://doi.org/10.1186/s13049-021-00860-y>.
- Nguyen, H. P., Prasad, N. R., Hung, D. H., and Drake J. T. (2001) 'Approach to combining case based reasoning with rule based reasoning for lung disease diagnosis', *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, Vancouver, BC, Canada, pp. 883–888, Volume 2, <https://doi.org/10.1109/NAFIPS.2001.944721>.
- Prentzas, J. and Hatzilygeroudis, I. (2007) 'Categorizing approaches combining rule-based and case-based reasoning', *Expert Syst. J. Knowl. Eng.* 24(2), pp. 97–122.
- Rietzke, E. (2021) 'Modellierung, Visualisierung und Ausführung wissensintensiver Prozesse unter Verwendung Semantischer Technologien', Dissertation, Universität Trier.
- Rietzke, E., Maletzki, C., Bergmann, R., Kuhn, N. (2021) 'Execution of Knowledge-Intensive Processes by Utilizing Ontology-Based Reasoning', *J. Data Semant.* 10(1-2), pp. 3-18.
- Saraiva, R.M., Bezerra, J., Perkusich, M.B., Almeida, H.O., & Siebra, C.D. (2015). A Hybrid Approach Using Case-Based Reasoning and Rule-Based Reasoning to Support Cancer Diagnosis: A Pilot Study. *Studies in health technology and informatics*, 216, pp. 862–866.
- Schultheis, A., Zeyen, C and Bergmann, R. (2023) 'An Overview and Comparison of Case-Based Reasoning Frameworks', in *Case-Based Reasoning Research and Development. ICCBR 2023. Lecture Notes in Computer Science* 14141, https://doi.org/10.1007/978-3-031-40177-0_21
- Sudowe, H. (2017) 'Notruf - der erste Kontakt zur Einsatzstelle: Wie funktioniert "handeln am Telefon"?'', in A. Hackstein, H. Sudowe (Eds.), *Handbuch Leitstelle*, Volume 2, pp. 175–181.
- Vaculín, R., Hull, R., Heath, T., Cochran, C., Nigam, A., Sukaviriya, P. (2011) 'Declarative business artifact centric modeling of decision and knowledge intensive business processes'. in *15th IEEE Int. Conf. on Enterprise Distr. Object Computing (EDOC 2011)*.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Optimierung der Entscheidungsfindung in autonomen Fahrsystemen mit neuro-symbolischem Wissen

14

Ya Wang , Kumar Manas  und Adrian Paschke 

Kernaussagen

1. Die Entwicklung einer semantischen Ontologie im Bereich des Straßenverkehrs ist für eine effektive Daten- und Wissensrepräsentation und das Erlernen von Regeln sowie das semantische Interpretieren und Schlussfolgern unerlässlich. Die Auswahl einer domänenspezifischen Ontologie, die mit bestehenden Fahrsimulations- und Teststandards kompatibel ist, wird als Startpunkt für die Ontologieentwicklung empfohlen.
2. Die Extraktion der strukturierten Daten von autonomen Fahrzeugen ist von entscheidender Bedeutung. Allerdings zeigen State-of-the-Art-Algorithmen zur Szenengraphgenerierung niedrige Erkennungsraten, was darauf hindeutet, dass sie noch nicht für den Einsatz in autonomen Fahrsystemen bereit sind, aber als zusätzliche Datenquelle zur Absicherung von Entscheidungen dienen können.

* indicates equal contribution

Y. Wang (✉) · K. Manas · A. Paschke
Freie Universität Berlin, Berlin, Deutschland
E-Mail: ya.wang@fokus.fraunhofer.de

K. Manas
E-Mail: kumar.manas@fu-berlin.de

A. Paschke
E-Mail: adrian.paschke@fokus.fraunhofer.de

Y. Wang · A. Paschke
Fraunhofer Institut für Offene Kommunikationssysteme FOKUS, Berlin, Deutschland

K. Manas
Continental AG, Hannover, Deutschland

© Der/die Autor(en) 2025

K. Hinkelmann et al. (Hrsg.), *Hybride KI mit Machine Learning und Knowledge Graphs*,
https://doi.org/10.1007/978-3-658-44781-6_14

319

3. Große Sprachmodelle können textuelle Verkehrs- und Fahrregeln in formale logische Regeln umwandeln. Diese Regeln sollen die Vorhersage und Planung von Trajektorien erleichtern. Allerdings sind die aktuellen Modelle noch nicht ausreichend entwickelt, um eine vollständig automatisierte Entscheidungsfindung zu unterstützen.
4. Durch die Nutzung der formalen Logikregeln kann ein Modul zur Trajektorienvorhersage und -planung sowohl die Genauigkeit als auch die Effizienz erheblich verbessern.
5. Neuronale induktive Logikprogrammierung bietet im Vergleich zu rein symbolischen Ansätzen mehr Vorteile bei der Verarbeitung von großen und verrauschten Daten. Für ein effektives Regellernen ist allerdings das Sammeln einer erheblichen Menge an Daten unerlässlich.

14.1 Einleitung

Im sich rasch entwickelnden Bereich des autonomen Fahrens ist es entscheidend, hohe Leistung, Erklärbarkeit und regelkonforme Entscheidungsfindung gleichzeitig zu erreichen, für Sicherheit, Nutzerakzeptanz und rechtliche Haftung der Hersteller. Ansätze auf Basis von Deep Learning haben ihre Wirksamkeit in verschiedenen Aufgaben unter Beweis gestellt, doch sie scheitern oft, wenn es um Interpretierbarkeit und Legalität geht (Wörmann et al. 2023). Im Gegensatz dazu haben regelbasierte Ansätze, die zusätzliche Erklärbarkeit und Nachvollziehbarkeit bieten, Schwierigkeiten, mit verrauschten, unsicheren und großen Daten in hochdynamischen Verkehrsszenen in einer Vielzahl von Szenarien umzugehen (Zhang et al. 2021). Um diese Probleme zu mildern, beschreibt dieses Kapitel einen hybriden neurosymbolischen Ansatz, der die Robustheit und Erklärbarkeit von regelbasierten Systemen mit der Anpassungsfähigkeit von maschinellen Lernmodellen synergisiert und so die Wirksamkeit und Zuverlässigkeit der Entscheidungsfindung in autonomen Fahrsystemen erhöht. In unserem Ansatz integrieren wir formalisiertes Welt- und Normwissen in Entscheidungsprozesse und nutzen Prädikatenlogik erster Stufe (FOL – First-Order Logic) und durch Algorithmen des induktiven logischen Programmierens (ILP) gelernte Regeln, um die Systemleistung zu steigern und ihre Erklärbarkeit zu verbessern. Zusätzlich werden verkehrsrechtliche Regeln in temporaler Logik für die regelkonformen Trajektorien als konkreter Anwendungsfall in Autonomous Driving (AD) formalisiert.

Abb. 14.1 ordnet dieses Kapitel thematisch ein bezüglich der Branche, Anwendung, Daten sowie Technologien.

Abb. 14.2 zeigt das Boxology-Modell¹ des hybriden KI-Systems. Bestehende Verkehrsregeln und Daten von On-board-Sensoren fließen als Datenbasis in das System ein.

¹ Für eine Einführung in die Boxology siehe Kap. 1.

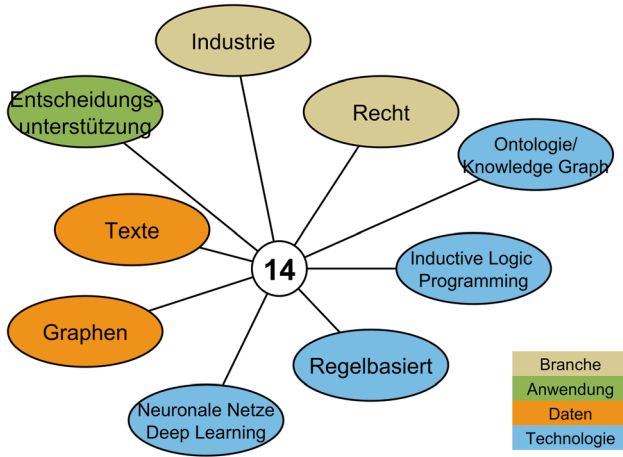


Abb. 14.1 Thematische Einordnung des Kap. 14

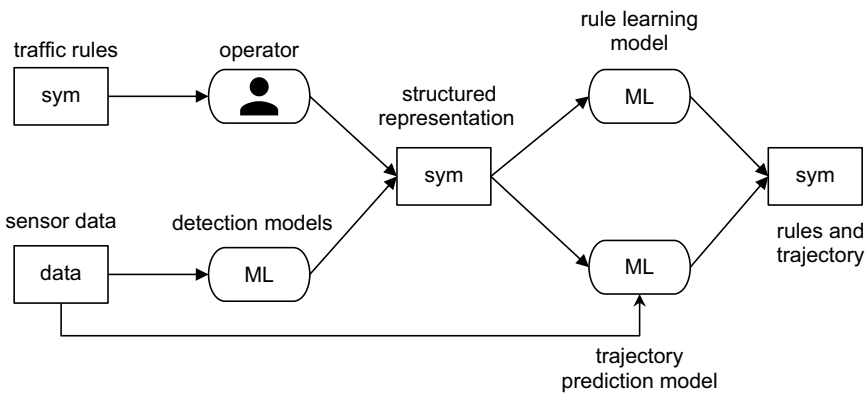


Abb. 14.2 Boxology-Modell des hybriden KI-Systems

Mithilfe verschiedener Detektionsmodelle und eines menschlichen Bedieners (Human in Loop) werden diese Informationen so strukturiert, dass sie sowohl maschinenverarbeitbar als auch für den Menschen verständlich sind. Maschinelle Lernmodelle nutzen dann diese strukturierte Repräsentation und die Sensordaten, um Klassifizierungsregeln zu lernen. In ähnlicher Weise werden bei der Vorhersage und Planung repräsentierte Regeln und Sensordaten verwendet, um regelkonforme Trajektorien vorherzusagen.

Wir strukturieren unser Kapitel wie folgt: Wir beginnen mit einem Überblick über unsere Softwarearchitektur, die mehrere Module umfasst, darunter Datenquelle, Wissensextraktion und -formalisierung, Regellernen für die Klassifikation und Regelintegration für Trajektorienvorhersage und -planung. Als Nächstes gehen wir auf den Auswahlprozess unserer Verkehrsontologie ein, die grundlegende Schicht für die

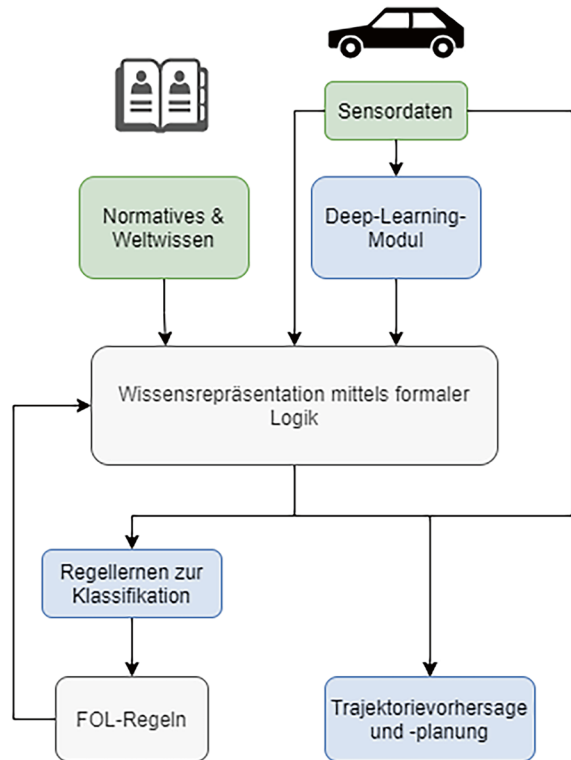
Wissensrepräsentation in unserem System. Darauf aufbauend diskutieren wir unsere Methoden zur Datenerhebung aus verschiedenen Fahrzeugquellen und die Techniken, die wir zur Darstellung dieser Daten als Prädikate verwenden. Anschließend beleuchten wir die verschiedenen Methoden, die wir für die Formalisierung von verkehrsrechtlichen Normen angenommen haben, und heben ihren einzigartigen Beitrag zum System hervor. Anschließend präsentieren wir zwei spezifische Anwendungen – Trajektorienvorhersage und Objektklassifikation –, die von der neuralsymbolischen Integration profitieren. Durch diese Anwendungen illustrieren wir, wie unser System die Erklärbarkeit erhöht und sicherstellt, dass die Entscheidungsfindung den gesetzlichen Normen entspricht. Abschließend heben wir die Stärken und Schwächen unseres Ansatzes hervor und geben Empfehlungen für die zukünftige Nutzung und Weiterentwicklung unserer Softwarearchitektur.

14.2 Softwarearchitektur

Die Entwicklung fortschrittlicher Rechenhardware und Algorithmen hat es modernen Fahrzeugen ermöglicht, mit zahlreichen Sensoren ausgestattet zu sein, die in der Lage sind, enorme Mengen an Daten aus der Fahrzeugumgebung zu erfassen. Zum Beispiel muss ein Fahrassistenzsystem der Stufe 4² etwa 100 TB Daten pro Tag verarbeiten (Grabowski und Wang 2023). Diese Daten sind oft unstrukturiert, heterogen und nicht sofort semantisch interpretierbar. Um diese Daten nutzen zu können, müssen sie in einem Format dargestellt werden, das sowohl für Menschen als auch für Maschinen verständlich ist. Wie in Abb. 14.3 dargestellt, werden Sensordaten, die innerhalb eines Fahrzeugs generiert und mit den Rechenergebnissen von On-board-Deep-Learning-Modulen kombiniert werden, mithilfe formaler Logik dargestellt, die auf der von uns entwickelten Straßenverkehrsontologie basiert. Darüber hinaus formalisieren wir normatives und weltliches Wissen, wie z. B. Verkehrsregeln, um die Entscheidungsfindung im Straßenverkehr zu unterstützen und eine Modulzertifizierung hinsichtlich der Konformität mit gesetzlichen und sicherheitsrelevanten Regeln zu ermöglichen. Dieses formalisierte Wissen wird dann in zwei spezifischen Anwendungsfällen eingesetzt: bei der Klassifizierung des Objekts innerhalb der Verkehrsszene und bei der Vorhersage der Fahrzeugtrajektorie auf Basis dieses Wissens. Erstens kann ILP – eine Art von Regellernalgorithmus – verwendet werden, um Regeln der FOL auf dem Wissensgraphen zu lernen. Diese Logikregeln können, wenn sie mit vorherigem Wissen integriert werden, nicht nur die Funktionalität von Deep-Learning-Modellen verbessern, sondern auch die Erklärbarkeit und Robustheit der Entscheidungsfindung gewährleisten, indem sie eine hochrangige Schlussfolgerung innerhalb autonomer Fahrsysteme erleichtern. Zweitens können

²Fahrassistenzsystem der Stufe 4: Ein System, bei dem das Fahrzeug in definierten Situationen autonom operiert. Menschliche Intervention kann gegebenenfalls erforderlich sein.

Abb. 14.3 Überblick über die Softwarearchitektur



formalisierte Verkehrsregeln als Einschränkungen in das Modul zur Entscheidungsfindung im Straßenverkehr implementiert werden, was die Vorhersage des potenziellen Verhaltens von Straßenverkehrsteilnehmern unterstützt. Beispielsweise können Verkehrsregeln mit einem neuronalen Netz kombiniert werden, um die Trajektorie eines Fahrzeugs vorherzusagen; diese Integration ermöglicht es, datengesteuerte Vorteile mit Fahrregeln als Einschränkungen für die Vorhersage zu verbinden. Der folgende Abschnitt enthält weitere Einzelheiten zu den Technologie- und Implementierungsaspekten dieser beiden Anwendungsfälle.

14.3 Technologien und Implementierung

14.3.1 Ontologie für den Straßenverkehr

Der Begriff „Ontologie“ stammt ursprünglich aus der Philosophie, wo er sich auf die Untersuchung der Natur von Existenz und der Realität bezieht, einschließlich der Beziehungen zwischen verschiedenen Entitäten und Kategorien von Dingen (Quine 1969). In der Informatik wird Ontologie als eine formale Spezifikation eines gemeinsamen

Vokabulars für einen bestimmten Bereich definiert, das auch die Definitionen von Konzepten, Kategorien und Beziehungen zwischen ihnen enthält (Gruber 1993; Guarino 1995). Der Zweck der Erstellung einer Ontologie besteht darin, ein standardisiertes, gemeinsames Verständnis einer Domäne bereitzustellen, das von verschiedenen Personen oder Computersystemen zur Kommunikation, zum Wissensaustausch und zur Problemlösung verwendet werden kann (Studer et al. 1998). Ontologien werden häufig mithilfe der Beschreibungslogik (DL) repräsentiert, einer Untermenge der FOL, die nur unäre und binäre Beziehungen verwendet (Baader et al. 2007). Durch die Einschränkung der Ausdruckskraft der FOL erreicht die DL ein gutes Gleichgewicht zwischen Ausdruckskraft und einfacher rechnerischer Verarbeitung, was effiziente Schlussfolgerungen und Entscheidungsverfahren ermöglicht (Horrocks 2008). Im Bereich des Straßenverkehrs gibt es eine Vielzahl von Begriffen und Definitionen, die oft vage oder sogar widersprüchlich sind. Es wurde bereits viel Arbeit geleistet, um diese Begriffe für eine einheitliche Darstellung des Verkehrsgeschehens zusammenzufassen, zu konsolidieren und zu harmonisieren (Westhofen et al. 2022). Beispielsweise geben Ulbrich et al. (2015) eine klare Definition der Begriffe Szene, Situation und Szenario für das Testen und die Simulation fortschrittlicher Fahrerassistenzsysteme. Scholtes et al. (2021) schlagen ein 6-Ebenen-Modell für die Darstellung einer komplex strukturierten Verkehrsumgebung vor. Maierhofer et al. (2022) formalisierten teilweise maschinenlesbare deutsche Kreuzungsvorschriften in temporaler Logik für autonome Fahrzeuge. Darüber hinaus hat die Association for Standardisation of Automation and Measuring Systems (ASAM) im Jahr 2020 ein Projekt zur Entwicklung einer Ontologie initiiert, die einen gemeinsamen Satz von Definitionen, Eigenschaften und Beziehungen für wesentliche Konzepte im Bereich des Straßenverkehrs festlegt. Für unsere Softwarearchitektur haben wir die von ASAM entwickelte OpenX-Ontologie (Tao et al. 2021) als Grundlage gewählt, da sie mit weitverbreiteten Fahrsimulations- und Teststandards wie OpenDRIVE, OpenSCENARIO und OpenLABEL kompatibel ist, die in Plattformen wie Carla verwendet werden. Die OpenX-Ontologie besteht aus drei Modulen, die sich in ihrem Allgemeingrad unterscheiden: die Kernontologie, die Domänenontologie und die Anwendungsontologie. Die Kernontologie, die unabhängig von einer bestimmten Domäne ist, definiert grundlegende Konzepte wie physische Objekte, Zustände und Ereignisse, die mithilfe des High-Quality Data Model Framework (HQDM) entwickelt wurden. Die Domänenontologie definiert wesentliche Konzepte, die spezifisch für die Straßenverkehrsdomäne sind, und ist in drei Schichten gegliedert: EnvironmentalCondition, RoadTopologyAndTrafficInfrastructure und TrafficParticipantAndBehavior. Die Anwendungsontologie schließlich umfasst Konzepte, die sich auf eine bestimmte Anwendung beziehen, wie z. B. ElapsedTime in einer Simulation. Die OpenX-Ontologie besteht aus 347 Klassen, 96 Objekteigenschaften und 2 Dateneigenschaften und verwendet OWL (World Wide Web Consortium [W3C], 2004) als Ontologiesprache und Semantic Web Rule Language (SWRL; Horrocks et al. 2004) als Regelsprache. Für unseren speziellen Bedarf haben wir einen relevanten Teilbereich extrahiert und ihn mit zusätzlichen Konzepten und Relationen ergänzt, insbesondere solche aus rechtlicher Perspektive.

14.3.2 Strukturierte Darstellung und Extraktion von Fahrdaten

Der Wissensgraph wurde erstmals 2012 von Google eingeführt und ist seitdem ein prominentes Konzept in der Produkt- und Dienstleistungsbranche geworden. Im Wesentlichen ist ein Wissensgraph eine Art der Wissensrepräsentation, die Entitäten, Attribute und Beziehungen verwendet. Entitäten werden durch Knoten repräsentiert, Attribute durch Beschriftungen oder Eigenschaften und Beziehungen durch Kanten. Im Vergleich zur traditionellen Wissensbasis besteht ein Wissensgraph typischerweise nur aus einer Behauptungsbox (ABox), die eine Reihe von Faktenaussagen enthält, ohne eine separate terminologische Box (TBox; Gruber et al. 1993) zur Definition von Terminologie. Zusätzlich kann ein Wissensgraph eine Inferenzmaschine enthalten, die zur Ableitung neuer Fakten oder zur Beantwortung von Fragen über den Wissensgraphen genutzt werden kann. In unserer Implementierung dient die Ontologie als TBox für eine umfassende und einheitliche Repräsentation des Straßenverkehrs, die die Erzeugung des Flusses von Wissensgraphen (d. h. ABox) ermöglicht. Wir verarbeiten Rohdaten, die von Fahrzeugen gesammelt wurden, und nutzen sie, um unseren aktuellen Wissensgraphen zu aktualisieren, indem wir sie auf die ABox abbilden. Ergänzend dazu werden auch Rechenergebnisse aus verschiedenen bordeigenen Deep-Learning-Modulen in diese Graphen integriert. Diese Praxis bildet einen entscheidenden Teil der Wissenserfassung, den ersten Schritt in Techniken des Wissensgraphen, die strukturierte Informationen aus verschiedenen Quellen extrahieren. Im Bereich des Straßenverkehrs stammen solche Daten aus verschiedenen Quellen. Dazu gehören bordeigene Sensoren wie Kameras, Lidar und Radar, aber auch externe Informationsanbieter wie High-Definition-Karten (HD-Karten), GPS, GNSS und V2V-Kommunikationen. Weiterhin tragen Electronic Control Units (ECUs) zu diesem Datenpool bei, indem sie priorisierte Nachrichten über Fahrzeugkommunikationsnetzwerke wie das Controller Area Network (CAN) übertragen. Diese Nachrichten sind zugänglich und können von anderen Geräten, die mit dem Netzwerk verbunden sind, verarbeitet werden. Daten, die eine semantische Bedeutung haben, wie GPS, können direkt auf die ABox abgebildet werden. In einer Simulationseinstellung wie Carla können wir Daten über ihre API extrahieren und die OpenDrive- und OpenScenarios-Dateien parsen, um mehr Informationen über die Straßeninfrastruktur und andere Verkehrsteilnehmer in der Fahrumgebung zu erhalten. Da wir in der frühen Phase eine verfügbare Ontologie haben, wenden wir die Top-down-Methodik, wie sie in (Fensel et al. 2020) beschrieben ist, zur Generierung von Daten in der ABox an. In diesem Schritt mappen wir eingehende Daten mit vordefinierten Mapping-Regeln und erweitern die Ontologie bei Bedarf, um neue Konzepte aufzunehmen, die aus den Daten entstehen können. Um mehr über die Erweiterung unserer Ontologie zu erfahren, verweisen wir auf unsere Arbeit (Wang et al. 2022) für zusätzliche Details. Es gibt auch mehrere bestehende Tools und Techniken, die entwickelt wurden, um das Mapping von strukturierten oder halbstrukturierten Daten in RDF zu unterstützen, das ein bevorzugtes Format der Darstellung von Wissensgraphen ist. Ein solches Tool ist XMLtoRDF (Van Deursen et al. 2008), das ein Mapping-Dokument im XML-Format produziert, das ein XML-

Schema mit einer OWL-Ontologie verlinkt. Ein weiteres Tool, Tripliser (David Rogers 2011), ist eine Java-Bibliothek und ein Kommandozeilen-Tool, das Triple-Graphen aus XML-Daten erstellen kann. GRDDL (W3C 2007) ist ein weiterer Ansatz, der XML-Daten in RDF übersetzt, während Virtuoso Sponger (OpenLink 2023) ein Tool ist, das Linked Data aus verschiedenen Datenquellen generieren kann und eine Reihe von Datenrepräsentations- und Serialisierungsformaten unterstützt. R2RML ist eine Spezifikation, die definiert, wie relationale Daten in RDF übersetzt werden, und RML (Dimou et al. 2014) erweitert die Anwendbarkeit von R2RML (W3C 2012), um Mappings für Daten in anderen Formaten zu definieren. In unserer Implementierung nutzen wir OWLready2 (Sorbonne Paris North University 2023) und RDFLib (RDFLib Team 2023) die Python-Pakete, die für die ontologieorientierte Programmierung konzipiert sind. Mit einem umfassenden Satz von Funktionen bieten sie uns eine flexible Möglichkeit, effiziente Mapping-Skripte zu entwickeln, die Daten auf die ABox mappen können.

Wenn es um Daten geht, die nicht direkt interpretierbar sind – wie Kamera- und Lidar-Daten –, verlassen wir uns auf Deep-Learning-Modelle, um vorverarbeitete Ergebnisse zu erzeugen, die anschließend zur weiteren Analyse auf die Ontologie abgebildet werden. Es ist zu bemerken, dass Kameras unter den verschiedenen Datenquellen, die in autonome Fahrsysteme einfließen, eine besondere Rolle spielen und ein Gleichgewicht zwischen zuverlässiger Leistung und angemessenem Preis bieten. Die von Kameras erfassten Bilder können zur Erstellung eines visuell fundierten Szenengraphen verwendet werden, der eine wertvolle Repräsentation der Verkehrsszene liefert. Dieser Prozess, der als Scene Graph Generation (SGG) bezeichnet wird, ist entscheidend, um aussagekräftige Erkenntnisse aus den Bildern zu gewinnen. Der Prozess zur Erzeugung von Graphen beinhaltet drei wesentliche Erkennungsaufgaben. Die erste besteht darin, Instanzen von Objekten wie Menschen, Fahrzeuge und Teile anderer Objekte zu erkennen. Die zweite Aufgabe besteht darin, Attribute zu identifizieren, die den Zustand des aktuellen Objekts beschreiben, wie seine Form, Farbe und Haltung. Die dritte Aufgabe besteht darin, die Beziehung zu finden, die die Verbindung zwischen Paaren von Objekten beschreibt, wie ihre Handlungen und Positionen. Diese Attribute und Beziehungen bilden die Grundlage des Graphen und können als <Subjekt-Prädikat-Objekt>-Triplette ausgedrückt werden. Zum Beispiel kann eine Szene, die beschreibt, dass ein Fußgänger sich bewegt, als <Fußgänger, bewegt sich, nil> (wobei ‚nil‘ ‚nicht existent‘ bedeutet) ausgedrückt werden und ein Fahrzeug fährt auf einer Straße als <Fahrzeug, fährt auf, Straße>. Diese Ausdrücke können leicht ins RDF-Format übersetzt und anschließend auf die ABox abgebildet werden.

Die populäre Detektionsarchitektur ist grob in zwei Arten unterteilt. Die erste Art beinhaltet zwei Stufen: Objekterkennung mit einem vortrainierten Objekterkennungsmodell wie Faster RCNN und paarweise Beziehungserkennung basierend auf den erkannten Merkmalen der beiden Objekte oder ihren Klassenprognosen. Die andere Art beinhaltet das gemeinsame Erkennen und Erkennen der Objekte und ihrer Beziehungen. Beachten Sie, dass die Experimente von Hen et al. (2021) gezeigt haben, dass das Modell, das die Objekt- und Beziehungserkennungsmodule trennt und nicht denselben Backbone teilt, besser abschneidet als das gekoppelte Modell. Die Erkennung

von Beziehungen ist die Schlüsselaufgabe bei der Generierung von Graphen und kann durch die Integration verschiedener Arten von Vorwissen verbessert werden, einschließlich Sprachpriorität, statistischer Priorität und allgemeiner Wissenspriorität. Die Verwendung von Sprachprioritäten wurde von Lu et al. (2016) eingeführt, die semantische Wörter verwenden, um die Genauigkeit der Beziehungsvorhersage zu verbessern. Sprachprioritäten unterstützen die Erkennung visueller Beziehungen, indem sie semantisch verwandte Objekte beobachten. Pferde und Elefanten können beide von Menschen geritten werden, wie in den ähnlichen Phrasen „ein Mensch reitet ein Pferd“ und „ein Mensch reitet einen Elefanten“ zu sehen ist. Durch die Verwendung von Sprachprioritäten und das Studium häufiger Beispiele, wie „ein Mensch reitet ein Pferd“, können wir trotzdem die Beziehung zwischen einem Menschen und einem Elefanten ableiten, ohne dass eine häufige Koexistenz im Trainingssatz erforderlich ist. Die statistische Priorität ist eine Art von Vorwissen, das häufig in SGG verwendet wird, da Objekte in visuellen Szenen typischerweise eine starke strukturelle Regularität aufweisen (Zellers et al. 2018). Zum Beispiel ist es typisch, dass Menschen Schuhe tragen und Berge von Wasser umgeben sind. Beachten Sie, dass die Frequenzpriorität sehr effektiv bei der Vorhersage der Beziehungsklassifikation ist, wenn die Objekt- und Subjektlabels gegeben sind (Chang et al. 2021). Darüber hinaus können allgemeine Wissensgraphen bei der Erkennung von Beziehungen helfen. GB-Net (Graph Bridging Network) kombiniert Szenen- und Wissensgraphen in einem einzigen Framework und definiert die Szenengraphenerzeugung als Brückenmapping zwischen den beiden (Zareian et al. 2020). Die Evaluation der visuellen Beziehungserkennung beinhaltet typischerweise die folgenden Methoden:

- Prädikaterkennung, die erforderlich ist, um mögliche Prädikate zwischen Objektpaaren vorherzusagen, ohne durch die Objekterkennung eingeschränkt zu sein. Das Ziel ist es, die Leistung der Beziehungsvorhersage zu studieren (Girshick et al. 2013).
- Phrasenerkennung, die darin besteht, ein <Subjekt-Prädikat-Objekt>-Triplet für ein gegebenes Bild vorherzusagen und einen Begrenzungsrahmen für die gesamte Beziehung zu lokalisieren (Sadeghi und Farhadi 2011).

Gängige Datensätze, die für Benchmark-Aufgaben zur Erzeugung von Szenengraphen verwendet werden, sind Visual Genome (VG; Krishna et al. 2016), Open Images (Google 2023) und Visual Relationship Dataset (VRD; Lu et al. 2016). Unter diesen ist Visual Genome das Größte, bestehend aus 108 K Proben, 33,8 K Objektkategorien und 42 K Objektbeziehungen. Zum jetzigen Zeitpunkt ist das bestperformende Modell auf dem Visual Genome Dataset Causal-TDE (Tang et al. 2020), das zwischen guten und schlechten Vorurteilen unterscheidet. Weitere beliebte Algorithmen für die Erzeugung von Szenengraphen sind Iterated Message Passing (IMP; Xu et al. 2017), Multilevel Scene Description Networks (MSDN; Li et al. 2017), Graph R-CNN (Yang et al. 2018), Neural Motif (NM; Zellers et al. 2018) und ReIDN (Zhang et al. 2019). Es ist jedoch zu beachten, dass alle diese Algorithmen niedrige Recall-Raten aufweisen, die auf VG-Datensätzen

unter 50 % liegen. Die Hauptherausforderungen bei der Erzeugung von Szenengraphen ergeben sich aus verschiedenen Faktoren, einschließlich Langschwanzverteilungen in visuellen Beziehungen, Erkennung von Beziehungen zwischen entfernten Objekten, Erzeugung von Szenengraphen basierend auf dynamischen Bildern, Erkennung sozialer Beziehungen basierend auf Szenengraphen und Durchführung visueller Schlussfolgerungen (Chang et al. 2021). Angesichts des aktuellen Standes der Technik zur Erzeugung von Szenengraphen ist ihre Reife allein nicht ausreichend für autonome Fahrsysteme. Daher ist es notwendig, ihre Ergebnisse mit mehreren Fahrzeugdatenquellen zu überprüfen, um das Risiko von verpassten Erkennungen oder Ungenauigkeiten zu minimieren.

14.3.3 Formalisierung gesetzlicher Verkehrsregeln

Gesetzestexte und Weltwissen über Fahr- und Verkehrsregeln können in einer maschineninterpretierbaren logischen Form formalisiert werden. Es gibt mehrere Möglichkeiten, Regeln in logischer Form zu formalisieren. FOL (Karimi und Duggirala 2020), metrische temporale Logik (MTL; Maierhofer et al. 2020), lineare temporale Logik (LTL) und Signal-temporale Logik (STL; Li et al. 2021) sind einige davon. Da das Autofahren ein zeitliches Ereignis ist, bei dem vergangene Handlungen zukünftige Handlungen beeinflussen, ist die temporale Logik die aktuellste Wahl für die Regelformalisierung. Außerdem wird die temporale Logik häufig in Verifikationssystemen verwendet, um festzustellen, ob das System der Spezifikation entspricht. In Maierhofer et al. (2022) und Maierhofer et al. (2020) werden Verkehrsregeln für das Fahren auf Autobahnen und Kreuzungen in MTL formalisiert. Karimi und Duggirala (2020) formalisierten die Verkehrsregeln für unkontrollierte Kreuzungen in FOL. Die manuelle Regelformulierung erfordert jedoch Erfahrung mit Logik, Domäne und technologischem Know-how. Die Formalisierung von Regeln kann die Sicherheit beim autonomen Fahren erhöhen, da sie zur Überprüfung der von Algorithmen generierten Trajektorien verwendet werden kann.

Zunächst erläutern wir den Prozess der Erstellung der Formalisierung von Rechtsregeln, gefolgt von ihrer Verwendung bei der Überprüfung von Trajektorien. Wie in Abb. 14.4 dargestellt, müssen wir zunächst die Regeln und das Expertenwissen, das wir formalisieren wollen, anwendungsbezogen sammeln. Nehmen wir zum Beispiel an, ein Entwickler möchte überprüfen oder sicherstellen, dass das autonome Fahren die Überholregel korrekt befolgt. In diesem Fall wählen wir nur relevante Regeln durch Vorverarbeitung aus, um andere, nicht notwendige Regelformalisierung zu vermeiden und Zeit und Rechenressourcen zu sparen. Verkehrsregelquellen wie die Straßenverkehrsordnung (StVO)³ und das California Driver Handbook⁴ können gute Quellen sein. Nachdem wir

³https://www.gesetze-im-internet.de/stvo_2013

⁴https://www.dmv.ca.gov/web/eng_pdf/dl600.pdf

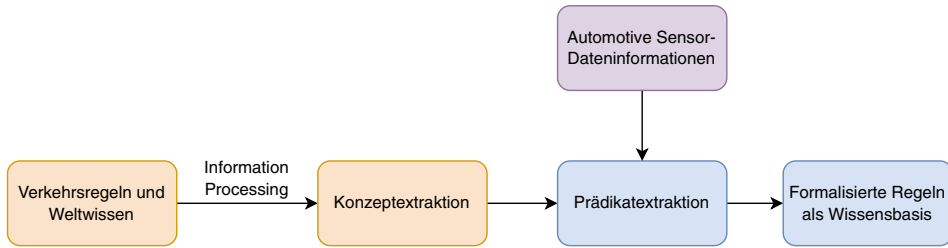


Abb. 14.4 Illustration der Schritte der Regelformalisierung im Kontext des autonomen Fahrens

das Wissen von Experten und Rechtsquellen gesammelt haben, müssen wir wichtige Zusammenhänge aus dem Text extrahieren und ihn konkretisieren. Zum Beispiel heißt es in der StVO § 4(1): „Der **Abstand** zu einem vorausfahrenden Fahrzeug muss in der Regel **so groß** sein, dass auch dann hinter diesem gehalten werden kann, wenn es plötzlich gebremst wird. Wer vorausfährt, darf nicht ohne zwingenden Grund stark bremsen.“ Im Gegensatz zu Menschen brauchen Maschinen präzise und oft mathematische Interpretationen von Begriffen wie Sicherheitsabstand. Begriffe wie Sicherheitsabstand können mit Funktionen und Argumenten wie Geschwindigkeit, Beschleunigung und Reaktionszeit konkretisiert werden.

In diesem Abschnitt wird beschrieben, wie Regeln in MTL formalisiert werden, aber derselbe Ansatz kann auch auf andere Arten von Logik angewendet werden, je nach den Bedürfnissen und Anwendungsfällen des Benutzers. Wir werden Verkehrsregeln aus Ländern mit Rechtsverkehr als Beispiel verwenden, aber die Regeln können mit geringfügigen Änderungen leicht für Länder mit Linksverkehr angepasst werden. Nehmen wir zum Beispiel die Verkehrsregel: „Ego-Fahrzeuge dürfen nicht schneller fahren als Fahrzeuge auf der linken Spur.“ Diese Regel hat einige zusätzliche Bedingungen, aber der Einfachheit halber überspringen wir diese Teile der Regel und erklären, wie diese Regel auf der Grundlage der temporalen Logik modelliert werden kann, wie sie in Maierhofer et al. (2020) eingeführt wurde. Ego-Fahrzeuge sind Fahrzeuge, die automatisiert fahren und deren Verhalten für uns von primärem Interesse ist.

Regel: Ego-Fahrzeuge dürfen nicht schneller fahren als Fahrzeuge auf der linken Spur.

Formalisierung in MTL:

$$G(\text{left_of}(\text{otherVehicle}, \text{egoVehicle}) \wedge \neg \text{drives_faster}(\text{egoVehicle}, \text{otherVehicle}))$$

In dieser Gleichung prüft das Prädikat *left_of*, ob sich ein anderes Fahrzeug links vom Ego befindet, und das Prädikat *drives_faster* prüft, ob das Ego-Fahrzeug schneller ist als das andere Fahrzeug. *G* ist ein globaler Operator der MTL mit temporaler Natur, der erzwingt, dass diese Regel für alle Zeitzustände während des Fahrens gelten sollte. Der Leser kann sich auf Thati und Roşu (2005) beziehen, um mehr über temporale Logik und ihre Verwendung in cyber-physischen Systemen zu erfahren. Anhand der

Prädikatsauswertung stellen wir fest, ob die Regel erfüllt ist. Dies zeigt die Bedeutung des Prädikats und wir müssen dem Prädikat in den formalisierten Verkehrsregeln für die Bewertung eine Bedeutung geben. Um die Lücke zwischen der formalisierten Verkehrsregel und ihrer Bewertung zu schließen, müssen wir mit begrenzten Informationen arbeiten, die von den Sensoren und Karteninformationen des automatisierten Fahrens bereitgestellt werden. Diese begrenzten Daten können Position, Geschwindigkeit, Beschleunigung, Richtungswinkel, Straßeninformationen wie Fahrbahnmarkierungen, Verkehrsinfrastruktur wie Ampeln, Geschwindigkeitsbegrenzungen und andere Informationen umfassen, die auf den verwendeten Sensor- und Karteninformationen basieren. In der oben formulierten Regel kann *left_of* anhand von Sensordaten ausgewertet werden, indem die aktuelle Fahrspur-ID der Fahrzeuge abgefragt wird, aber *drives_faster* kann nicht direkt anhand von Sensor- und Kartendaten ausgewertet werden. Daher müssen wir zusätzliche Funktionen verwenden, um die Lücke zwischen dem Prädikat *drives_faster* und den Sensordaten zu schließen. Zur Auswertung von *drives_faster* können wir die Geschwindigkeit als Sensordatenparameter verwenden, um festzustellen, ob das Ego-Fahrzeug schneller als die anderen Fahrzeuge fährt oder nicht, wie folgt:

$$\text{drives_faster}(\text{egoVehicle}, \text{otherVehicle}) \Leftrightarrow \text{proj}_v(\text{egoVehicle}) > \text{proj}_v(\text{otherVehicle})$$

In der obigen Gleichung wird überprüft, ob das Ego-Fahrzeug eine höhere Geschwindigkeit hat als das andere Fahrzeug. $\text{proj}_v(\text{egoVehicle})$ und $\text{proj}_v(\text{otherVehicle})$ sind die Funktionen, die die Geschwindigkeit des Ego- bzw. des anderen Fahrzeugs projizieren. Anhand dieses Vergleichs können wir feststellen, welches Fahrzeug schneller fährt. In ähnlicher Weise können auch andere Funktionen und Prädikate definiert werden, um nicht direkt aus Sensoren und HD-Karten extrahierbare Informationen zu nutzen. Einige Prädikate sind komplexer; wir nennen sie Metapredikate und sie bestehen aus einer Kombination anderer Prädikate und Funktionen. Interessierte Leser können in Maierhofer et al. (2022, 2020), Karimi und Duggirala (2020) weitere Details über die Erstellung von Funktionen und Prädikaten zur Formalisierung von gesetzlichen Verkehrsregeln nachlesen. Diese Literatur enthält zahlreiche Regeln und Richtlinien für die Erstellung logischer Formalisierungen.

Dies ist der klassische Weg, um formale logikbasierte Regeln zu erstellen. Eine solche Formalisierungsmethodik erfordert Fachwissen sowohl in der Logik als auch in der Domäne und behindert manchmal die Skalierbarkeit und Anpassungsfähigkeit von formalen Logiksystemen. Als Nächstes untersuchen wir die Fähigkeit großer Sprachmodelle (LLMs) wie BERT (Devlin et al. 2019), T5 (Raffel et al. 2020) und GPT-3 (Brown et al. 2020) zur automatischen Wissensdarstellung von Verkehrsregeln.

14.3.4 Regelformalisierung und große Sprachmodelle

Aktuelle Fortschritte im Bereich der Sprachmodellierung führen zu hochmodernen Ergebnissen bei Sprachübersetzungs- und Inferenzaufgaben (Raffel et al. 2020). Sie

fallen in der Regel unter den Begriff große Sprachmodelle oder Large Language Models (LLMs) und neuere Modelle folgen der Transformer-Architektur (Vaswani et al. 2017). Wir führen die Leser in LLMs und ihre Anwendungsfälle in der Regelformalisierung ein. Wir können uns die Umwandlung von natürlichem Text in formale Logik als eine Sprachübersetzungsaufgabe vorstellen, z. B. die Übersetzung vom Englischen ins Deutsche. Wir werden kurz die Funktionsweise eines Sprachmodells und zwei breit angelegte Ansätze zum Fine-Tuning und Prompting erläutern, die in der Sprachübersetzung für eine neue Reihe von Übersetzungsaufgaben verwendet werden.

LLMs sind auf Deep Learning basierende Modelle, die auf der Grundlage des aus Datensätzen gewonnenen Wissens Texte zusammenfassen, übersetzen, das nächste Wort vorhersagen und Texte generieren. Diese Modelle werden so trainiert, dass sie die wahrscheinlichsten Wörter vorhersagen, und zwar auf der Grundlage der bedingten Wahrscheinlichkeitsverteilung unter Berücksichtigung des Kontexts und der Wörter, die bereits vor ihnen erschienen sind. Die Modelle berechnen die Wahrscheinlichkeit jedes möglichen Wortes, das als Nächstes kommen könnte, und bewerten sie. Auf der Grundlage dieser Wahrscheinlichkeitsbewertungen werden neue und wahrscheinlichste Wörter ausgewählt und diese Auswahl von Wörtern ist das Wesen von LLMs bei Übersetzungsaufgaben. Das Vortraining von Sprachmodellen erfolgt auf einem großen Korpus von Online-Daten, wie z. B. Wikipedia, sodass das Modell die Sprache und die statistische Verteilung der Wörter verstehen kann. Es gibt zwei Methoden, um vortrainierte Allzweckmodelle für unseren speziellen Anwendungsfall der Übersetzung von natürlich-sprachlichen Verkehrsregeln in formale Logik zu nutzen:

- **Fine-Tuning:** Bei einem Fine-Tuning-Ansatz wird ein zuvor trainiertes Sprachmodell auf einem neuen Datensatz weiter trainiert, der kleiner ist und auf Anwendungsfälle ausgerichtet ist. Bei diesem Datensatz kann es sich um eine Sammlung von Texten in natürlicher Sprache und entsprechenden formalen logischen Darstellungen handeln. Diese kleinen, aber kuratierten Daten ermöglichen es dem Modell, neue Sprachübersetzungsaufgaben (hier die Übersetzung von Text in formale Logik) zu verstehen und die Modellparameter zu aktualisieren, um eine neue Sprachstruktur zu lernen. *Hahn et al. (2022)* verwendeten eine Fine-Tuning-Methode für die Übersetzung von Text in natürlicher Sprache in FOL und reguläre Ausdrücke (Regex).
- **Prompting:** Beim Prompting wird dem Modell die Anweisung in natürlicher Sprache gegeben, gefolgt von null oder mehr Beispielen für die gewünschte Aufgabe. Dies wird auch als „few-shot learning“ bezeichnet. Der Begriff "shot" bezieht sich auf die Anzahl der Beispiele, die dem Modell neben der Anweisung gegeben werden. In unserem Anwendungsfall kann das Prompting aus der Anweisung: „Übersetze den folgenden Satz in eine metrische temporale Logik“, mit Input-Output-Paaren in Form von Text in natürlicher Sprache und seiner temporalen logischen Repräsentation bestehen. Wir können die temporale Logik durch Beispiele aus der FOL oder andere logische Formen für unser Prompting ersetzen. Auf diese Weise kann das Modell schnell verstehen, wie es sich an neue Aufgaben anpassen kann, da es bereits während des Vortrainings eine Vorstellung von Sprache im Allgemeinen hat.

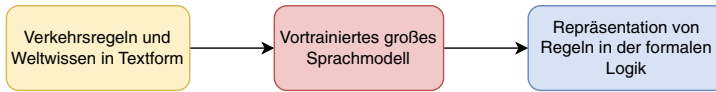


Abb. 14.5 Flussdiagramm der Übersetzung von natürlicher Sprache in formale Logik durch Fine-Tuning

Abb. 14.5 veranschaulicht den Fluss von Verkehrsregeln in natürlicher Sprache zur Regelformalisierung mithilfe von LLMs. Für die Übersetzung von Text in formale Sprache können LLMs wie BERT (Devlin et al. 2019), T5 (Raffel et al. 2020) und GPT-3 (Brown et al. 2020) und andere aktualisierte Versionen dieser Sprachmodelle gute Ausgangspunkte sein und können auf der Grundlage von Lizenz und Open-Source-Natur ausgewählt werden. Da sich die Forschung zunehmend auf diese LLMs konzentriert, können die Nutzer von Tools nach neuen Modellen suchen, die eine bessere Unterstützung oder einen einfacheren Zugang zu den Modellparametern bieten könnten. Manas und Paschke (2023) übersetzten textuelle Verkehrsregeln mit LLM in metrische temporale Logik. Sie verwendeten Prompting zusammen mit der zusätzlichen Verwendung von Semantic Role Labeling (SRL) als weiche Einschränkung für die LLM-Aufforderungen. SRL hilft zusätzlich bei der Beseitigung von Ambiguität in natürlichsprachlichen Regeln, basierend auf den Rollen der Wörter in den Regeln.

Beispiel für die Erstellung eines Datensatzes für Fine-Tuning oder Prompting für den Anwendungsfall AD:

Anweisung (für Prompting erforderlich): Übersetzen Sie Verkehrsregeln in natürlicher Sprache in metrische temporale Logik.

Verkehrsregel in natürlicher Sprache: Ego vehicle will not exceed the speed limit of the lane it is driving on, and it will not exceed the maximum velocity allowed for its vehicle type, and ego will not exceed the speed limit such that it can no longer react to traffic regulations.

MTL-Übersetzung:

$$G(\text{keep_lane_speed_limit}(\text{ego}) \wedge \text{keep_vehicle_type_limit}(\text{ego}) \wedge \text{keep_braking_speed_limit}(\text{ego}))$$

Aus Gründen der Standardisierung haben wir die Promptings in englischer Sprache verfasst, sie können aber auch in Deutsch oder einer anderen Sprache formuliert werden.

Benutzer können ihren eigenen benutzerdefinierten Datensatz für Fine-Tuning oder Prompting erstellen, um das Modell auf neue Aufgaben auszurichten. Bei einer solchen automatischen Formalisierung müssen sich die Entwickler der Prädikate bewusst sein. Unserer Erfahrung nach ist dies schwierig, da Sprachmodelle probabilistisch sind und sie aufgrund der Unsicherheiten des Modells Schwierigkeiten haben, Regeln mit einer festen Liste von Prädikaten zu formalisieren. Zum Beispiel können LLMs manchmal das Prädikat „overtake“ und manchmal „overtaking“ für dieselbe Regel als Eingabe generieren. Dies kann durch die Einführung zusätzlicher Beschränkungen in das Modell begrenzt werden, was in diesem Kapitel jedoch nicht behandelt werden kann.

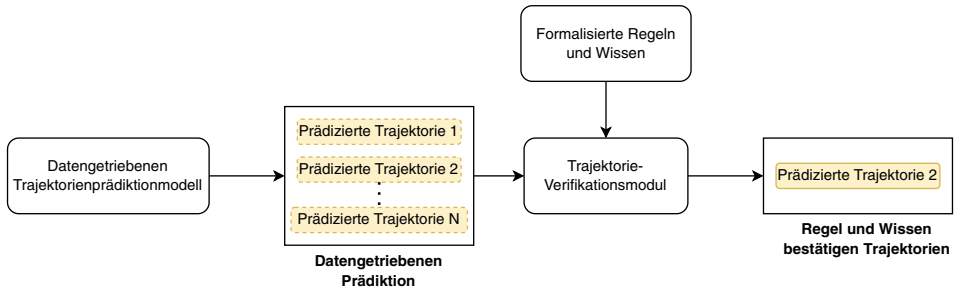


Abb. 14.6 Hybridmodell im Kontext der Trajektorienvorhersage eines autonomen Fahrzeugs

14.3.5 Formalisierte Regelintegration für die Trajektorienvorhersage

In diesem Abschnitt wird erläutert, wie hybride Modelle datengesteuerte und regelbasierte Ansätze zur Trajektorienvorhersage und -planung beim autonomen Fahren kombinieren können. Hybride Modelle können Trajektorien generieren, die mit formalisierten Verkehrsregeln übereinstimmen und die schnelle Entscheidungsfähigkeit neuronaler Netze nutzen, wie in Abb. 14.6 dargestellt. Wir können ein beliebiges vorhandenes maschinelles Lernmodell verwenden, um mögliche Trajektorien auf der Grundlage von Interaktionen und semantischen Informationen, wie z. B. HD-Karten, zu erstellen. Dann können wir unsere formalisierten Regeln und unser Wissen anwenden, um die Trajektorien herauszufiltern, die die Regeln verletzen. Wir können diese Informationen auch als Verlustfunktion verwenden, um das maschinelle Lernmodell so zu trainieren, dass es regelkonforme Trajektorien erzeugt. Alternativ können wir formalisierte Regeln als Verifizierungsschritt nach der Generierung der Trajektorien verwenden, wodurch wir die Generierungs- und Verifizierungskomponenten des Modells trennen können. Hybride Modelle sind von Vorteil, wenn die Regeln unklar oder nur auf bestimmte Situationen anwendbar sind. Das Modell kann sich auf das datengesteuerte System verlassen, um in solchen Fällen die wahrscheinlichste Trajektorie auszuwählen.

Eine parametrisierte Form der temporalen Logik, die sogenannte STL, kann in die auf neuronalen Netzen basierende Trajektorienvorhersage integriert werden. Li et al. (2021) zeigten, dass STL-basierte formale Logik auf differenzierbare Weise mit einem neuronalen Netz verwendet werden kann, wobei die Basistrajektorienvorhersage durch die STL-Robustheitsschicht geleitet wird, um die vom Basisprädiktor erzeugte Trajektorie zu korrigieren. Die STL-Robustheitsschicht besteht aus der STL-Formel und zusätzlichen Umgebungsinformationen, die für die Anwendung von Regeln auf die Trajektorie benötigt werden. Dies ist ein recht interessanter Ansatz, bei dem das Modell versucht, gleichzeitig aus Daten und Regeln zu lernen, und auch lernt, Regeln und Daten zu kombinieren, um ein gutes Gleichgewicht zwischen regelbasiertem und datengesteuertem Lernen zu erreichen. Abb. 14.7 zeigt diesen Arbeitsablauf, bei dem die logischen STL-

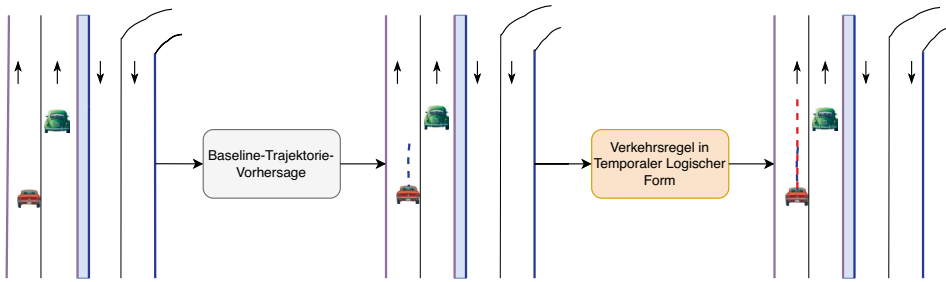


Abb. 14.7 Bahnkorrektur auf der Grundlage der STL-Regeln. Der Pfeil auf der Fahrbahn zeigt die Fahrtrichtung an, während die gestrichelten Linien die zukünftige Flugbahn des Ego-Fahrzeugs (das rote Fahrzeug) anzeigen. (In Anlehnung an Li et al. 2021)

Formeln Trajektorien korrigieren. Hier ist das grün gefärbte Auto ein Ego-Auto, das blaue Auto steht für andere Verkehrsteilnehmer und die Punkte in der Abbildung stellen die Trajektorie dar. Wie hier gezeigt, wird die erste Trajektorie des Egos durch das Trajektorienvorhersagemodul vorhergesagt. Dann nimmt eine zusätzliche Schicht mit formaler Logik (aktiviert durch Backpropagation) die erste Vorhersage und korrigiert die Vorhersage auf der Grundlage der im zweiten Modell kodierten logischen Regel. Eine bessere Vorhersage, die auf einer Kombination aus lern- und regelbasierter Architektur basiert, hilft bei der sicheren und robusten Planung für das Modul für automatisiertes Fahren. Diese Art der Modellarchitektur ist näher an hybriden Modellkonzepten, bei denen das Beste aus der symbolischen und der subsymbolischen Welt kombiniert wird, um bessere Ergebnisse für das Problem der Trajektorienvorhersage zu erzielen, die für eine sichere und langfristige Planung beim automatisierten Fahren unerlässlich ist.

Eine weitere Möglichkeit, formalisierte Regeln zu verwenden, sind Modelle, die auf Reinforcement Learning (RL) basieren. Zu diesem Zweck können Regeln als Einschränkungen verwendet werden oder sie können verwendet werden, um die Reward-Funktion von RL-Modellen zu formulieren. Lin et al. (2022) verwendeten temporale logische Regeln in einem RL-basierten Modell als Markov-Entscheidungsprozess, um herauszufinden, welche Fahrzeugaktion die Regel verletzt wird. In ihrer Arbeit verwendeten sie ein RL-basiertes Modell, das anhand von Regelsätzen trainiert wird, und testeten dann die Auswirkungen der Hinzufügung/Änderung neuer Regeln zu bestehenden RL-basierten trainierten Modellen. Diese Arbeit besteht aus zwei Modulen, nämlich einer RL-basierten und einer Sample-based Backup Policy (Kingston et al. 2018). Da ein RL-basiertes Modell nicht ohne Weiteres neue hinzugefügte oder geänderte Regeln integrieren kann, wird eine Sample-based Backup Policy verwendet, wenn Trajektorien entdeckt werden, die gegen neue oder aktualisierte Regeln verstoßen. Auf diese Weise beinhaltet dasselbe Modell die komplexe Entscheidungsfindung des RL-basierten Modells und die Flexibilität bei der Handhabung neuer Regeln durch die Sample-based Backup Policy. Die meisten der oben diskutierten Modelle sind jedoch aus

einer Planungsperspektive heraus entwickelt worden und konzentrieren sich mehr auf die Aktionen der eigenen Fahrzeuge als auf die Überprüfung, ob andere Agenten oder Fahrzeuge in der Umgebung mögliche Aktionen auf der Grundlage der Regeln haben. Ein möglicher Grund dafür sind die Zeit- und Rechenbeschränkung, die ein limitierender Faktor sein können, wenn ein Modell versucht, alle möglichen Trajektorien und deren Plausibilität anhand der Regeln zu überprüfen, da es mehrere Agenten gibt, von denen jeder mehrere mögliche, von Modellen generierte Trajektorien hat.

Formalisierte Regeln können auch die Trajektorien überwachen und bei der Entscheidungsfindung helfen. Bei der Überwachung der Trajektorien geht es darum, die vom Trajektorienvorhersagemodul generierte Trajektorie zu verfolgen, um die formalisierten Regeln einzuhalten. Sie können sowohl online als auch offline überwacht werden. Die Offline-Überwachung eignet sich hervorragend, um die Leistung zu überprüfen und Informationen detaillierter zu analysieren, da sie Regeln auf Trajektorien anwenden und feststellen kann, welche Trajektorien die Regeln verletzt haben. Auf der Grundlage dieser Bewertungsdaten können wir den Regelkonformitätsstatus des Modells durchführen. Allerdings helfen sie den laufenden Systemen nicht direkt bei der Rückmeldung. Online-Überwachungssysteme können Rückmeldungen in Echtzeit liefern; wenn sie dafür ausgelegt sind, können sie sofort reagieren. Online-Monitore gewährleisten die Sicherheit und Zuverlässigkeit eines bereits laufenden Systems. Aufgrund des Echtzeitcharakters können wir bei Online-Monitoren jedoch nicht auf die gesamte von den Fahrzeugen durchfahrene Trajektorie zugreifen. Wir erhalten Informationen über vergangene Trajektorien und mögliche zukünftige Trajektorien oder Absichten. Für die Monitore werden die Trajektorien als Signale betrachtet und diese Signale werden anhand der Regeln überwacht. Krasowski und Althoff (2021) und Maierhofer et al. (2020, 2022) verwendeten Monitore, um Trajektorien gegen formalisierte Regeln zu prüfen.

14.3.6 Induktives Regellernen für die Klassifikation

Wissensgraphen (KGs) werden durch Regeln bereichert, die interpretierbare Muster in Daten erfassen und die Vervollständigung von KGs sowie die Ableitung neuer Fakten erleichtern. Im Bereich des semantischen Webs werden oft handgefertigte Regeln in Form von Horn-Klauseln in eine Wissensbasis aufgenommen, die OWL-Axiome ergänzen, um das Schließen zu unterstützen. Allerdings kann das Erstellen von Regeln für große Datensätze zeitaufwendig sein und die Regeln können sogar unzuverlässig werden, wenn neue Daten eingeführt werden. Um dieser Herausforderung zu begegnen, gibt es eine wachsende Nachfrage nach dem direkten Erlernen von Regeln aus Daten, was durch ILP erreicht wird. Der Begriff ILP wurde erstmals von Muggleton (1991) eingeführt und beinhaltet das Erlernen von Logikprogrammen (eine Reihe von logischen Regeln), die Trainingsbeispiele verallgemeinern. Statt sich ausschließlich auf statistische Mechanismen zu verlassen, wie es die meisten ML-Ansätze tun, basiert ILP auf logischem Schließen und integriert Ideen aus automatisiertem Schließen und Wissensrepräsentation. Der

ILP-Ansatz hat den Vorteil, dass er von sehr kleinen Datensätzen verallgemeinern kann und seine Lösungen verständlich und überprüfbar sind. Während verschiedene ILP-Frameworks verschiedene Strategien anwenden, um eine gegebene Aufgabe zu lösen, stimmen die meisten von ihnen in der folgenden Definition eines ILP-Problems überein. Ein ILP-Problem $(B; E+; E-)$ besteht aus den folgenden Daten:

- B ist eine Reihe von Hintergrundannahmen,
- $E+$ ist eine Reihe von positiven Beispielen,
- $E-$ ist eine Reihe von negativen Beispielen.

Das Ziel der ILP besteht darin, ein Logikprogramm zu konstruieren, auch bekannt als Hypothese, das in Kombination mit gegebenen Hintergrundannahmen die positiven Beispiele genau ableiten und die negativen Beispiele ablehnen kann. Das Logikprogramm wird typischerweise mithilfe der Prädikatenlogik dargestellt, die durch eine Reihe von Konstanten, Prädikaten, Funktionen und Variablen definiert ist. Konstanten C repräsentieren die Diskursobjekte, n -ary Prädikate können als Abbildungen von C^n auf die boolesche Menge $B = \{True, False\}$ gesehen werden und n -ary Funktionen sind Abbildungen von C^n auf die Menge der Konstanten C , während Variablen nichtspezifizierten Konstanten entsprechen. Ein Atom ist ein auf ein Tupel von Argumenten angewandtes Prädikat. Die Argumente können Konstanten oder Variablen sein. Wenn alle Argumente eines Atoms Konstanten sind, spricht man von einem Grundatom. Wohlgeformte Formeln der FOL werden rekursiv als Kombinationen von Atomen mit logischen Verknüpfungen (wie Negation, Konjunktion, Disjunktion, Implikation) und existenziellen oder universellen Quantoren (wie \exists, \forall) definiert. Klauseln sind eine spezielle Klasse von Formeln, die als Disjunktion von Literalen geschrieben werden können, die Atome oder ihre Negationen sind. Horn-Klauseln sind Klauseln mit höchstens einem positiven Literal, während definitive Horn-Klauseln genau ein positives Literal enthalten. Im Kontext der ILP spielen definitive Klauseln eine entscheidende Rolle, da sie als Wenn-dann-Regeln umgeschrieben werden können:

$$H \leftarrow B_1 \wedge B_2 \wedge \dots \wedge B_k$$

wo H als Kopfatom und B_i als Körperatom bezeichnet wird. Die Variablen, die im Kopfatom erscheinen, sind typischerweise mit einem universellen Quantor instanziiert, während die anderen Variablen in den Körperatomen existenziell quantifiziert sind (Glanois et al. 2021). Es ist oft üblich, ein Logikprogramm als eine Menge von disjunktiven Horn-Klauseln auszudrücken, um das Lernproblem zu vereinfachen.

Im Bereich der autonomen Fahrsysteme erfordern handgefertigte Regeln ständige Aktualisierungen und Änderungen, wenn neue Fälle auftreten. Ein effizienterer Ansatz wäre, Regeln direkt aus Daten zu lernen, die aus Simulatoren und realen Fahrdaten gesammelt werden, und dabei die Technik der ILP zu nutzen. Es gibt verschiedene ausgereifte symbolische ILP-Systeme, wobei Prolog das einflussreichste ist und viele Systeme beeinflusst hat. Aleph (Srinivasan 2007) ist ein auf Prolog basierendes ILP-System,

das inverse Implikation und eine Allgemein-zu-spezifisch-Suchstrategie zur Erlernung logischer Regeln aus relationalen Daten verwendet. Der Suchraum für ein Logikprogramm ist typischerweise sehr groß. Aleph verwendet Modus-Deklarationen, um diesen Regelsuchraum einzuschränken. Diese Deklarationen spezifizieren, welche Prädikatsymbole in einer Regel erscheinen dürfen, ihre Häufigkeit des Auftretens und die Argumenttypen, die sie annehmen können. Aleph beginnt mit einer leeren Hypothese und spezialisiert sie dann schrittweise durch Hinzufügen von Bedingungen nach einem Set-Covering-Ansatz (Cropper und Dumančić 2022):

1. Wählen Sie ein positives Beispiel zur Verallgemeinerung aus. Wenn keins existiert, stoppen Sie und geben Sie die aktuelle Hypothese zurück; ansonsten gehen Sie zum nächsten Schritt über.
2. Konstruieren Sie die spezifischste Klausel, auch bekannt als die unterste Klausel, die mit den Modus-Deklarationen konsistent ist und das Beispiel beinhaltet.
3. Suchen Sie nach einer Klausel, die allgemeiner als die unterste Klausel ist und die beste Punktzahl hat.
4. Fügen Sie die Klausel zur Hypothese hinzu und entfernen Sie alle positiven Beispiele, die sie abdeckt. Kehren Sie zum Schritt 1 zurück.

Aleph begrenzt den Suchraum, indem es von der allgemeinsten obersten Klausel \perp ausgeht, um die Klausel zu spezialisieren, indem es Literale zu ihrem Körper hinzufügt, die aus der untersten Klausel oder durch Instanziierung von Variablen ausgewählt werden. Es bewertet jede Klausel mit einer Abdeckungsfunktion, die die Anzahl der positiven und negativen Beispiele angibt, die die Klausel jeweils abdeckt, und weist jeder Klausel in der Suche mit einer beschränkten Breitensuche und A^* einen Score zu. Sobald die beste Klausel gefunden ist, fügt Aleph sie zur Hypothese hinzu und entfernt alle positiven Fälle, die von der neuen Hypothese abgedeckt werden. Dieser Prozess wird fortgesetzt, bis keine weiteren positiven Beispiele mehr zu behandeln sind. Schließlich kann das erlernte Logikprogramm als Disjunktion von Horn-Klauseln dargestellt werden. In der Operational Design Domain (ODD) für ein autonomes Fahrzeug ist es üblich, Regeln für die Erkennung von Randfahrtszenarien manuell zu erstellen, um die Fahrsicherheit zu gewährleisten. Bei der Simulation solcher Fahrtszenarien und dem Aufbau eines ABox für Hintergrundwissen können wir bestehende ILP-Systeme nutzen, um ein Logikprogramm für ein Zielprädikat zu lernen, wie z. B. $pos(S)$, das eine Verkehrsszene, S , als einen Randfall klassifiziert. Zum Beispiel können wir eine Ausnahmeregel induzieren, die es einem Ego-Fahrzeug erlaubt, eine durchgezogene Linie zu überqueren, als einen Randfall.

$pos(S) \leftarrow scene(S), hasParticipant(S, V1), hasParticipant(S, V2), vehicle(V1), egoVehicle(V2), lane(L1), lane(L2), oncomingLane(L2), EmptyLane(L2), solidLine(M), leftConnectedTo(M, L2), rightConnected(M, L1), locatedOn(V2, L1), behind(V2, V1), block(V1, L1)scene(S), hasParticipant(V1), hasParticipant(V2), vehicle(V1), egoVehicle(V2), lane(L1), lane(L2), oncomingLane(L2), EmptyLane(L2), solidLine(M),$

leftConnectedTo(M, L2), rightConnected(M, L1), locatedOn(V2, L1), behind(V2, V1), block(V1, L1)

Diese Regel legt fest, dass eine Szene S als Regelabweichung (d. h. $\text{pos}(S)$) klassifiziert wird, wenn sie die gegebene Reihe von Bedingungen erfüllt, einschließlich der Anwesenheit von zwei Fahrzeugen ($V1$ und $V2$), fester und leerer Fahrspuren ($L1$ und $L2$), einer durchgezogenen Linie (M), der positionellen Beziehungen zwischen ihnen. Darüber hinaus muss die Ego-Fahrspur durch ein anderes Fahrzeug blockiert sein.

Für eine detaillierte Anleitung zum Aufbau von Hintergrundwissen und zur Definition von positiven und negativen Prädikaten für das Lernen empfehlen wir, sich das Online-Tutorial von Aleph und das Papier von Muggleton (1995) anzusehen. Darüber hinaus gibt es verschiedene andere symbolische Inductive-Logic-Programming-(ILP-)Systeme, wie zum Beispiel TILDE (Blockeel und Raedt 1998), ASPAL (Corapi et al. 2012) und Metagol (Cropper und Muggleton 2016). TILDE ist eine Methode, die Entscheidungsbäume verallgemeinert, indem sie Informationsgewinn verwendet, um Trainingsbeispiele zu zerlegen und zu analysieren. ASPAL ist ein System, das auf Metaebene arbeitet, indem es einen Answer Set Programming (ASP) Solver verwendet, um das ILP-Problem zu lösen, und es hat nachfolgende Arbeiten, wie ILASP, erheblich beeinflusst. Metagol erstellt einen Beweis einer Menge von Beispielen mit einem Prolog-Meta-Interpreter und extrahiert dann ein Programm aus dem Beweis. Die Auswahl eines geeigneten ILP-Systems hängt von der Aufgabe und den bisherigen Erfahrungen des Benutzers ab.

Der Erfolg der symbolischen ILP hängt von verschiedenen Faktoren ab, einschließlich der Art und Weise, wie Modus-Deklarationen angegeben werden, wie Grundklauseln erstellt werden und welcher Klauselsuchmechanismus verwendet wird. Diese Anforderungen können ihre Nutzung für Benutzer ohne Fachwissen und Berufserfahrung einschränken. Jüngste Fortschritte in der ILP haben zur Entwicklung neuer Ansätze geführt, die die Leistungsfähigkeit von neuronalen Netzen nutzen, um Regeln effizienter zu lernen, indem sie die Regelsuche auf die Lösung numerischer Optimierungsprobleme mit gradientenbasierten Methoden delegieren. Diese Ansätze ersetzen oft absolutes logisches Schließen durch eine entspannte Version, die kontinuierliche Werte erzeugt, die das Vertrauen in die Schlussfolgerung widerspiegeln. Eine häufige Strategie besteht darin, Prädikate als binäre Tensoren über der Domäne der Konstanten darzustellen und das Schließen durch Ketten von Tensorprodukten durchzuführen, die Klauseln emulieren. Mit dieser Technik können diese Ansätze effizient interpretierbare und genaue Regeln aus großen, komplexen Datensätzen lernen. Insbesondere ermöglichen sie die Integration von symbolischer und subsymbolischer Verarbeitung, was zu flexibleren und leistungsfähigeren Modellen führt, die komplexe Beziehungen und Muster in den Daten erfassen können. Um zu illustrieren, wie das funktioniert, nehmen wir ein Beispiel von einem neuronalen induktiven Regellernansatz (Yang und Song 2020), der auf die Klassifikation eines Verkehrsteilnehmers angewendet wurde.

Das aus Bildern und anderen Datenquellen extrahierte Szenendiagramm kann als eine relationale Wissensbasis interpretiert werden, die aus einer Reihe von Prädikaten besteht, die die Fakten der Fahrzeugumgebung beschreiben. Dies dient als Hintergrundwissen für

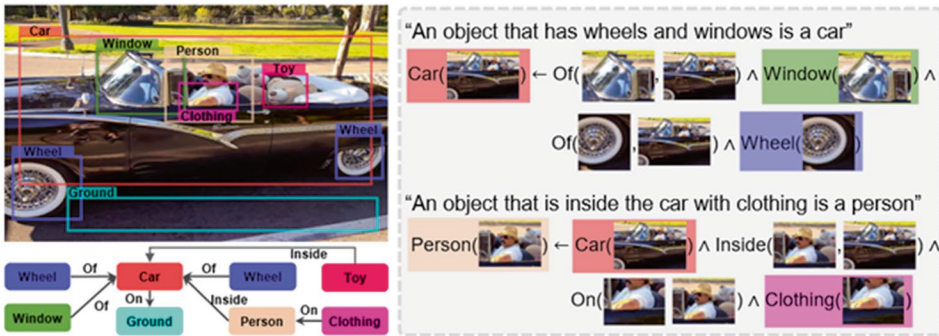


Abb. 14.8 Ein Szenendiagramm, das die Beziehungen von Objekten beschreibt. (Angepasst von Yang und Song 2020)

ein ILP-Problem. Das Erlernen von Regeln der FOL über mehrere Szenendiagramme kann eine globale Erklärung für die Klassifizierung bieten und so die Robustheit des Wahrnehmungsmoduls in einem autonomen Fahrzeug verbessern. Wie in Abb. 14.8 gezeigt, erklärt die gelernte Regel für das Zielprädikat $Car(X)$, dass ein Objekt als Auto klassifiziert wird, weil es Räder und Fenster hat. Durch die Integration von kontextuellen Informationen aus der Fahrzeugumgebung kann das Vertrauen in die Detektionsergebnisse durch diese Art von „gehobenem“ Wissen erhöht werden. Dies ist besonders wertvoll, wenn die an Bord erzielten Detektionsergebnisse geringes Vertrauen erwecken.

Der Prozess der Regelentdeckung in ILP ist herausfordernd aufgrund der kombinatorischen Natur des Suchprozesses. Wenn die Anzahl der Regeln und Variablen in den Daten wächst, kann der Suchraum möglicher Kombinationen schnell unhandhabbar werden. Neben der Verwendung von Modus-Deklarationen oder der Definition einer Regelvorlage können wir eine Technik des mehrstufigen Rasonierens anwenden, die ein effektives Verfahren zur Beantwortung von Anfragen in Wissensgraphen ist. Das Erlernen von Logikprogrammen in ILP kann als Suche nach relationalen Pfaden angesehen werden, die vom Subjekt zum Objekt führen. Dies kann als kettenartige Regel der ersten Ordnung dargestellt werden. Um den Regelsuchraum zu verfeinern, beschränken wir den Kopf der Regel auf ein einstelliges Prädikat, das nur die Zielvariable enthält, und den Körper der Regel nur auf die Zielvariable und Variablen, die direkt mit ihr in Beziehung stehen. Zum Beispiel könnte eine gelernte FOL-Regel zur Klassifizierung einer Person auf der Straße folgendermaßen aussehen:

$$\begin{aligned}
 Person(X) \leftarrow & (Shirt(Y_1) \wedge Wearing(X, Y_1)) \vee (Pants(Y_2) \wedge Wearing(X, Y_2)) \\
 & \vee (Street(Y_3) \wedge WalkingOn(X, Y_3))
 \end{aligned}$$

In dieser Regel sind die Variablen Y_1 and Y_2 über das Prädikat „wearing“ mit der Variable X verknüpft. Diese Regel definiert eine Person als ein Objekt, das entweder ein Hemd oder eine Hose trägt oder auf der Straße läuft. Es ist wichtig zu beachten, dass wir

Variablen, die nicht direkt mit der Zielvariablen in der Regel in Beziehung stehen, ausschließen. Zum Beispiel wäre das Ableiten einer Regel wie $\text{Person}(X) \leftarrow \text{Pants}(Y)$ unmöglich, weil X nicht mit Y durch irgendwelche Prädikate in Beziehung steht und wir annehmen, dass es keine klare kausale Beziehung zwischen diesen beiden Variablen gibt. Obwohl dies die Ausdrucksfähigkeit der Regel einschränken kann, ist es unwahrscheinlich, das Vorhandensein einer Person in einem Bild einfach durch das Überprüfen des Vorhandenseins einer Hose zu erschließen.

Basierend auf dieser Annahme können wir diese Regeln in Skolem-Normalform umwandeln, indem wir alle Variablen unter dem existenziellen Quantor im Regelkörper durch Funktionen ersetzen. Funktionen können mit vorhandenen vordefinierten Prädikaten innerhalb eines Datensatzes ausgedrückt werden, die wiederum Variablen zurückgeben. Zum Beispiel kann ein gegebenes Prädikat $\text{Inside}(X, Y)$ als Funktion $\varphi_{\text{inside}}(X)$ dargestellt werden, die die Variable Y zurückgibt. Im Fall von einstelligigen Prädikaten wie $\text{Clothing}(Z)$ gibt die Funktion $\varphi_{\text{Clothing}}()$ mit leeren Argumenten die Variable Z zurück. Nach dem Ersetzen von Variablen durch Funktionen erhalten wir primitive Aussagen, die aus Prädikaten bestehen, die Funktionen als ihre Argumente haben. Zum Beispiel erhalten wir die einstellige primitive Aussage $\text{Car}(\varphi_{\text{inside}}(X))$, die $\text{Inside}(X, Y) \wedge \text{Car}(Y)$ ersetzt, und die zweistellige primitive Aussage $\text{Inside}(\varphi_{\text{Clothing}}(), \varphi_{\text{Identity}}(X))$, die $\text{Inside}(Z, X) \wedge \text{Clothing}(Z)$ ersetzt. Sie mappen den Eingangsraum auf einen Skalarwert zwischen 0 und 1 in weicher Logik. Das Erlernen einer solchen primitiven Aussage kann als das Finden von relationalen Pfaden interpretiert werden, die durch das Produkt von Tensoren berechnet werden können, die zu mehreren Prädikaten gehören. Schließlich wird die Formel gebildet, indem Prädikate mit logischen Verknüpfungen kombiniert werden. Dieser Prozess ermöglicht es uns, jede FOL-Formel in skolemisierte Formeln zu transformieren. Zum Beispiel können wir transformieren

$$\text{Person}(X) \leftarrow \text{Inside}(X, Y) \wedge \text{Car}(Y) \wedge \text{Inside}(Z, X) \wedge \text{Clothing}(Z)$$

in:

$$\text{Person}(X) \leftarrow \text{Car}(\varphi_{\text{inside}}(X)) \wedge \text{Inside}(\varphi_{\text{Clothing}}(), \varphi_{\text{Identity}}(X)).$$

Hierbei gibt die Funktion $\varphi_{\text{Identity}}$ ihr Eingangsargument selbst zurück, was es ermöglicht, Kopfvariablen in der Formel des Rumpfes wiederverwenden zu können. Darüber hinaus können wir die Funktion rekursiv anwenden, um komplexere Beziehungen darzustellen, wie zum Beispiel „die Freunde der Freunde einer Person“, was als $\varphi_{\text{Friend}}(\varphi_{\text{Friend}}(X))$ dargestellt werden kann. Diese Transformation bildet die Grundlage des neuronalen induktiven Regellernens, das das Problem des Regellernens in die Suche nach relationalen Pfaden zwischen Variablen umwandelt. Darüber hinaus wird durch die Tatsache, dass während der Auswertung nur eine Variable konkretisiert werden muss, der Suchraum für Regeln erheblich vereinfacht.

Der gesamte Regelgenerierungsprozess kann mit 3-Stufen-Transformer-Modulen implementiert werden. Das erste Modul generiert Funktionen, indem es Variablen Prädikaten zuordnet. Das zweite Modul generiert primitive Aussagen, indem es die generierten

Funktionen Prädikaten zuordnet. Das letzte Modul berechnet die logischen Kombinationen über alle primitiven Aussagen. Für weitere Details zur Implementierung verweisen wir auf das Originaldokument von Yang und Song (Yang und Song, 2020). Der letzte Schritt gibt alle möglichen Formeln f_L aus und einer Aufmerksamkeitsmatrix S_0^T , die ihre Vertrauenswürdigkeit angibt. Um diese Formeln zu bewerten, wird eine Minibatch von Abfragen $\{ \langle x, P^*, x', y \rangle_i \}_{i=1}^b$ aus der Wissensbasis (KB) entnommen. Hierbei ist y die Abfragebezeichnung, die anzeigt, ob das Triplet in der Wissensbasis (KB) vorhanden ist oder nicht. Die Verlustfunktion wird als Kreuzentropie zwischen der Abfragebezeichnung und dem berechneten Formelwert definiert, wie folgt:

$$\arg \min_w \frac{1}{b} \sum_i^b \text{Cross Entropy}(y_i, S_0^T f_L(x, x'))$$

Da die Aufmerksamkeiten während des Regelgenerierungsprozesses differenziell erzeugt werden, kann der Verlust über die Aufmerksamkeiten in die Transformer-Netzwerke zurückpropagiert werden, um ein End-to-end-Training zu ermöglichen.

Neuronale ILP-Ansätze sind äußerst effizient bei der Verarbeitung von rauschhaften und umfangreichen Datensätzen. Durch die Extraktion gelernter Aufmerksamkeitsmatrizen für Funktionen, primitive Aussagen und Formeln können die FOL-Regeln explizit rekonstruiert werden, wodurch eine globale Erklärung des Zielprädikats, wie z. B. der Klassifizierung, ermöglicht wird. Neben dem oben genannten Ansatz gibt es in den letzten Jahren weitere neuralbasierte ILP-Architekturen. Beachtenswerte Beispiele sind die Ansätze von Yang et al. (2017), Payani und Fekri (2019) sowie Wang et al. (2020). Diese können als Alternativen zur Lösung von Aufgaben wie der Klassifizierung von Fußgängern, Fahrzeugen und Verkehrszeichen in autonomen Fahrsystemen dienen.

14.4 Empfehlungen

Die Gewährleistung einer sicheren und zuverlässigen Entscheidungsfindung auf hoher Ebene in autonomen Fahrsystemen ist eine entscheidende Herausforderung. Wir schlagen eine Softwarearchitektur vor, die a) Entscheidungen zur Objekterkennung mithilfe von erlernten logischen Regeln erster Ordnung erklären und verifizieren kann, die die Unzulänglichkeiten aktueller Deep-Learning-basierter Modelle beheben, b) Vorhersage und Planung von Trajektorien basierend auf formalisierten Verkehrsregeln.

Für die Objekterkennung umfasst die Pipeline die Auswahl einer geeigneten Ontologie, die Konvertierung und Organisation von Rohsignalen in strukturierte Daten und die Anwendung von ILP-Algorithmen zur Automatisierung des Regellernprozesses. Trotz ihrer Effektivität haben wir jedoch einige Einschränkungen festgestellt, die berücksichtigt werden müssen. Die ursprüngliche OpenX-Ontologie für den Straßenverkehr umfasst 393 Klassen, 100 Eigenschaften und 2005 Axiome, was sie für Echtzeitschlussfolgerungen zu schwer macht. Darüber hinaus sind die aktuellen Algorithmen zur

Generierung von Szenegraphen aus Bildern noch nicht für die industrielle Anwendung geeignet. Sie weisen bei mehreren Benchmark-Datensätzen eine geringe Leistung auf, wobei die Wiedererkennungsraten in der Regel unter 50 % liegen. Dies zeigt, dass dieser Bereich intensiverer Forschung und Entwicklung bedarf. Traditionelles ILP erfordert gute Regelvorlagen oder Modus-Deklarationen für das Lernen von Regeln, was es für Benutzer schwierig macht, denen es an Fachwissen im Anwendungsbereich mangelt und die nicht mit falsch beschrifteten oder verrauschten Daten umgehen können, die in automatisierten Systemen sehr häufig vorkommen. Andererseits ist neuralbasiertes ILP schwer über die Trainingsdaten hinaus zu verallgemeinern, was zu einer geringeren Dateneffizienz führt. Zusammenfassend haben wir die folgenden Vorschläge:

1. Eine maßgeschneiderte Ontologie soll speziell für die jeweilige Aufgabe erstellt werden und möglicherweise auf der existierenden Ontologie aufbauen. Alle Elemente, die nicht mit der Aufgabe zusammenhängen, sollten entfernt werden, um sowohl die Schlussfolgerungseffizienz als auch die Leistung des Regellernens zu optimieren.
2. Bei der Verwendung von aus Bildern extrahierten Szenegraphen ist Vorsicht geboten. Es ist wichtig, die Ergebnisse zu überprüfen und zu verbessern, indem man Informationen aus anderen Datenquellen im Fahrzeug vergleicht.
3. Sensoren generieren jede Sekunde eine große Menge an Daten. Daher ist es entscheidend, redundante Daten herauszufiltern und aufgabenbezogene Daten für eine effektive Abbildung auf eine Ontologie zu identifizieren.
4. Neuronales ILP bietet mehr Vorteile als symbolische Ansätze bei der Verarbeitung großer und fehlerhaft gekennzeichneteter Daten. Es ist jedoch wichtig, mehr Daten aus realen Fahrscenarien oder Simulationen zu sammeln, um das Regellernen zu verbessern.

Die Trajektorienvorhersage und Planungs-Pipeline bestehen aus folgenden Schritten: a) Auswahl der Verkehrsregeln, die unser Modell befolgen soll, b) Formalisierung dieser Regeln unter Verwendung einer geeigneten Logik auf der Grundlage des Anwendungsfalls, c) Entscheidung, ob die formalisierten Regeln zur Überprüfung der Modellkonformität oder zur durchgängigen Trajektorienvorhersage und -planung verwendet werden sollen. Die erste Option ist in der Regel vom Hauptmodul für die Trajektorienvorhersage oder -planung getrennt, während bei der zweiten Option die formalisierten Regeln zusammen mit Daten für eine regelkonformere Trajektorienvorhersage oder -planung verwendet werden. Zusammenfassend haben wir die folgenden Empfehlungen:

1. Erstellen Sie eine Regelsammlung, die ein Modell zur Vorhersage und Planung von Trajektorien befolgen soll. Zum Beispiel Regeln zum Überholen oder zum Umgang mit Regelausnahmen bei Vorhandensein eines vorrangigen Fahrzeugs (Krankenwagen).
2. Zur Überprüfung der Modellkonformität werden Szenarien aus dem Datensatz extrahiert oder mit verfügbaren Tools (z. B. Commonroad⁵) erstellt. Anschließend wird das

⁵<https://commonroad.in.tum.de/>

Trajektorienvorhersage- oder Planungsmodell verwendet, um eine Trajektorie zu erstellen. Schließlich wird die Regel als Monitor verwendet, um festzustellen, ob die erzeugte Trajektorie den festgelegten Regeln entspricht oder nicht.

3. Wenn die Regeln gemeinsam mit dem datengesteuerten Modell verwendet werden, müssen wir darauf achten, die Auswirkungen (oder die Gewichtung) der Regeln gegenüber den Daten zu bestimmen. Dies kann je nach Zielsetzung unterschiedlich sein, wie z. B. eine stärkere Konzentration auf die Einhaltung von Regeln, sicheres Fahren oder effizientes Fahren, da diese Ziele manchmal im Widerspruch zueinander stehen.

14.5 Zusammenfassung

In diesem Kapitel haben wir verschiedene neurosymbolische Techniken untersucht, die die Entscheidungsfindung beim autonomen Fahren erleichtern sollen. Zu diesen Techniken gehören die Entwicklung einer Ontologie des Straßenverkehrs, die Extraktion strukturierter Daten, die Formalisierung von Regeln und das Lernen von Regeln. Das formalisierte Wissen, das aus diesen Techniken abgeleitet wird, kann für das Regellernen bei Klassifizierungsaufgaben verwendet werden, wodurch die Erklärbarkeit und Zuverlässigkeit der Ergebnisse erhöht wird. Darüber hinaus können diese Regeln in Entscheidungsprozesse zur Vorhersage von Fahrzeugtrajektorien und zur Planung integriert werden. Eine solche Integration erweist sich in sicherheitskritischen Szenarien des autonomen Fahrens als bedeutsam, wo sie wesentliche Garantien für Sicherheit und Zuverlässigkeit bieten kann.

Literatur

- Baader, F. et al. (2007) *The Description Logic Handbook: Theory, Implementation and Applications*, Cambridge Core. Cambridge University Press. <https://doi.org/10.1017/CBO9780511711787>
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. and Agarwal, S., 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33, pp.1877–1901
- Blockeel, H. and Raedt, L.D. (1998) 'Top-down induction of first-order logical decision trees', *Artificial Intelligence*, 101(1), pp. 285–297. [https://doi.org/10.1016/S0004-3702\(98\)00034-4](https://doi.org/10.1016/S0004-3702(98)00034-4)
- Chang, X. et al. (2021) 'Scene Graphs: A Survey of Generations and Applications', [arXiv:2104.01111 \[cs\]](https://arxiv.org/abs/2104.01111) [Preprint]. <http://arxiv.org/abs/2104.01111>
- Corapi, D., Russo, A. and Lupu, E. (2012) 'Inductive Logic Programming in Answer Set Programming', in *Inductive Logic Programming. International Conference on Inductive Logic Programming*, Springer, Berlin, Heidelberg, pp. 91–97. https://doi.org/10.1007/978-3-642-31951-8_12
- Cropper, A. and Dumančić, S. (2022) 'Inductive logic programming at 30: a new introduction'. [arXiv. http://arxiv.org/abs/2008.07912](https://arxiv.org/abs/2008.07912)

- Cropper, A. and Muggleton, S. (2016) *Metagol System*, *GitHub*. <https://github.com/metagol/metagol>
- David Rogers (2011). Tripliser. <http://daverog.github.io/tripliser/>
- Devlin, J., Chang, M.W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (pp. 4171–4186). Association for Computational Linguistics.
- Dimou, A., Sande, M. V., Colpaert, P., Verborgh, R., Mannens, E., & de Walle, R. V. (2014). RML: A generic language for integrated RDF mappings of heterogeneous data. In *Proceedings of the Workshop on Linked Data on the Web (LDOW2014) Co-Located with the 23rd International World Wide Web Conference (WWW2014) (Vol. 1184). CEUR Workshop Proceedings
- Fensel, D. et al. (2020) *Knowledge Graphs: Methodology, Tools and Selected Use Cases*. Cham: Springer International Publishing. <https://doi.org/10.1007/978-3-030-37439-6>
- Girshick, R. et al. (2013) ‘Rich feature hierarchies for accurate object detection and semantic segmentation’. <https://doi.org/10.48550/arXiv.1311.2524>
- Glanois, C. et al. (2021) ‘Neuro-Symbolic Hierarchical Rule Induction’. arXiv. <http://arxiv.org/abs/2112.13418>
- Google. (2023). Open Images Dataset V7 and Extensions . <https://storage.googleapis.com/openimages/web/index.html>
- Grabowski, M. and Wang, Y. (2023) ‘A CONCEPT TO SUPPORT AI MODELS BY USING ONTOLOGIES - PRESENTED ON THE BASIS OF GERMAN TECHNICAL SPECIFICATIONS FOR LANE MARKINGS’, *27th ESV 2023* [Preprint].
- Gruber, T.R. (1993) ‘A translation approach to portable ontology specifications’, *Knowledge Acquisition*, 5(2), pp. 199–220. <https://doi.org/10.1006/knac.1993.1008>
- Guarino, N. (1995) ‘Ontologies and knowledge bases: towards a terminological clarification’, in, pp. 25–32.
- Han, X. et al. (2021) ‘Image Scene Graph Generation (SGG) Benchmark’, [arXiv:2107.12604](https://arxiv.org/abs/2107.12604) [cs] [Preprint]. <http://arxiv.org/abs/2107.12604>
- Hahn, C., Schmitt, F., Tillman, J.J., Metzger, N., Siber, J., & Finkbeiner, B. (2022). Formal Specifications from Natural Language. ArXiv, abs/2206.01962.
- Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosz, B., & Dean, M. (2004). SWRL: A Semantic Web Rule Language Combining OWL and RuleML. World Wide Web Consortium (W3C). <https://www.w3.org/Submission/SWRL/>
- Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., Bernstein, M. S., & Li, F.-F. (2016). Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations [Manuscript]. [arXiv:1602.07332](https://arxiv.org/abs/1602.07332) [cs.CV]. <https://arxiv.org/abs/1602.07332>
- Karimi, A., & Duggirala, P. S. (2020). Formalizing traffic rules for uncontrolled intersections. 2020 ACM/IEEE 11th International Conference on Cyber-Physical Systems (ICCPS), 41–50. <https://doi.org/10.1109/ICCPS48487.2020.00012>
- Kingston, Z., Moll, M., & Kavraki, L. E. (2018). Sampling-Based Methods for Motion Planning with Constraints. Annual Review of Control, Robotics, and Autonomous Systems, 1(1), 159–185. <https://doi.org/10.1146/annurev-control-060117-105226>
- Li, X., Rosman, G., Gilitschenski, I., DeCastro, J., Vasile, C.-I., Karaman, S., & Rus, D. (2021). Differentiable Logic Layer for Rule Guided Trajectory Prediction. Proceedings of the 2020 Conference on Robot Learning, 2178–2194. <https://proceedings.mlr.press/v155/li21b.html>
- Lin, J., Zhou, W., Wang, H., Cao, Z., Yu, W., Zhao, C., Zhao, D., Yang, D., & Li, J. (2022). Road Traffic Law Adaptive Decision-making for Self-Driving Vehicles. 2022 IEEE 25th International

- Conference on Intelligent Transportation Systems (ITSC), 2034–2041. <https://doi.org/10.1109/ITSC55140.2022.9922208>
- Lu, C., Krishna, R., Bernstein, M., & Fei-Fei, L. (2016). Visual Relationship Dataset . <https://cs.stanford.edu/people/ranjaykrishna/vrd/>
- Li, Y. et al. (2017) ‘Scene Graph Generation from Objects, Phrases and Region Captions’. <https://doi.org/10.48550/arXiv.1707.09700>
- Lu, C. et al. (2016) ‘Visual Relationship Detection with Language Priors’, in B. Leibe et al. (eds) *Computer Vision – ECCV 2016*. Cham: Springer International Publishing (Lecture Notes in Computer Science), pp. 852–869. https://doi.org/10.1007/978-3-319-46448-0_51
- Maierhofer, S., Moosbrugger, P. and Althoff, M. (2022). Formalization of Intersection Traffic Rules in Temporal Logic. *2022 IEEE Intelligent Vehicles Symposium (IV)*. <https://doi.org/10.1109/iv51971.2022.9827153>
- Maierhofer, S., Rettinger, A.-K., Mayer, E. C., & Althoff, M. (2020). Formalization of Interstate Traffic Rules in Temporal Logic. *2020 IEEE Intelligent Vehicles Symposium (IV)*, 752–759. <https://doi.org/10.1109/IV47402.2020.9304549>
- Manas, K., & Paschke, A. (2023). Semantic Role Assisted Natural Language Rule Formalization for Intelligent Vehicle. *Rules and Reasoning, RuleML+RR 2023*
- Muggleton, S. (1991) ‘Inductive logic programming’, *New Generation Computing*, 8(4), pp. 295–318. <https://doi.org/10.1007/BF03037089>
- Muggleton, S. (1995) ‘Inverse entailment and progol’, *New Generation Computing*, 13(3), pp. 245–286. <https://doi.org/10.1007/BF03037227>
- OpenLink. (2023). VirtSponger. OpenLink Software. <https://vos.openlinksw.com/owiki/wiki/VOS/VirtSponger>
- Payani, A. and Fekri, F. (2019) ‘Inductive Logic Programming via Differentiable Deep Neural Logic Networks’, [arXiv:1906.03523 \[cs\]](https://arxiv.org/abs/1906.03523) [Preprint]. <http://arxiv.org/abs/1906.03523>
- Quine, W.V. (1969) *Ontological Relativity and Other Essays, Ontological Relativity and Other Essays*. Columbia University Press. <https://www.degruyter.com/document/doi/https://doi.org/10.7312/quin92204/html>
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W. and Liu, P.J., 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1), pp. 5485–5551
- RDFLib Team. (2023). RDFLib: Python library for working with RDF, a simple yet powerful language for representing information. [Software]. <https://rdflib.readthedocs.io/>
- Sadeghi, M.A. and Farhadi, A. (2011) ‘Recognition using visual phrases’, in *CVPR 2011*, pp. 1745–1752. <https://doi.org/10.1109/CVPR.2011.5995711>
- Scholtes, M. et al. (2021) ‘6-Layer Model for a Structured Description and Categorization of Urban Traffic and Environment’. [arXiv](https://arxiv.org/abs/2012.06319). <https://doi.org/10.48550/arXiv.2012.06319>
- Sorbonne Paris North University. (2023). OWLready2: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. [Software]. <https://owlready2.readthedocs.io/en/v0.42/>
- Srinivasan, A. (2007). Aleph: A Learning Engine for Proposing Hypotheses. <https://www.cs.ox.ac.uk/activities/programinduction/Aleph/aleph.html>
- Studer, R., Benjamins, V.R. and Fensel, D. (1998) ‘Knowledge engineering: Principles and methods’, *Data & Knowledge Engineering*, 25(1–2), pp. 161–197. [https://doi.org/10.1016/s0169-023x\(97\)00056-6](https://doi.org/10.1016/s0169-023x(97)00056-6)
- Tang, K. et al. (2020) ‘Unbiased Scene Graph Generation from Biased Training’, [arXiv:2002.11949 \[cs\]](https://arxiv.org/abs/2002.11949) [Preprint]. <http://arxiv.org/abs/2002.11949>

- Tao, J., AVL LIST GMBH, FZI Forschungszentrum Informatik, Deepen AI and Edge Case Research, FEV Software and Testing Solutions GmbH, et al (2021).: ASAM OpenX Ontology User Guide (Nov 2021), <https://www.asam.net/standards/asamopenxontology/>
- Ulbrich, S. et al. (2015) ‘Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving’, in *2015 IEEE 18th International Conference on Intelligent Transportation Systems - (ITSC 2015)*, IEEE 18th International Conference on Intelligent Transportation Systems - (ITSC 2015), Gran Canaria, Spain: IEEE, pp. 982–988. <https://doi.org/10.1109/ITSC.2015.164>
- Van Deursen, D., Poppe, C., Martens, G., Mannens, E., & Van de Walle, R. (2008). XML to RDF conversion: A generic approach. In *Proceedings of the 4th International Conference on Automated Solutions for Cross Media Content and Multi-Channel Distribution (AXMEDIS2008)* (pp. [page range, if available]). IEEE
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, 30. <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91f-bd053c1c4a845aa-Abstract.html>
- Wang, P.-W. et al. (2020) ‘Differentiable learning of numerical rules in knowledge graphs.’ ICLR. 2020
- Wang, Y., Grabowski, M. and Paschke, A. (2022) ‘An Ontology-based Model for Handling Rule Exceptions in Traffic Scenes’. <https://research.nii.ac.jp/~ksatoh/WAICOM2022/WAICOMproceedings.pdf#page=93>
- Westhofen, L. et al. (2022) ‘Using Ontologies for the Formalization and Recognition of Criticality for Automated Driving’, [arXiv:2205.01532 \[cs\]](https://arxiv.org/abs/2205.01532) [Preprint]. <http://arxiv.org/abs/2205.01532>
- World Wide Web Consortium (W3C). (2004). OWL Web Ontology Language Overview. <https://www.w3.org/TR/owl-features/>
- World Wide Web Consortium (W3C). (2007). Gleaning Resource Descriptions from Dialects of Languages (GRDDL). <https://www.w3.org/TR/grddl/>
- World Wide Web Consortium (W3C). (2012). R2RML: RDB to RDF Mapping Language. <https://www.w3.org/TR/r2rml/>
- Wörmann, J. et al. (2023) ‘Knowledge Augmented Machine Learning with Applications in Autonomous Driving: A Survey’. [arXiv. http://arxiv.org/abs/2205.04712](https://arxiv.org/abs/2205.04712)
- Xu, D. et al. (2017) ‘Scene Graph Generation by Iterative Message Passing’. <https://doi.org/10.48550/arXiv.1701.02426>
- Yang, F., Yang, Z. and Cohen, W.W. (2017) ‘Differentiable Learning of Logical Rules for Knowledge Base Reasoning’, [arXiv:1702.08367 \[cs\]](https://arxiv.org/abs/1702.08367) [Preprint]. <http://arxiv.org/abs/1702.08367>
- Yang, J. et al. (2018) ‘Graph R-CNN for Scene Graph Generation’. <https://doi.org/10.48550/arXiv.1808.00191>
- Yang, Y. and Song, L. (2020) ‘LEARN TO EXPLAIN EFFICIENTLY VIA NEURAL LOGIC INDUCTIVE LEARNING’, p. 15.
- Zareian, A., Karaman, S. and Chang, S.-F. (2020) ‘Bridging Knowledge Graphs to Generate Scene Graphs’. [arXiv. http://arxiv.org/abs/2001.02314](https://arxiv.org/abs/2001.02314)
- Zellers, R. et al. (2018) ‘Neural Motifs: Scene Graph Parsing with Global Context’, [arXiv:1711.06640 \[cs\]](https://arxiv.org/abs/1711.06640) [Preprint]. <http://arxiv.org/abs/1711.06640>
- Zhang, J. et al. (2019) ‘Graphical Contrastive Losses for Scene Graph Parsing’, [arXiv:1903.02728 \[cs\]](https://arxiv.org/abs/1903.02728) [Preprint]. <http://arxiv.org/abs/1903.02728>
- Zhang, J. et al. (2021) ‘Neural, symbolic and neural-symbolic reasoning on knowledge graphs’, *AI Open*, 2, pp. 14–35. <https://doi.org/10.1016/j.aiopen.2021.03.001>

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Wissensgraphen und maschinelles Lernen im Spannungsfeld juristischer Sprache

15

Christian Dirschl und Christian Hartz

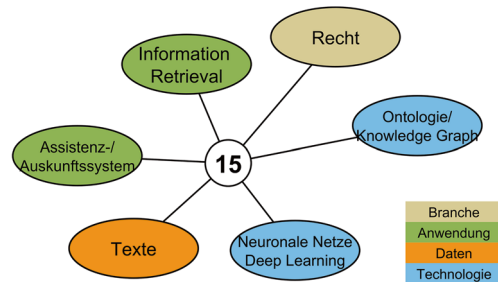
Kernaussagen

- Die juristische Arbeitsweise ist geprägt durch den Umgang mit Text, der oft kompliziert, verklausuliert und mit Fachwörtern gespickt ist.
- Die juristische Domäne hat spezielle Voraussetzungen für KI-Anwendungen, die den Nutzen allgemeiner Machine-Learning-Verfahren stark einschränken.
- Erklärbarkeit und Transparenz sind eine unabdingbare Voraussetzung für viele Anwendungsfälle im juristischen Umfeld und sorgen für die unabdingbare Möglichkeit der Validierung und Verifizierung.
- Die juristische Domäne benötigt einen hybriden Ansatz für viele Fragestellungen
- KI ist ein Gamechanger und führt letztendlich zu einer Semantifizierung der gesamten Organisation.
- Hybride Systeme, die klassisches Information Retrieval, Wissen aus Ontologien und Deep Learning kombinieren, haben in der vorgestellten Lösung die besten Ergebnisse erzielt und sind insbesondere wegen der Notwendigkeit, Transparenz und Erklärbarkeit ins Zentrum fast aller Anwendungen zu stellen, insgesamt vorzuziehen.

C. Dirschl (✉) · C. Hartz
Wolters Kluwer Deutschland GmbH, Hürth, Deutschland
E-Mail: Christian.Dirschl@wolterskluwer.com

C. Hartz
E-Mail: christian.hartz@wolterskluwer.com

Abb. 15.1 Thematische Einordnung des Kap. 15



15.1 Einleitung

Die Anwendung von KI im juristischen Umfeld ist gespickt mit spezifischen Herausforderungen, die sowohl aus der Domäne kommen als auch aus der speziellen Arbeitsweise von Jurist:innen. Wolters Kluwer hat es sich zur Aufgabe gemacht, Rechtsinformationen eingebettet in den Workflow seiner Kund:innen zielgerichtet am Point of Need zur Verfügung zu stellen und den Transformationsprozess der ganzen Branche hin zu einem digitalen Ökosystem maßgeblich zu begleiten und zu forcieren. Wolters Kluwer unterstützt viele Initiativen intern und extern, die sich mit der Anwendung von KI im juristischen Umfeld beschäftigen. Das geht von internen Entwicklerteams über internationale Produktentwicklungen bis hin zu Forschungsprojekten wie DataBri-X.¹

Die Digitalisierung ist ein umfassender Transformationsprozess, der die gesamte Organisation betrifft. Es hat sich gezeigt, dass die umfassende Nutzung von KI einen ähnlich weitreichenden Veränderungsprozess in Gang setzt, den wir als „Semantifizierung der Organisation“ bezeichnen (vgl. Dirschl et. al 2022).

Hybride Systeme, wie sie in Abb. 15.1 für das folgende Kapitel dargestellt sind, werden diesen Prozess weiter befeuern und auch für eine erfolgreiche Nutzung benötigt. Die Kombination aus klassischem Information Retrieval, Wissen aus Ontologien bzw. Wissensgraphen und Deep Learning hat in der hier vorgestellten Lösung die besten Ergebnisse erzielt.

15.2 Die juristische Welt

15.2.1 Text als primärer Informationsträger

Die juristische Arbeitsweise ist geprägt durch den Umgang mit Text, sei es die Erstellung von Schriftsätzen an Parteien eines Rechtsstreits oder in außergerichtlichen Verfahren.

¹ Home – DataBri-X Project.

Gemein ist diesen juristischen Texten, dass die Verwendung von Wörtern im juristischen Bereich sich teilweise von der Verwendung derselben Wörter im alltäglichen Leben unterscheiden kann. So ist bspw. das in der Allgemeinheit als Leihwagen bezeichnete Fahrzeug, das die Werkstatt oder eine Autovermietung entgeltlich zur Verfügung stellt, gar keine Leihe, sondern eine Miete. § 598 BGB i. d. F. v. 01.01.2002, die maßgebliche Vorschrift über die Leihe, geht nämlich von einer Unentgeltlichkeit aus: „Durch den Leihvertrag wird der Verleiher einer Sache verpflichtet, dem Entleiher den Gebrauch der Sache unentgeltlich zu gestatten.“

Auch der Grad an Formalität ist in juristischen Texten oftmals deutlich höher. Schließlich kommt hinzu, dass juristische Texte komplizierter geschrieben, verklausuliert, verschachtelt und mit Fachwörtern gespickt sind.

Letztlich ist auch die Vielfalt an Arten juristischer Texte zu berücksichtigen. Betrachtet man lediglich die häufigeren Dokumentarten, die in einer juristischen Recherche Verwendung finden, ist dies bereits eine ganze Anzahl an Arten:

- Entscheidungen der Gerichte
 - Endurteile, Zwischenurteile
 - Beschlüsse
- Zeitschriftenartikel
 - Besprechungen von Gerichtsentscheidungen
 - Aufsätze
- Handbücher
- Lexika
- Kommentare
- Sonstige Fachbücher
- Gesetze

Diese Dokumentarten unterscheiden sich nach Formalität (bspw. sind Gesetze deutlich formaler als Aufsätze) und Komplexität (Gesetze und Entscheidungen sind teilweise unverständlicher als die erklärenden Kommentare oder Handbücher).

All diese Kombinationen erschweren die Verarbeitung der Texte auch im Rahmen des maschinellen Lernens, sodass allgemeine Sprachmodelle ihre Schwierigkeiten domänenbedingt im juristischen Umfeld haben. Der Einsatz hybrider Systeme, bei denen die juristischen Konzepte mit eingebunden werden, können dieses Problem abmildern.

15.2.2 Daten zur Kontextualisierung und Semantifizierung

Auch sind die Daten in der juristischen Welt sehr unterschiedlich und oft unstrukturiert. Oftmals handelt es sich um Text, insbesondere bei den von Jurist:innen selbst generierten und versendeten Daten. Teilweise werden diese strukturell aufbereitet und semantifiziert,

um auf sie im Rahmen von Rechercheportalen wie Wolters Kluwer-Online zugreifen zu können.

Zusätzlich unterliegen sogenannte Primärdaten wie Gesetze und Gerichtsentscheidungen gewissen strukturellen Konventionen, die man für die Analyse heranziehen kann und muss, um den jeweiligen Kontext identifizieren zu können. Diese strukturellen Elemente lassen sich einerseits manuell oder (halb-)automatisch auszeichnen und somit nutzbar machen. Andererseits dienen sie auch dazu, die für die juristischen Daten typische enge Vernetzung von Informationen durch direkte Verlinkung (s. §x Abs. y Gesetz z) in maschinenlesbarer Form prozessierbar zu machen.

Um dieser Datenvielfalt zu begegnen, werden im professionellen Umfeld Auszeichnungssprachen wie XML verwendet, um die strukturellen Informationen explizit zu machen und um Anreicherungen zuzuordnen zu können. Zusätzlich werden Ontologien, Thesauri und Taxonomien verwendet, um einerseits diese Anreicherungen kontrolliert durchzuführen und andererseits um generelles juristisches Wissen (wie einen juristischen Themenbaum) maschinell verarbeitbar machen zu können.

In der Nutzung der Daten zeigt sich, dass die korrekte Kontextualisierung der Informationen entscheidend für die Akzeptanz der darauf aufbauenden Such-, Navigations- oder Visualisierungsfunktionen ist. Dies bedeutet, dass eine hohe Integrierung von Strukturinformationen und semantischer Auszeichnung anzustreben ist.

Dies führt letztendlich zu einer sehr granularen Sichtweise auf die Daten. Die Granularität wird teilweise bis auf Satzebene heruntergebrochen, da z. B. in einem Absatz in einer Gerichtsentscheidung jeder Satz eine andere semantische Funktion haben kann und diese semantische Funktion entscheidend für die korrekte juristische Analyse des Textes ist.

So erkennt man im folgenden Beispiel der Annotation einer Entscheidung des Landesarbeitsgerichts Hessen (Abb. 15.2), dass der erste Satz – in rot – die eigentliche Entscheidung des Gerichts beinhaltet. Der zweite Satz (erster Halbsatz) – in grün bzw. die Hervorhebung der Zitierungen in gelb – ist eine abstrakte Rechtsausführung, die leicht auf andere (ähnliche) Verfahren übertragen werden kann; dieser enthält auch einige zusätzliche Informationen wie Verweise auf das Bürgerliche Gesetzbuch (BGB) oder auf Sekundärliteratur (hier zwei Zeitschriften und ein Handbuch). Der zweite Satz (zweiter Halbsatz) – in blau – („zwischen den Parteien aber nicht wirksam erfolgt“) hin-

3.

Die gesetzliche Regelung, wonach der Arbeitgeber die notwendigen Betriebsmittel zu stellen hat, wurde zwischen den Parteien nicht wirksam abbedungen. Eine von §§ 611a, 615 S. 1, 618 BGB abweichende vertragliche Vereinbarung der Arbeitsvertragsparteien dergestalt, dass der Arbeitnehmer die Betriebsmittel einzubringen hat, ist zwar grundsätzlich zulässig (vgl. etwa Däubler, *AGB Kontrolle im Arbeitsrecht*, Anhang Rz. 133; Zöll/Kielkowski *BB* 2012, 2625; Seel *MDR* 2014, 69), zwischen den Parteien aber nicht wirksam erfolgt.

Abb. 15.2 Manuelle Annotation eines Absatzes des Urteils des LAG Hessen, v. 19.02.2021, Az.: 14 Sa 306/20

gegen ist die Anwendung von Halbsatz 1 auf den konkreten Fall mit der Information, dass selbst wenn dies gelte, es gerade im vorliegenden Fall nicht wirksam eingebunden wurde. Sucht bspw. ein Nutzer nach der Frage, ob notwendige Betriebsmittel immer vom Arbeitgeber gestellt werden müssen, so liefert der zweite Satz Halbsatz 1 die passende Information. Für den Ausgang dieses konkreten Verfahrens ist allerdings Satz 2 Halbsatz 2 relevanter.

Die Semantifizierung der Daten macht letztendlich das implizite Wissen, das in komplexen und verklausulierten juristischen Formulierungen steckt, explizit und maschinenlesbar. Es hat sich allerdings gezeigt, dass die Sichtbarmachung der juristischen Inhalte in den Texten selbst nicht ausreichend ist. Der Nutzer braucht bei der Analyse der Daten sehr viel mehr Hintergrundwissen, um die richtigen Schlussfolgerungen zu ziehen. Somit beinhaltet die Semantifizierung neben einer granularen einheitlichen juristischen Auszeichnung der Daten auch die Erstellung und Nutzung von juristischem Wissen unabhängig von den Dokumenten selbst. Dies kann im einfachen Fall die direkte Nutzung der Struktur des juristischen Themenbaums sein, erweitert sich aber sehr schnell um die Modellierung ganzer juristischer Prozesse oder Regelwerke. Ein einfaches Beispiel hierfür wäre die Modellierung des sogenannten Instanzenzugs, also des gesetzlich definierten Ablaufs eines Gerichtsverfahrens über die verschiedenen Instanzen hinweg (Abb. 15.3). In jeder Instanz werden bestimmte Aspekte des Gerichtsverfahrens abgearbeitet.

Amtsgerichte und Landgerichte stellen die Eingangsinstanz dar, jeweils abhängig von der Höhe des Streitgegenstandes oder einer bestimmten Art der Streitigkeit. Sofern

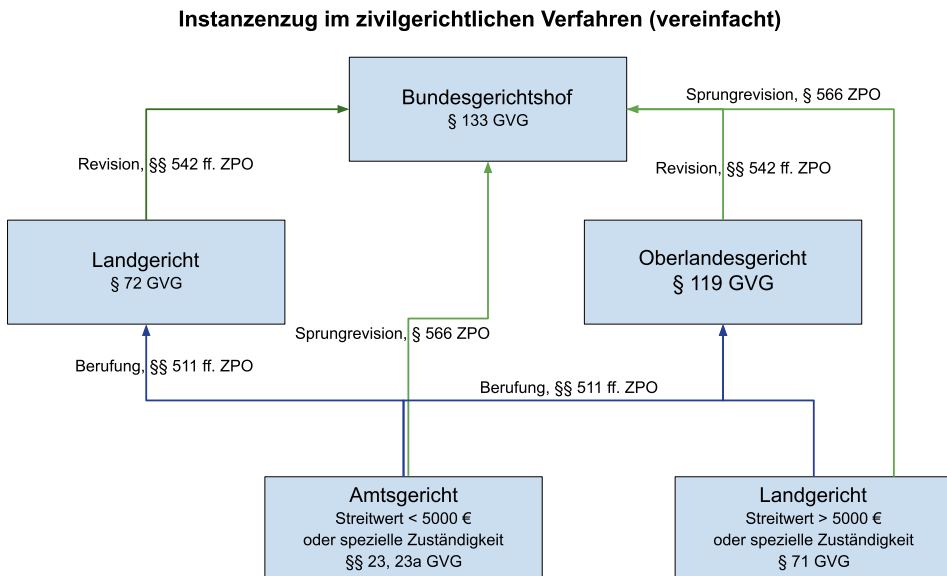


Abb. 15.3 Vereinfachte Visualisierung des Instanzenzugs im zivilgerichtlichen Verfahren

die Voraussetzungen für eine Berufung oder Revision gegeben sind, können höhere Gerichte mit dem Streitstand beschäftigt werden. Hierbei befasst sich die Revision (grüne Linien) in der Regel lediglich mit rechtlichen Fragestellungen, wobei der von der Eingangsinstanz festgestellte Sachverhalt nur unter ganz bestimmten Gründen noch einmal angepasst werden kann. Die Berufung (blaue Linien) hingegen kann auch als Tatsacheninstanz verwendet werden, sodass hier der Sachverhalt anders bewertet werden kann.

In Bezug auf das Bild der Annotation oben (Abb. 15.2) heißt das, dass die blauen Sätze in den Entscheidungen der Revision ggf. gar nicht vorhanden sind, sondern lediglich Informationen der grünen Sätze vom Gericht verarbeitet werden. Das bedeutet aber auch gleichzeitig, dass, um die Entscheidung gänzlich verstehen zu können, die Informationen aus der Vorinstanz Berücksichtigung finden müssen.

Analysiert und semantifiziert man somit nur die höchstrichterlichen Entscheidungen, werden bestimmte Aspekte komplett unter den Tisch fallen, da sie nur in den Vorinstanzen verschriftlicht wurden.

15.2.3 Juristische Prozesse

Viele Prozesse im juristischen Umfeld sind grundsätzlich strukturiert, jedoch werden nicht alle durch IT-Systeme vorgegeben oder durchgeführt. Einen großen Einfluss hat das Zusammenspiel zwischen Jurist:innen und juristischem Personal. Betrachten wir bspw. das Fristenmanagement, so ist es oftmals der Fall, dass die Fristenkalender von Rechtsanwaltsfachangestellten geführt werden, die jeden potenziellen Fristablauf verzeichnen und den Rechtsanwält:innen vorlegen, damit sie ihre Arbeit entsprechend einplanen können. In vielen Fällen wird das Fristenbuch noch in Papierform gepflegt und vorgelegt.

Um all diese mehr oder weniger digitalen Vorgänge sukzessive gänzlich digital zu überführen, müssen diese Prozesse digital abbildbar und einfach sein. Das Erkennen von Parteien, Datum und möglichem Fristende durch eine Kombination aus Named Entity Recognition und hinterlegtem juristischen Wissen (bspw. wie lange die Notfrist i. S. d. § 517 ZPO für die Einlegung der Berufung nach Zustellung des Urteils ist) hilft hier bei der Digitalisierung.

15.2.4 Juristische Use Cases

Bevor wir uns einzelne Use Cases etwas genauer ansehen, sei ein Querschnittsthema angesprochen, das sich durch alle diese Use Cases zieht: Erklärbarkeit. Jurist:innen sind darauf angewiesen, die passende Information genau dann zu erhalten, wenn sie sie be-

nötigen. Daneben müssen sie aber auch für das Ergebnis ihrer Arbeit gerade stehen und unterliegen den entsprechenden Haftungsnormen. Explainability ist Voraussetzung, um es den Jurist:innen zu ermöglichen, die gefundenen Ergebnisse zu validieren und zu verifizieren.

Typische Use Cases, bei denen maschinelles Lernen mit einem hybriden Ansatz bei Wolters Kluwer zum Einsatz kommt bzw. kommen kann, sind (Abb. 15.4):

- Bearbeitung und Informationsgewinnung aus nutzergenerierten Informationen (bspw. Aufbereitung aller Dokumente in einem juristischen Fall),

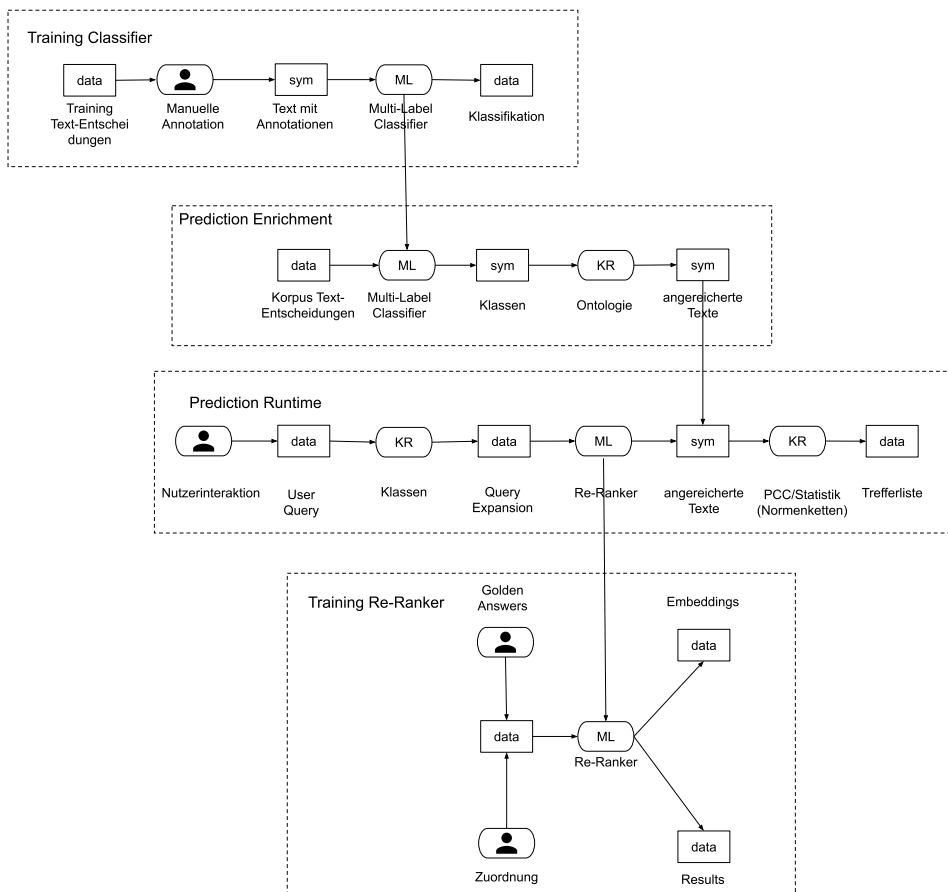


Abb. 15.4 Boxology-Modell des hybriden KI-Systems

- Aufbereitung und Auswertung von Entscheidungen der deutschen, europäischen und internationalen Gerichte,
- Erstellen bzw. Erschließen von Sekundärliteratur wie Handbücher, Kommentare, Zeitschriften, bspw. durch automatisches Verlinken passender Zusatzinformationen etc.,
- Erstellen von juristischen Auskunftssystemen (Chatbots auf Basis von Frage-Antwort Systemen),
- Unterstützung bei der Erstellung von Gesetzesentwürfen,
- Validierung der Umsetzung von europäischem Recht in nationales Recht,
- Validierung der Einheitlichkeit der nationalen Rechtsprechung.

15.3 Übersicht über das hybride System

Legal Analytics beinhaltet aus Boxology-Sicht² vier Komponenten.

1. das Training des Classifier, bei dem die Rechtsprechungsdokumente manuell Satz für Satz mittels eines semantischen Modells annotiert werden,
2. das Training des Re-Ranker, bei dem mittels von Subject Matter Experts erstellten Golden Answers ein Modell trainiert wird, das in der Lage ist, die gewünschten Entscheidungen zu finden,
3. die Prediction im Rahmen des Enrichment, bei der der gesamte juristische Korpus mittels des Classifier annotiert wird,
4. die Prediction zur Laufzeit, bei der die Sucheingabe des Nutzers erweitert und passende Ergebnisse zurückgeliefert werden.

15.4 Juristische Informationssysteme – Status und Ausblick

Juristische Informationssysteme sind darauf ausgerichtet, den Jurist:innen Informationen einfach zur Verfügung zu stellen. Dabei sollen sie helfen, juristische Probleme zu lösen. Grundsätzlich können hier zwei verschiedene Szenarien unterschieden werden:

- Arbeit an einem Fall,
- Informationsgewinnung, um auf dem aktuellen Stand zu sein bzw. zur Fortbildung.

Anwält:innen sind gemäß § 43 BRAO angehalten, ihren Beruf gewissenhaft auszuüben. Als weitere Grundpflicht normiert § 43a VIII BRAO eine Fortbildungspflicht. Daher müssen Rechtsanwält:innen sich regelmäßig über die aktuelle Rechtsprechung informie-

²Für eine Einführung in die Boxology siehe Kap. 1.

ren. Dies ist nach der derzeitigen Rechtsprechung auch gänzlich analog möglich, wobei die Zeitgemäßheit angezweifelt werden kann (vgl. Hartz 2022).

Bei der Arbeit an einem Fall geht es darum, Informationen zur Lösung des Falles bzw. zur Argumentation zu finden und entsprechend aufzubereiten. Es geht also nicht um die generelle Informationsbeschaffung, sondern um einen konkreten Bezug und eine Lösung genau dieses Falls. Derzeit findet die Aktenarbeit oftmals analog, die Recherche jedoch digital statt.

Die juristischen Informationssysteme setzen dabei auf klassische Suchtechnologien, die mit Elementen künstlicher Intelligenz bzw. maschinellem Lernen erweitert werden können. So ist Solr oder Elasticsearch meist die Basis der Suchkomponente (Abb. 15.5).

Synonyme, Taxonomien und die Erkennungskomponenten für juristische Informationen können gemeinsam im ersten Schritt genutzt werden, um die Benutzereingabe zu erweitern (sog. Query Expansion, vgl. 1 in Abb. 15.5).

Der Inhalt, der aus Rechtsprechung und Gesetzen (sog. Primärcontent) auf der einen Seite und Kommentaren, Handbüchern etc. (sog. Sekundärcontent) auf der anderen Seite besteht, wird mit entsprechenden Metadaten gespeichert (vgl. 2 in Abb. 15.5).

Im dritten Schritt der Suche (3 in Abb. 15.5) findet neben dem klassischen Retrieval auch ein Re-Ranking statt. Dies kann ganz klassisch ein statistischer Ansatz wie das TF-IDF-Maß oder Okapi BM25 sein, bei dem spezifische Metadaten zusätzlich höher gewichtet werden können.

Abb. 15.6 zeigt den Einsatz von maschinellem Lernen und Wissensgraphen in der Legal-Analytics-Applikation im juristischen Umfeld und gleichzeitig auch eine Integration in den Workflow, wie bspw. Outlook, Word oder Adobe Acrobat. Die Machine-Learning-Komponenten unterstützen hier entweder bei einer zusätzlichen Query Expansion, im Rahmen der weiteren Aufbereitung des Contents und/oder bei der Extraktion von und der Arbeit mit Argumenten. Die einzelnen Aufgaben, für die maschinelles Lernen hier gemeinsam mit einem Wissensgraphen zum Einsatz kommt, sollen nachfolgend beschrieben werden.

Die hier vorgestellte Lösung (Abb. 15.7) findet sich als Wolters Kluwer Analytics in einer Beta-Applikation.³

Durch die oben in Abb. 15.6 vorgestellte Technologie, ist es den Jurist:innen einfach möglich, eine natürlichsprachliche Eingabe durchzuführen, und die passenden Gesetze und Entscheidungen werden gefunden. Dies stellt jedoch lediglich einen Anwendungsbereich von hybrider KI im juristischen Umfeld dar. Die weiteren Anwendungsfelder sollen im Folgenden beleuchtet und der Einsatz von hybrider KI hierbei erklärt werden.

³Die Beta Applikation kann zum Zeitpunkt des Verfassens des Kapitels unter <https://www.wolterskluwer-analytics> abgerufen werden.

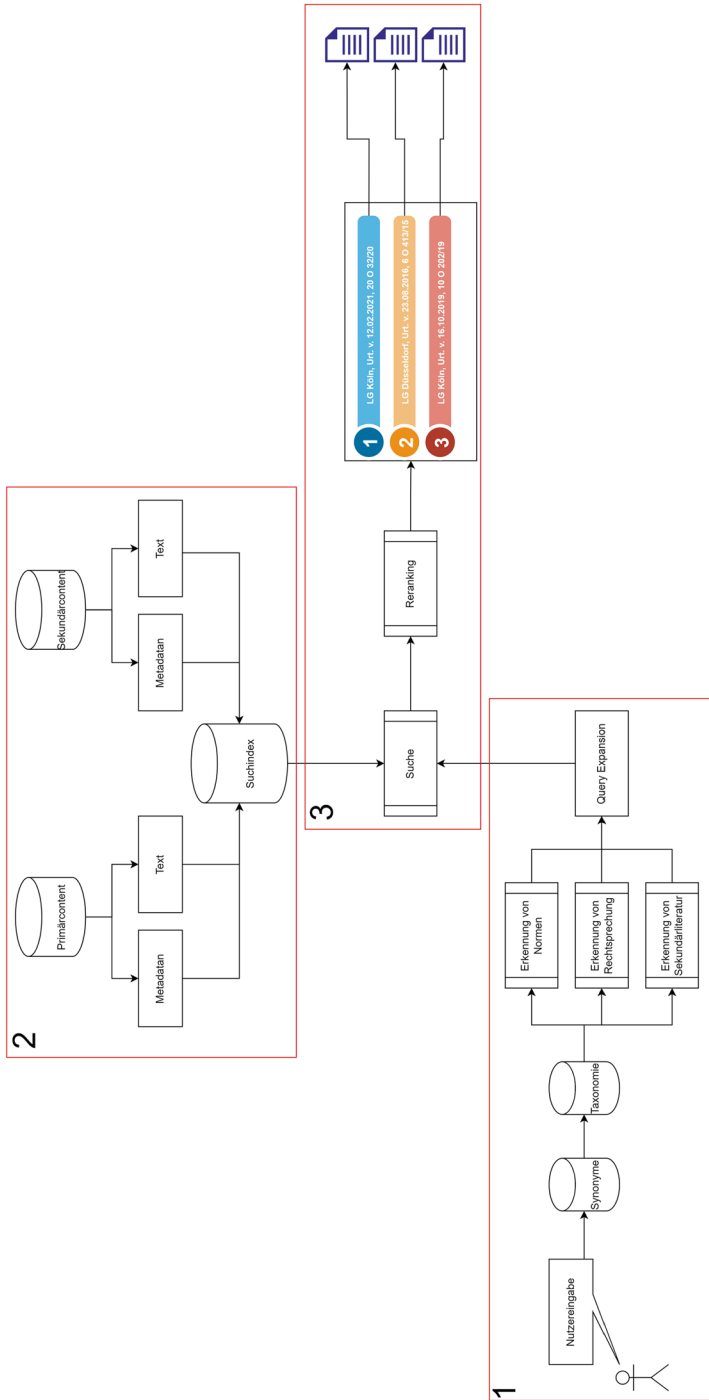


Abb. 15.5 Schema klassische juristische Recherche (vgl. Hartz 2022)

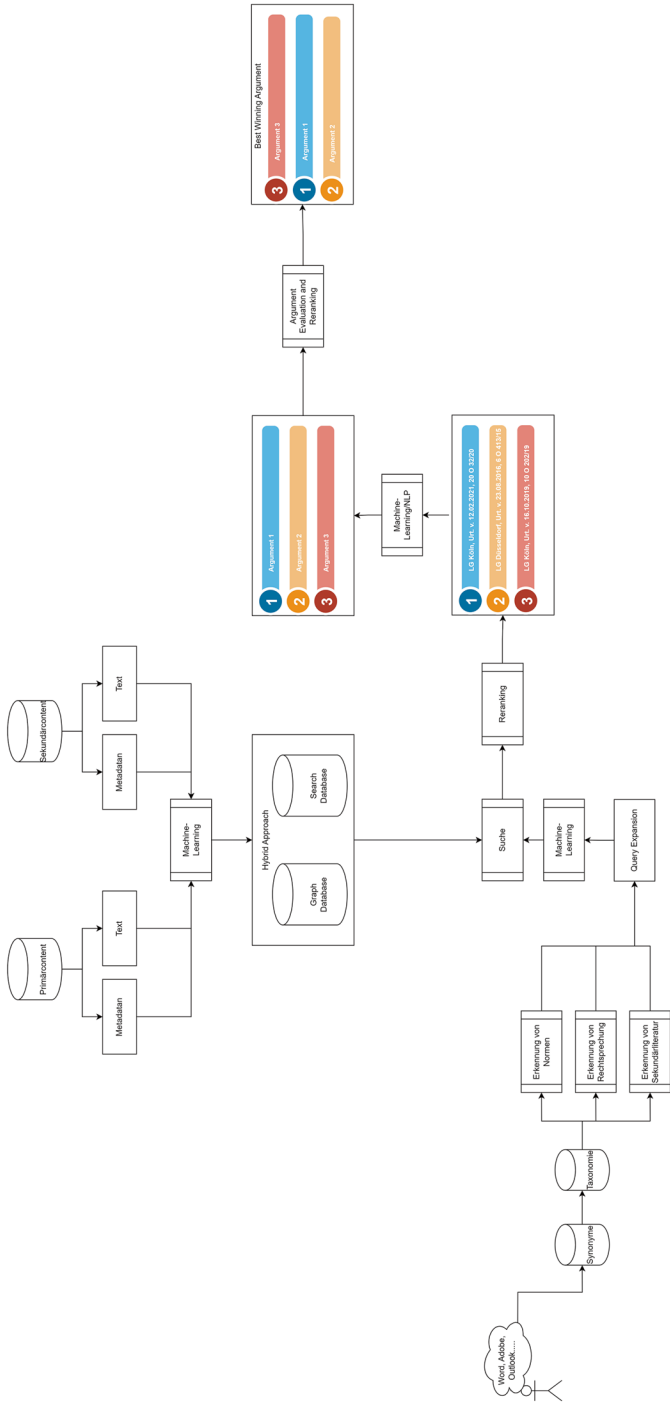


Abb. 15.6 Schema hybrider Ansatz im Rahmen einer Legal-Analytics-Lösung

Legal Analytics beta
Mein Wolters Kluwer Online
christian.hartz@wolterskluwer.com

IHRE FALLBESCHREIBUNG Alles anzeigen | Bearbeiten

Ich als Käufer erstellte bei eBay einen Computer für 1€. Verkäufer möchte den Computer nicht für 1€ versenden und fricht den Vertrag an. Ich möchte Schadensersatz

RECHTSNORMEN (AUS DEN ENTSCHEIDUNGSGRÜNDEN)	Alle anzeigen
§ 119 BGB - Anfechtbarkeit wegen Irrtums <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 281 BGB - Schadensersatz statt der Leistung wegen ni... <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 280 BGB - Schadensersatz wegen Pflichtverletzung <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 145 BGB - Bindung an den Antrag <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 138 BGB - Sittenwidriges Rechtsgeschäft; Wucher <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 433 BGB - Vertragstypische Pflichten beim Kaufvertrag <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 437 BGB - Rechte des Käufers bei Mängeln <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 122 BGB - Schadensersatzpflicht des Anfechtenden <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 142 BGB - Wirkung der Anfechtung <input type="checkbox"/>	<input checked="" type="checkbox"/>
§ 434 BGB - Sachmangel <input type="checkbox"/>	<input checked="" type="checkbox"/>
Norm auswählen <input type="button" value="⊕"/>	
IN DEN ENTSCHEIDUNGSGRÜNDEN GEFUNDENE WEITERE RECHTSPRECHUNG	<input checked="" type="checkbox"/>
BGH, 08.06.2011 - VIII ZR 305/10 <input type="checkbox"/>	

Übersicht

14 ENTSCHEIDUNGEN, SORTIERT NACH 16 NORMENKOMBINATIONEN

Filter
Report

1. § 275 BGB - Ausschluss der Leistungspflicht | § 119 BGB - Anfechtbarkeit wegen Irrtums

Zeige alle 2 Entscheidungen mit dieser Normenkette

KG Berlin, 14.04.2020 - 18 U 19/19

Auch zu etwaigen gesetzlichen Rücktritts- oder Abbruchgründen wie sie sich etwa aus §§ 119 ff. oder 275 ff. BGB ergeben konnten, hat er nichts vorgetragen.
2. § 119 BGB - Anfechtbarkeit wegen Irrtums | § 323 BGB - Rücktritt wegen nicht oder nicht vertragsgemäß erbrachter Leistung

Zeige alle 2 Entscheidungen mit dieser Normenkette

BGH, 23.09.2019 - VIII ZR 284/14

Als ein solcher berechtigter Grund, der den in den eBay-Bedingungen ausdrücklich genannten Beispielen vergleichbar ist, kommen jedoch aufgrund des grundsätzlichen Verweises in § 9 Nr. 11 der eBay-Bedingungen auf die gesetzliche Berechtigung zur Angebotsstreichung nur derartige Umstände in der Person des Bikers in Betracht, die Umstände vergleichbar sind, die zur Anfechtung des Angebots (§§ 119 ff. BGB) oder zum Rücktritt vom Vertrag (§ 323 Abs. 4 BGB) führen würden.
3. § 119 BGB - Anfechtbarkeit wegen Irrtums | § 463 BGB - Voraussetzungen der Ausübung | § 459 BGB - Ersatz von Verwendungen

Zeige alle 1 Entscheidungen mit dieser Normenkette

BGH, 16.03.1973 - V ZR 118/71

Vielmehr bestimmt sich die Haftung des Verkäufers für Eigenschaften der Sache - sieht man von dem Falle eines Mangelfolgebahrens an anderen Beschäftigten des Käufers (positive Vertragsverletzung) ab - allein nach den Gewährleistungsvorschriften der §§ 459 ff. BGB; der Käufer kann seine Vertragsklärung nach Gefährdung nicht mehr wegen Irrtums über eine Eigenschaft der Kaufsache anfechten (§ 119 Abs. 2 BGB) und sich nicht darauf berufen, daß eine bestimmte Eigenschaft für beide Teile Geschäftsgrundlage des Vertrages gewesen sei (BGZ 135, 339, 346; BGHZ 34, 32; vgl. auch BGH WM 1971, 1016). Schadensersatz sieht § 463 BGB nur bei unrichtiger Zusicherung und bei arglistigen Verschweigen oder arglistiger Vorpreislegung von Sacheigenschaften vor.
4. § 433 BGB - Vertragstypische Pflichten beim Kaufvertrag | § 440 BGB - Besondere Bestimmungen für Rücktritt und Schadensersatz | § 281 BGB - Schadensersatz statt der Leistung wegen nicht oder nicht wie

Abb. 15.7 Wolters Kluwer Analytics, alle Rechte vorbehalten

15.5 KI-Anwendungsfelder

15.5.1 Überblick – Maschinelles Lernen und Wissensgenerierung

Grundsätzlich lassen sich die folgenden Bereiche für den Einsatz von maschinellem Lernen im juristischen Umfeld herausarbeiten:

- Klassifikationsaufgaben,
- Extraktionsaufgaben,
- Clustering-Aufgaben,
- Analyseaufgaben,
- Ranking und Re-Ranking.

Aber auch beim Erstellen und Pflegen von Wissensrepräsentationen, dem Erzeugen des Wissensgraphen und der Repräsentation des sonstigen Domänenwissens kann maschinelles Lernen Verwendung finden.

- Erkennen und Extrahieren neuer Themenfelder der Dokumente
- Automatische Zuordnung der Dokumente zu existierenden Themenfeldern
- Automatische Erzeugung einer Teilbaumstruktur
- Entfernen veralteter Topics
- Verknüpfung von Wissensgraphen auf Basis von Kookkurrenzen
- Repriorisierung von Aktualität

Zum Dritten erzeugt auch die umgekehrte Vorgehensweise notwendige Synergien, damit hybride Systeme besser sind als Einzelsysteme. Maschinelles Lernen profitiert von Wissensgraphen und Domänenwissen in vielfältiger Hinsicht. Neben der bereits erwähnten Explainability und Transparenz kann (und ist in unserer Lösung) Domänenwissen in jedem Schritt des maschinellen Lernens hilfreich sein.

1. **Datenbereitstellung:** Bereits bei der Datenbereitstellung wird Domänenwissen z. B. in Form von Wissensgraphen herangezogen werden. Dieses Wissen, also bspw. die regionale Verteilung von Gerichten, wird direkt als eigenes Datenmodell in den Lernprozess integriert. Ebenso dient es dazu, die richtigen Daten für die Modellierung auszuwählen, beispielsweise gemäß der Gerichtshierarchie.
2. **Datenaufbereitung:** Auch bei der Datenaufbereitung ist Domänenwissen hilfreich. Einerseits können die Daten direkt mit Zusatzinformationen aus dem Domänenwissen angereichert werden, beispielsweise Klassifikationen oder Querverweise zu anderen Dokumenten. Andererseits können Wissensgraphen helfen, eine homogene Datenbasis zu kreieren, damit am Ende das Modell für alle vorgesehenen Use Cases auch gleich gute Ergebnisse liefern kann (z. B. Erkennung der Unterrepräsentation juris-

tischer Fachthemen oder Datenbias bedingt durch regionale Verteilung der Gerichtsentscheidungen).

3. **Datenaufteilung:** Auch hier wird Domänenwissen zur weiteren Homogenisierung bei der Aufteilung in Trainings- und Testdaten herangezogen. Einerseits wird eine Überrepräsentanz bestimmter Gerichte verhindert, um eine Balance zu halten, andererseits wird hierdurch auch eine Überrepräsentanz bspw. hinsichtlich des Strafmaßes eliminiert. Dies allerdings nur, solange man nicht genau diese Abhängigkeit im Modell berücksichtigt haben will.
4. **Modelltraining:** Im Modelltraining können die Wissensgraphen entweder integriert in die Trainingsdaten oder als zusätzliche Trainingsmodelle integriert werden, um ein konsistentes und qualitativ besseres Gesamtmodell erzeugen zu können. In unserem Fall werden sie als Feature verwendet. Somit können schon vorab bekannte Beziehungen explizit gemacht werden oder ansonsten unerwünschte und trotzdem wirksame Abhängigkeiten abgemildert werden.
5. **Modellvalidierung:** Hier kommt der Explainability-Aspekt zum Tragen. Aber auch die Generierung von Golden Answers, also Daten, die gerade zur Überprüfung des Modells erzeugt werden, wird durch Domänenwissen nachhaltig unterstützt. Daneben gibt die Nutzung von Wissensgraphen als Datenmodelle bis zu einem gewissen Grad direkten Einblick in die Funktionsweise des Algorithmus.

Der Prozess des maschinellen Lernens kann von Wissensgraphen und Domänenwissen in vielfältiger Weise profitieren. Entscheidend sind die gute Qualität der Wissensgraphen und die adäquate Struktur zur direkten Nutzung in einem hybriden System.

15.5.2 Anwendungsfälle durch Semantifizierung der Inhalte

- **Klassifikationsaufgaben:**

Typische Klassifikationsaufgaben im juristischen Bereich sind bspw. die Klassifikation von Sätzen in Entscheidungen, aber auch die Klassifikation von Dokumenttypen, wie bspw. Beschluss, Zwischenurteil und Endurteil.

Beispiele für Sätze sind:

Handelt es sich

- um einen Anspruch, den eine Partei vorgebracht hat, oder
- um die Entscheidung des Gerichts;
- handelt es sich um Fakten, die vorgetragen wurden, oder
- um eine rechtliche Einordnung durch das Gericht.

Hierbei kann die Modellierung der verschiedenen Dokumenttypen vorgenommen werden, um anschließend die entsprechenden Klassen zu befüllen.

- **Extraktionsaufgaben:**

Extraktion meint hier unter anderem auch das Erkennen von Zitierungen wie Normen, Entscheidungen oder Sekundärliteratur, um sie anschließend weiter verwenden zu

können. Klassische Named Entity Recognition, also das Erkennen von Personen, Organisationen oder Orten, ist hier nicht ausreichend oder oftmals aufgrund der Anonymisierung von Rechtsprechungsdokumenten unmöglich. Aber auch weitergehende Informationen über das Dokument selbst, wie bspw. Gericht, Datum oder Aktenzeichen bzw. beteiligte Parteien, können darunter fallen.

Daneben kann sich die Extraktion auch auf Informationen wie Definitionen oder juristische Konzepte, die bereits in der bestehenden Taxonomie zu finden sind, beziehen. Gerade das Erkennen neuer juristischer Konzepte zur späteren Aufnahme in das Domänenwissen kann bei der Weiterentwicklung des bereits vorhandenen Wissens helfen.

Diese Extraktionsaufgaben beschränken sich nicht nur auf die bereits in der Datenbank vorhandenen Daten, sondern fallen auch bei der Bearbeitung der Benutzereingabe in der Query an. Insbesondere bei natürlichsprachlicher Sucheingabe ist es hilfreich, juristische Konzepte bereits in der Fragestellung des Nutzers zu erkennen, um sie in der dahinterliegenden Suche bevorzugt berücksichtigen zu können. Die Anwendung von Wissensrepräsentationen hilft hierbei, breitere, engere oder alternative Namen für die Konzepte zu finden.

Beispielhaft steht die Abkürzung a. l. i. c. für „actio libera in causa“, was ein Ausnahmetatbestand zur Schuldunfähigkeit nach § 20 StGB sein kann und Unterformen der fahrlässigen und der vorsätzlichen a. l. i. c. haben kann; eine Möglichkeit, den Defektzustand hervorzurufen, kann ein Vollrausch aufgrund des Alkoholkonsums sein, sodass auch hierzu eine Beziehung in der Ontologie hergestellt wird. All diese Informationen können bei der zugrunde liegenden Suche Verwendung finden, selbst wenn das Konstrukt der a. l. i. c. vom Nutzer nicht eingegeben, sondern lediglich umschrieben wird.

- **Clustering-Aufgaben**

Ähnliche Dokumente, ähnliche Sachvorträge, ähnliche geltend gemachte Ansprüche, all diese Informationen können u. a. mit Clustering bewältigt werden.

Im juristischen Umfeld ist dies auch dafür relevant, ähnliche Entscheidungen zu finden. Gemeinsam mit den zusätzlichen Informationen in den Metadaten und/oder im Wissensgraphen können solche Cluster noch besser ausgewählt werden. Oftmals sind diese Cluster allerdings nicht sehr aussagekräftig und die Inhalte müssen zusätzlich erschlossen werden, um herauszufinden, ob das Cluster für die weitere Verarbeitung sinnvoll erscheint. Beispielsweise ist das Ergebnis einer natürlichsprachigen Suche eine Menge an Dokumenten, bei denen unterschiedliche Anträge der Parteien gestellt wurden. Gegebenenfalls findet man 4–5 verschiedene große Cluster, die dieselben Anträge beinhalten. Wenn ein solcher Antrag allerdings 2–3 Sätze beinhaltet und gleichzeitig 200 Dokumente zu lesen sind, bevor man den Inhalt gänzlich durchdringen kann, sind diese Cluster kaum nutzbar. Zur besseren Übersichtlichkeit können die im Wissensgraphen oder in sonstigen Wissensrepräsentationen gespeicherten Informationen verwendet werden, um dem Cluster einen sprechenden Namen und ggf. wertvolle Zusatzinformationen zu geben.

- **Analyseaufgaben**

Jurist:innen, insbesondere Rechtsanwält:innen, unterliegen verschiedenen Haftungsnormen. Da es bei juristischen Streitigkeiten oftmals um hohe Streitwerte geht, hat der Gesetzgeber eine Pflicht-Haftpflichtversicherung für Rechtsanwält:innen vorgesehen.

Es ist Aufgabe der Jurist:innen, die aktuelle Rechtsprechung zu kennen und sie entsprechend anzuwenden. Verlieren Mandant:innen, weil Rechtsanwält:innen die Rechtsprechung nicht ordnungsgemäß angewandt haben, so können diese in Regress genommen werden. Dies ist insbesondere dann der Fall, wenn die Rechtsprechung nicht ordnungsgemäß recherchiert wurde.

Bei einer Suche zu einem juristischen Problem kommen aber bereits schnell mehrere Hundert Dokumente zusammen, die durchgesehen werden müssen, um sicherzustellen, dass wichtige Informationen nicht übersehen werden.

Eine Möglichkeit, die Zeit zur Durchsicht dieser Dokumente zu verkürzen, ist es, verschiedene Analysemöglichkeiten anzubieten. Sei es bspw. ein Herunterbrechen auf nur ganz bestimmte Ansprüche, die geltend gemacht werden (bspw. nur Schadensersatz aufgrund der Vertragsverletzung und nicht zusätzlich auch wegen eines deliktsrechtlichen Anspruchs), oder sei es das Einschränken auf die Fälle, bei denen die Beklagte gewonnen hat.

Grundlage für diese Analyse sind die bereits zuvor genannten Anwendungsfälle, wie bspw. die Extraktions- oder Klassifikationsaufgaben. Abb. 15.8 zeigt die Klassifikation der Sätze einer Entscheidung, mit denen Ansprüche geltend gemacht werden. Hierdurch kann die Entscheidung deutlich schneller und zielführender gelesen werden.

Gleichzeitig kann aber auch mit Darstellungsmöglichkeiten wie einem Graphen gearbeitet werden. So forscht Wolters Kluwer an möglichen Verwendungen von visuellen Elementen wie Graphen für die Analyse in der juristischen Fallbearbeitung (Abb. 15.9; vgl. dazu auch Nestsiarenia und Hartz 2023).

Eine weitere Möglichkeit in der Analyse ist auch das Herunterbrechen auf kleinere Informationselemente; hier seien insbesondere die Argumente genannt. In der Regel folgen juristische Texte, insbesondere aber juristische Entscheidungen, einer speziellen Logik bzw. einem speziellen Aufbau. Dieser Urteilsstil fängt mit dem Ergebnis an, das vorweg gestellt wird. Dann folgt die Begründung, wobei zunächst abstrakt definiert/erklärt wird und sodann dies auf den vorliegenden Fall angewendet wird. Innerhalb der Begründung werden sowohl das Ergebnis tragende Argumente vorgetragen als auch Argumente, die dem Ergebnis entgegenstehen und die sogleich entkräftet werden. Aufgrund dieser Struktur ist es für die Jurist:innen besonders interessant, zu wissen, welches Argument durch das Gericht abgelehnt und welches angenommen wurde. Auch wenn es sich im kontinentaleuropäischen Rechtssystem gerade nicht um Fallrecht handelt, spielen die Entscheidungen der Gerichte in der täglichen Arbeit dennoch eine große Rolle, insbesondere wenn es sich um obergerichtliche Rechtsprechung handelt.

Legal Analytics an

← Übersicht > BGH, 23.09.2015 - VIII ZR 284/14

Mein Walters Kluwer Online Report

christian.klatz@walterskluwer.com

Dokumentenübersicht

Normenketten in dieser Entscheidung Ansprüche

Entscheidungen Alle anzeigen

Der Kläger begeht Schadensersatz wegen einer vorzeitig abgebrochenen eBay-Auktion.

Mit der Klage nimmt der Kläger den Beklagten auf Zahlung von 3.888 € sowie vergerichtlicher Anwaltskosten in Höhe von 402,82 €, jeweils nebst Zinsen, in Anspruch.

Mit der vom Berufungsgericht zugelassenen Revision verfolgt der Kläger sein Zahlungsbegehren weiter.

zur die Revision des Klägers wird das Urteil der 4. Zivilkammer des Landgerichts Neuruppin vom 24. September 2014 aufgehoben.

Die Sache wird zur neuen Verhandlung und Entscheidung, auch über die Kosten des Revisionsverfahrens, an das Berufungsgericht zurückverwiesen.

Tatbestand

1 **Der Kläger begeht Schadensersatz wegen einer vorzeitig abgebrochenen eBay-Auktion.**

2 Der Beklagte bot am 26. Juni 2012 in Rahmen einer Auktion über die Internetplattform ebay unter Angabe eines Startpreises von 1 € einen Jugendstilschleierkörper zum Verkauf an. Die Versteigerung erfolgte auf der Grundlage der zu dieser Zeit maßgeblichen Allgemeinen Geschäftsbedingungen von ebay. Dort heißt es auszugsweise:

„§ 9 Nr. 11 Anbieter, die ein verbindliches Angebot auf der ebay-Webseite einstellen, dürfen nur dann Gebote streichen und das Angebot zurückziehen, wenn sie gesetzlich dazu berechtigt sind. Weitere Informationen.“

§ 10 Nr. 7 Bieter dürfen ein Gebot nur dann zurücknehmen, wenn sie gesetzlich dazu berechtigt sind. Weitere Informationen.“

3 In den von ebay eingerichteten "Hilfeseiten" zu dem Stichwort "Wie beende ich mein Angebot vorzeitig" heißt es (auszugsweise):

"Wenn Sie einen Artikel auf der ebay-Webseite einstellen, geben Sie grundsätzlich ein verbindliches Angebot zum Abschluss eines Vertrages über diesen Artikel ab und sind für die Angebotsdauer dieses Angebots gebunden. Es kann jedoch vorkommen, dass Sie ein Angebot vorzeitig beenden müssen, zum Beispiel, wenn Sie feststellen, dass Sie sich beim Einstellen des Artikels geirrt haben oder der zu verkaufende Artikel während der Angebotsdauer ohne ihr Verschulden beschädigt wird oder verlorengeht.

Gründe für die vorzeitige Beendigung eines Angebots

Wenn Sie ein Angebot vorzeitig beenden oder kurz vor dessen Ende Änderungen vornehmen, werden Käufer möglicherweise enttäuscht. In den folgenden Fälle dürfen Sie ihr Angebot jedoch vorzeitig beenden:

Grund: Der Artikel ist ohne Ihr Verschulden verloren gegangen, beschädigt oder

Abb. 15.8 Links Klassifikation der Sätze, rechts Hervorhebung des ausgewählten Satzes, alle Rechte vorbehalten

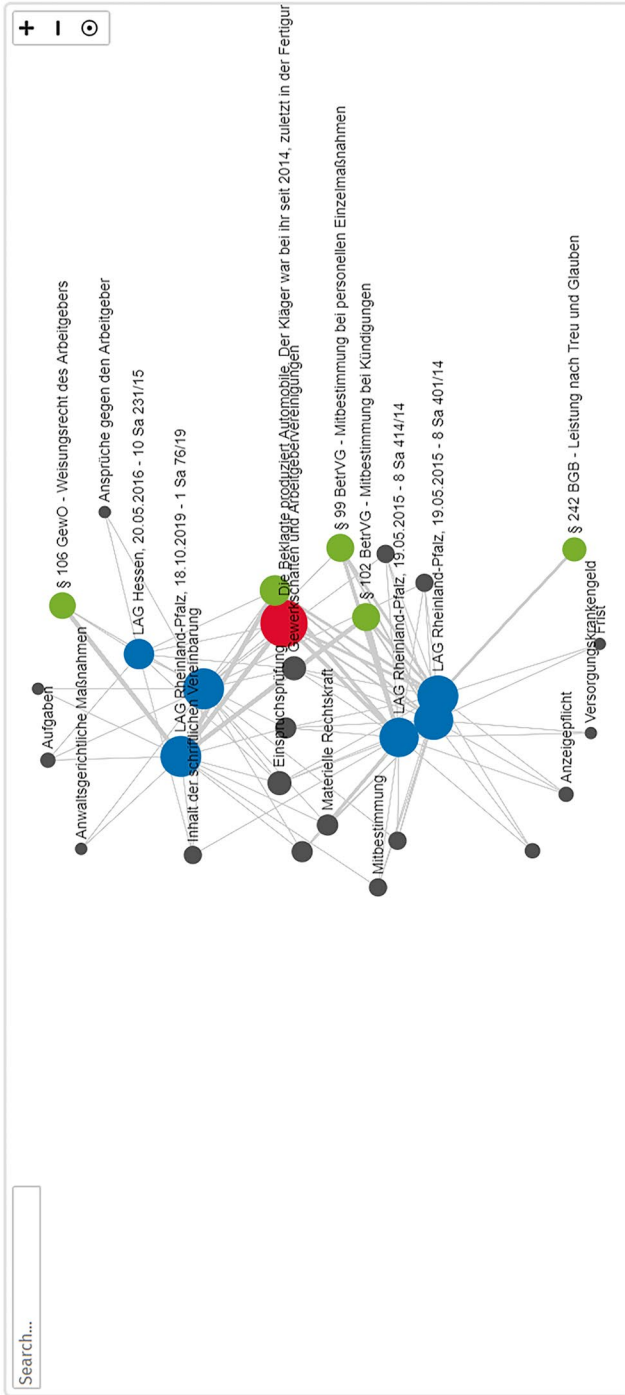


Abb. 15.9 Grafische Darstellung des Ergebnisses einer Recherche

Bringt man diese Informationen in eine Reihenfolge, so können Jurist:innen Antworten auf folgende Fragen finden:

- Was ist das tragende Argument für diesen Fall?
- Ist dieses Argument für meinen Fall anwendbar?
- Was sind die passenden Fakten aus diesem Fall – verglichen mit meinem Fall?
- Tragen die Fakten das Argument?
- Verfängt das Argument und kann ich damit überzeugen?
- Was sind mögliche Gegenargumente, mit denen ich umgehen können muss?
- Sind diese Gegenargumente ggf. sogar stärker als mein Argument, sodass ich auf eine andere Strategie wechseln sollte?

All diese Fragen lassen sich dann beantworten, wenn die Argumente entsprechend extrahiert und klassifiziert werden. Ist dies geschehen, kann in die Analyse der Argumente eingestiegen werden und diese können für die Strategie sinnvoll verwendet werden.

Es handelt sich also erneut um eine Kombination aus verschiedenen Elementen:

- maschinelles Lernen zur Erkennung der Argumente,
- maschinelles Lernen zur Klassifizierung, ob es sich um ein positives oder negatives Argument handelt,
- Zugriff auf die im Wissensgraphen abgespeicherten Informationen, um vergleichbare Entscheidungen mit ähnlichen Argumenten und ggf. unterschiedlichem Ausgang zu finden etc.

• **Ranking und Re-Ranking**

Bisher kommen im juristischen Bereich in im Einsatz befindlichen Recherche-systemen zumeist Elasticsearch und Solr mit zusätzlichen Elementen wie Synonym-listen und speziellen Boosting-Faktoren für das Ranking zum Einsatz. Mehr und mehr werden es kombinierte Lösungen. Gerade die Möglichkeit, Sätze statt Keywords einzugeben, erfordert neben der grundsätzlichen Möglichkeit, überhaupt mit Sätzen umgehen zu können, erweitertes Wissen. Dazu gehört bspw. das Erkennen eines Rechtsgebietes und/oder bestimmter Entitäten in der Nutzereingabe, wie bspw. juristische Konzepte aus der Ontologie.

Dies stellt jedoch lediglich den ersten Schritt dar. Klassisches Boosting, bspw. der aktuellen Entscheidungen oder der Entscheidungen der obergerichtlichen Rechtsprechung, bleibt weiterhin relevant. Ergänzt wird es zudem durch Methoden des maschinellen Lernens, die gemeinsam mit der klassischen Suche eingesetzt werden. So wird in der besprochenen Lösung Legal Analytics eine auf Wort- und Dokument-Embeddings basierende Lösung verwendet, um die gefundenen Entscheidungen erneut zu ranken. Diese Kombination aus klassischer Suche auf der einen Seite und aus bspw. vektorbasiertem Re-Ranking hat in unserer Erfahrung sehr gute Ergebnisse erzeugt. Betrachtet man das suboptimale Abschneiden vortrainierter Large Language Models im klassischen Information Retrieval Task, so scheint gerade diese Kombination aus klassischer Suche und maschinellem Lernen zukunftsfruchtig (vgl. [Dezhao 2022](#)).

15.5.3 Anwendungsfälle bei der Erstellung juristischer Dokumente

Auch bereits beim Erstellen juristischer Dokumente kommen maschinelles Lernen und insbesondere auch hybride Ansätze zum Einsatz.

- **Klassifikationsaufgaben:**

Im Bereich der Erstellung der juristischen Dokumente kommen Klassifikationsaufgaben zur Anwendung. Ein Beispiel sind das automatische Erkennen eines Rechtsgebietes des Dokuments und das Zuordnen des entsprechenden Taxonomieeintrags in das Rechtsprechungs- oder Sekundärdokument. Rechtsgebiet kann hier bspw. Strafrecht, Arbeitsrecht, Zivilrecht sein. Es können aber auch tiefere Konzepte aus der Taxonomie zugeordnet werden. Bei einer solchen Erkennung wird auf zusätzliche Informationen aus Metadaten oder dem Wissensgraphen über die Dokumente zugegriffen:

- Aktenzeichen enthält das Registerzeichen, die Informationen über das Aktenregister und die Kammer bzw. den Senat (bspw. steht StR in 4 StR 352/88 für ein Revisionsverfahren im Strafrecht, 4 ist der 4. Senat des BGH) und das Jahr (352/88 sagt aus, dass es sich um den Eingang bei Gericht mit der fortlaufenden Nummer 352 aus dem Jahr 1988 handelt),
- Normen und Entscheidungen, die innerhalb des Dokuments zitiert werden,
- Entscheidungen und Sekundärliteratur, die auf das Dokument verweisen, ggf. auch mit der Berücksichtigung, ob es ein zustimmender oder ablehnender Verweis ist.

- **Extraktionsaufgaben:**

Um die Nutzbarkeit der Recherche verbessern zu können, muss bereits beim Erstellen der Dokumente auf die Verlinkung hingewirkt werden. So können regelbasierte Citation Recognition Services zur Anwendung kommen oder alternativ durch maschinelles Lernen die Zitate erkannt werden. Nach der Erkennung können diese dann wiederum in den Wissensgraphen integriert werden, um so sicherzustellen, dass beim Zugriff zum Zeitpunkt der Recherche alle Informationen zur Verfügung stehen.

Auch die automatische Extraktion bzw. Erzeugung von Schlagworten spielt hier eine Rolle. Häufig geben Jurist:innen derzeit immer noch Schlagworte in juristische Recherchedatenbanken ein, sodass hiermit ein Boosting der besonders relevanten Dokumente einfach ermöglicht werden kann. Da es sich bei den Schlagworten in der Regel um juristische Konzepte handelt, können diese wiederum mit den vorhandenen Informationen in den Taxonomien abgeglichen werden; gerade bei der Erzeugung im Gegensatz zur Extraktion von Schlagworten hilft dies, eine gleichbleibende Qualität zu erreichen. Letztlich wird die Information dann erneut im Wissensgraphen abgelegt.

- **Textgenerierung:**

Neben den bereits erörterten Methoden des maschinellen Lernens können Algorithmen auch zur Erzeugung von Text eingesetzt werden. Dies können beispielsweise automatische Zusammenfassungen eines Dokumentes sein, die mithilfe von Sprachmodellen, die für diese Aufgabe entwickelt wurden, erzeugt werden.

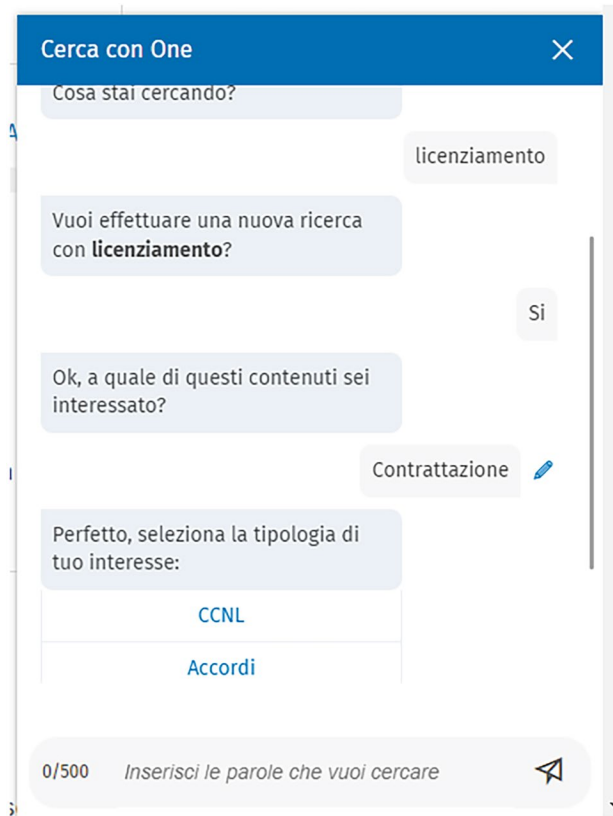


Abb. 15.10 Chatbot-Interaktion zum Auffinden von verschiedenen Dokumenten bei One LA-VORO von Wolters Kluwer Italien, alle Rechte vorbehalten

Es können aber auch Zusammenfassungen von mehreren Dokumenten oder Informationen aus verschiedenen Dokumenten, die zusammengetragen werden, sein. Ein Beispiel hierfür ist die Erzeugung von Definitionssammlungen. Diese können durch das Minen von Definitionen aus allen Rechtsprechungsdokumenten erstellt werden. Gleichzeitig können diese Definitionen dann erneut im Wissensgraphen abgelegt werden, um sie für den Nutzer an der entsprechenden Stelle abrufbar zu machen.

15.5.4 Anwendung bei innovativen Interaktionsmodellen

Neben der Verwendung hybrider Ansätze zum Erstellen von Dokumenten und in der juristischen Recherche kommt ein hybrider Ansatz auch in Chatbots und anderen auf Interaktion beruhenden Komponenten zum Einsatz (Abb. 15.10). Diese Interaktion lässt sich auf der einen Seite als alternative Eingabemethode für die juristische Recherche

verwenden und siedelt diese neben der reinen Keyword-Suche und der Sachverhalts- bzw. Dokumentensuche an. Der Vorteil hiervon ist, dass die Nutzer vereinfacht ausdrücken können, was gesucht wird, ohne bereits die passenden Stichworte und/oder Filterkriterien zu kennen. Gleichzeitig kann dennoch das Ergebnis in einer Trefferliste ausgegeben werden.

Neben der Verwendung von Chatbots kommen auch Voicebots zum Einsatz, sowohl im Bereich des Kundensupports als auch bei der Interaktion mit Recherchemöglichkeiten. Im Gesundheitsbereich von Wolters Kluwer werden sie auch eingesetzt, um bspw. Patienten anzurufen und die Einhaltung der Medikation zu überprüfen.

Zur Realisierung werden hier erneut verschiedene Kombinationen von Techniken verwendet. Entscheidungsbäume können bei der Abfrage bestimmter Informationen verwendet werden, aber auch die Verwendung von Spracherkennung in sowohl geschriebener als auch gesprochener Form ist hierbei notwendig. Schließlich wird auch erneut auf das im Wissensgraphen oder in sonstigen Repräsentationen festgehaltene Wissen zurückgegriffen, um die gesuchte Information abzurufen.

15.6 Technologien und Implementierung

Anhand der Beschreibung der Implementierung einer hybriden Analytics-Lösung in Abb. 15.11 soll gezeigt werden, welche möglichen Techniken zum Einsatz kommen und wie diese orchestriert werden.

Mit 1 dargestellt werden die Elemente, die Informationen speichern. Dies kann eine einfache Liste an Synonymen sein oder aber auch Metadaten im XML-Format auf Grundlage der entsprechenden Schemadatei. Auch bei Wolters Kluwer kommt hier eine Vielzahl verschiedener Techniken zum Einsatz. Ist bspw. das XML-Format der gewählte Standard zur Speicherung der Daten im Content-Management-System, so erfolgt die anschließende Aufbereitung in einer Kombination aus HTML und RDF-Daten. So kann eine einfache Visualisierung in der juristischen Recherchedatenbank erfolgen und die Trennung von Text und Wissen erleichtert die Implementierung semantischer Funktionalitäten.

Unter 2 sind verschiedene, eher etablierte Lösungen zu finden. Seien es Erkennungskomponenten auf Basis regulärer Ausdrücke, wie die Erkennung von Normen, Entscheidungen oder des Sekundärcontents, oder die Query Expansion mit Bordmitteln von Solr. Ein Beispiel hierfür ist die Suchkomponente. Experimente mit Vektorsuchen haben gezeigt, dass sie grundsätzlich in der Lage sind, auch im juristischen Umfeld zu guten Ergebnissen zu kommen, auf weitere Schwierigkeiten mit vektorbasierter Suche wird in Kap. 6 eingegangen. Verwendet man allerdings ein nicht weiter auf die juristische Sprache trainiertes Sprachmodell als Basis für die Vektorisierung, so schneidet eine solche Suche sogar schlechter als ein auf einem relativ simplen TF-IDF bzw. BM25 basierenden Ansatz ab.

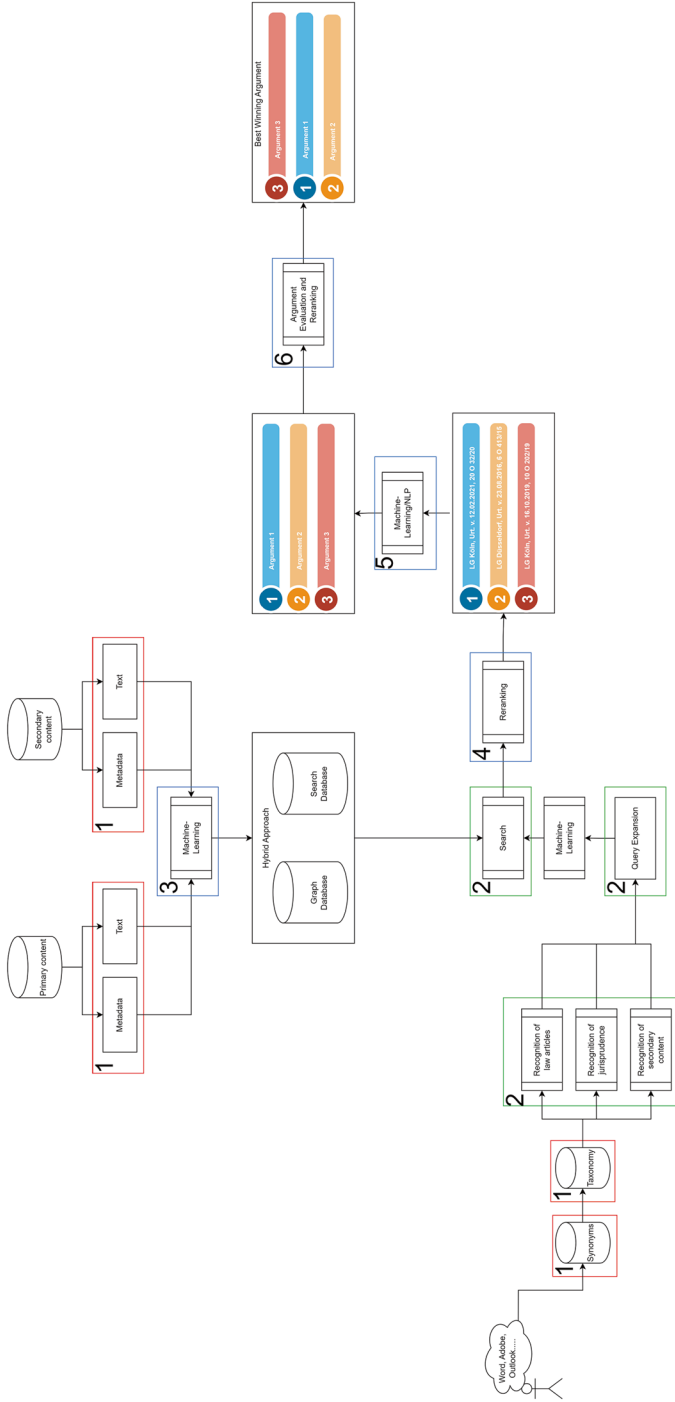


Abb. 15.11 Schema der Analytics-Lösung mit den verschiedenen Arten an Komponenten

Das Erzeugen eigener Word-Embeddings basierend auf dem gesamten zur Verfügung stehenden juristischen Content war eine Entscheidung, die wir bereits vor längerer Zeit getroffen haben, da Out-of-the-box-Sprachmodelle wie BERT nicht überzeugen konnten. Eine weitere Möglichkeit, die ebenfalls genutzt wurde, war das Fine-Tuning der Sprachmodelle für spezifische Aufgaben, wie bspw. das Erzeugen automatischer Zusammenfassungen oder das Erstellen eines Question-Answering-Systems.

Eine Aufgabe der Machine-Learning-Komponente unter 3 ist es, die zur Verfügung stehenden Dokumente aufzubereiten, in dem Daten extrahiert und annotiert werden. Gleichzeitig werden beispielsweise auch Dokumentvektoren aufgebaut, die später weitere Verwendung finden können. Zum Einsatz kommen hier auch Algorithmen wie BiLSTM zur Vorhersage der Klasse eines Satzes im Dokument. Die Ergebnisse dieses Vorgangs werden teilweise direkt in der Suchdatenbank gespeichert (bspw. bestimmte Sätze einer Klasse, die für die natürlichsprachige Suche besonders relevant sind) oder sie werden in die Graphdatenbank überführt.

Die Re-Ranking-Komponente ist, basierend auf den aufgebauten Embeddings, in der Lage, die durch die vorherige Suche bzw. Argumentextraktion ausgewählten Daten neu zu sortieren und in die optimale Reihenfolge zu bringen.

In Komponente 5 in Abb. 15.10 werden unter anderem auch Argumente selektiert. Klassifikationsalgorithmen, die entsprechende Sätze erkennen, werden gemeinsam mit Komponenten zur Verarbeitung natürlicher Sprache (NLP-Komponenten) eingesetzt, um die Argumente zunächst zu finden, sie danach aufzubereiten und letztlich deren Wichtigkeit für die Entscheidung und damit für die von der/dem Nutzer:in eingegebene Query zu evaluieren.

Durch die Kombination aus maschinellem Lernen, NLP-Komponenten und dem Speichern der Informationen in Such- und Graphdatenbank kann das von den Nutzer:innen geforderte hohe Niveau an Genauigkeit bei gleichzeitig hoher Erklärbarkeit sichergestellt werden.

15.7 Empfehlungen

- Im juristischen Umfeld ist der reine Einsatz von maschinellem Lernen oder der reine Einsatz von Wissensrepräsentationen nur bedingt möglich.
- Auch semantische Suche stößt zum Teil an die Grenzen, wenn sie rein vektorbasiert ist und nicht zusätzlich auch weitere Informationsquellen zur Verfügung hat.
- Fine-Tuning von einfachen Modellen ist in der Lage, größere vorgefertigte Modelle zu übertreffen.
- Assistenzsysteme sind vollautomatischen KI-Anwendungen vorzuziehen, da Letztere in kritischen Prozessen nicht akzeptiert werden.

15.8 Zusammenfassung und Ausblick

Hybride KI unterstützt den Semantifizierungsprozess von Unternehmen. Sie ist besser geeignet für die juristische Domäne als maschinelles Lernen oder Wissensgraphen alleine, insbesondere wegen der Notwendigkeit, Transparenz und Erklärbarkeit ins Zentrum fast aller Anwendungen zu stellen.

Aus diesem Grund stehen bei Wolters Kluwer auch eher Assistenzsysteme als Anwendungsbereiche im Vordergrund, da eine komplett automatische KI-Anwendung (selbst wenn sie funktioniert) von den Kund:innen in kritischen Prozessen nicht akzeptiert werden würde.

KI ist Teil des alltäglichen Arbeitens geworden, besonders in der Neuproduktentwicklung und bei der Softwareentwicklung. Die Aufwände rechnen sich allerdings nur, wenn die Technologien skalieren, sowohl über Länder und Sprachen als auch über Domänen hinweg.

Der bisherige Schwerpunkt unserer Aktivitäten lag im Bereich Wissensgenerierung und Semantifizierung. In Zukunft wollen wir zusätzlich die Themen Interaktionsformen und Dokumentenerstellung in den Fokus nehmen. Ebenso wollen wir die rechtlichen und ethischen Aspekte der KI-Nutzung systematischer durchleuchten und somit unsere Kernkompetenz als juristischer Informationsdienstleister dazu nutzen, die KI als gesellschaftliche Herausforderung voranzubringen.

Literatur

- Dezhao et. al., On the Effectiveness of Pre-Trained Language Models for Legal Natural Language Processing: An Empirical Study, 2022. <https://doi.org/10.1109/ACCESS.2022.3190408>, S. 75848 f.
- Dirschl, C. et al., "Semantify" Business and Content to Meet Demands for Expert Solutions in Professional Markets, in: The Semantic Web: ESWC 2022 Satellite Events, 2022, Pages 295–298
- Hartz, Christian, Hinter den Kulissen juristischer Suchmaschinen, CTRL 2/2022, S. 48 ff.
- Nestsiaerenia Ihar/Hartz, Christian, Data Science meets BGB: Eine Einführung in die juristische Datenvisualisierung, CTRL 1/2023, S. 93 ff.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Kosteneffiziente Rekognition durch Out-of-the-Box KI und Semantik

16

Achim Reiz  und Kurt Sandkuhl 

Kernaussagen

1. Künstliche Intelligenz hat das Potenzial, bestehende Geschäftsmodelle grundlegend zu verändern und Wettbewerbsvorteile zu heben.
2. Das Trainieren von neuronalen Netzen ist äußerst kostenintensiv. Treiber sind das Kuratieren der Datensätze und die benötigte Rechenleistung.
3. Spezialisierte Machine-Learning-Modelle lassen sich nur schwer auf neue Anforderungen anpassen. Jedwede Änderung bedingt ein erneutes Trainieren des Modells.
4. Es gibt eine Vielzahl von frei verfügbaren Trainingsdaten, vortrainierten Modellen und Software-as-a-Service-KI-Lösungen. Diese ermöglichen einen einfachen Einstieg in die Nutzung von KI, passen aber häufig nicht zu dem gegebenen Anwendungsfall im Unternehmen.
5. Die Kombination von Semantik mit Out-of-the-Box-KI und eigenen trainierten Modellen erlaubt das Kontextualisieren der Ergebnisse auf die individuelle Domäne.
6. Methoden aus der IT-Wirtschaftlichkeitsbetrachtung lassen sich auch für KI-Lösungen einsetzen
7. Die Nutzung dieses hybriden Ansatzes birgt substanzielle Einsparpotenziale.

A. Reiz (✉) · K. Sandkuhl
Universität Rostock, Rostock, Deutschland
E-Mail: achim.reiz@uni-rostock.de

K. Sandkuhl
E-Mail: kurt.sandkuhl@uni-rostock.de

16.1 Einleitung

Die Anwendung von künstlicher Intelligenz (KI) hat das Potenzial, verschiedenste Bereiche unserer alltäglichen Welt nachhaltig zu verändern. Die Erwartungen an KI sind dabei groß, wenn man Marktstudien, Prognosen von Experten oder Umfragen unter Unternehmen glaubt. Allein im produzierenden Gewerbe in Deutschland ist nach einer Studie¹ vom VDI/VDE in einem 5-Jahres-Zeitraum eine Steigerung der Wertschöpfung um über 30 Mrd. EUR zu erwarten. Eine McKinsey-Studie erwartet sogar, dass das Potenzial von KI höher ist, als es seinerzeit bei der Einführung der Dampfmaschine war.² Allgemeingültige Prognosen sind schwer zu erstellen, da das Potenzial je nach Branche und Unternehmensgröße stark variieren kann. Es herrscht aber Einigkeit unter Experten, dass KI auch hier zur Steigerung der Wettbewerbsfähigkeit, Schaffung neuer Produkte und Dienstleistungen, Verbesserung der Arbeitsqualität oder Realisierung digitaler Geschäftsmodelle dienen kann.³ Digitalisierte Geschäftsmodelle und Prozesse skalieren deutlich flexibler und besser, können Kosten einsparen und Kunden besser zufriedenzustellen. Das rechtzeitige Innovieren schafft hierbei oft den entscheidenden Wettbewerbsvorteil vor der Konkurrenz (Iansiti und Lakhani 2020, 8 ff.).

Die Nutzung von KI ist jedoch auch assoziiert mit Kosten bei deren Einführung und Betrieb. Ein Blick auf die Voraussetzungen, die in Unternehmen geschaffen werden müssen, um substantielle Vorteile aus der KI zu ziehen, zeigt, dass diese vielfältig und nicht zu unterschätzen sind. Wie auch bei anderen technologischen Innovationen sind dabei die Unternehmensstrategie, Fachwissen, die erforderlichen Ressourcen und die Bereitschaft zur Veränderung des Unternehmens wichtige Aspekte. Bei der Einführung von KI hat sich gezeigt, dass auch die richtigen Prozesse und vor allem Daten wichtig sind, die in diesem Beitrag im Fokus stehen sollen.

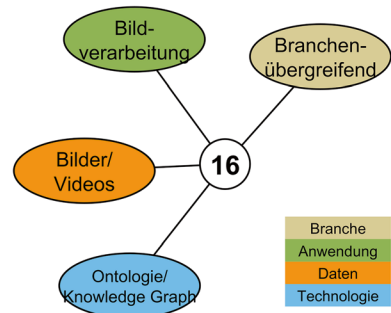
Daten haben bei allen KI-Verfahren eine hohe Bedeutung, sei es als Grundlage für das Trainieren von Modellen zum maschinellen Lernen (subsymbolische KI oder statistische Verfahren) oder als Wissensbestand und Faktensammlung für Expertensysteme oder semantische Verfahren (symbolische KI). Das Sammeln und Annotieren von Trainingsdaten für Verfahren des maschinellen Lernens erfordern oft einen hohen Personaleinsatz. Zwar gibt es Ansätze für das computerunterstützte Annotieren, im Kern jedoch müssen die zugrunde liegenden Trainingsdaten gewichtet und akkurat sein, was in der Regel Mitarbeiter sicherstellen müssen. Stehen die Daten bereit, ist der nächste Schritt das Trainieren des zu nutzenden Maschinelles-Lernen-(ML-)Modells. Hierfür ist für komple-

¹ Potenzial der Künstlichen Intelligenz im produzierenden Gewerbe in Deutschland. Herausgeber: Begleitforschung PAiCE, VDI/VDE Innovation+Technik GmbH.

² McKinsey Global Institute (2018): „Notes from the frontier: Modeling the impact of AI on the World Economy“, Discussion Paper, September 2018.

³ Künstliche Intelligenz im Mittelstand. Herausgeber: Begleitforschung Mittelstand-Digital. WIK GmbH.

Abb. 16.1 Thematische Einordnung des Kap. 16



xere Inputdaten große Rechenleistung erforderlich, oft verteilt über mehrere Maschinen. Jedes Verändern des ML-Modells bedingt ein erneutes Trainieren, d. h., schnelllebige Anforderungen verteuern die Nutzung eines solchen Systems. Weiter bindet auch die betriebliche Nutzung der KI unter Verwendung eines bereits trainierten ML-Modells signifikante Rechenkapazität.

Zwar gibt es eine Vielzahl frei verfügbarer Datensätze, vortrainierter ML-Modelle oder Out-of-the-Box-Erkennungslösungen, welche das Annotieren, Trainieren oder sogar das Bereithalten von Rechenkapazität für die Erkennung überflüssig machen. Deren Ergebnisse sind jedoch entweder sehr allgemein gehalten oder bereits auf eine bestimmte Domäne zugeschnitten und passen häufig nicht auf den gegebenen Anwendungsfall.

Die Verbindung von regelbasierten Wissenssystemen mit diesen einfach zu nutzen, vordefinierten Trainingsdaten, ML-Modellen oder Services kann diese auch für weitere Anwendungsbereiche erschließen. Die Ergebnisse aus der Rekognition, d. h. der Anwendung maschinellen Lernens zur Erkennung von Strukturen oder Mustern, lassen sich in einer Ontologie verknüpfen und neu interpretieren und somit zuschneiden auf die gegebene Domäne. Minimale, kostengünstige, selbst erstellte ML-Modelle lassen sich durch die Verknüpfung mit bestehenden Lösungen erweitern. Auch ein Anpassen der Verknüpfungen ist in der Ontologie schnell und kostengünstig möglich.

In diesem Kapitel schlagen wir eine Architektur für eine kontextualisierte Rekognitionslösung vor. Wir zeigen anhand einer Beispielapplikation, wie sich Out-of-the-Box-Methoden mit einer ontologiebasierten Übersetzung kombinieren lassen und welche ökonomischen Nutzeffekte sich erschließen lassen (Abb. 16.1).

16.2 Eine Architektur für eine kostengünstige KI

Vorgefertigte Erkennungslösungen auf der Basis maschinellen Lernens lassen sich mit vergleichsweise wenig Ressourcenaufwand aufsetzen und nutzen. Um die Ergebnisse in die individuelle Domäne zu übersetzen, sollen diese von einem wissensbasierten System interpretiert werden. Die Boxologie in Abb. 16.2 zeigt die Verbindung beider Systeme

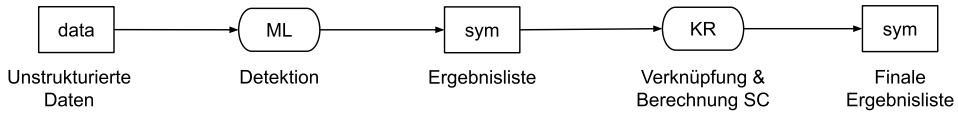


Abb. 16.2 Boxology-Modell des hybriden KI-Systems

und orientiert sich an den Designmustern für hybride Lernverfahren von (van Harmelen und Teije 2019).⁴

Unstrukturierte Daten wie Text, Bilder oder Tonaufnahmen werden erst über ein maschinelles Lernverfahren, vorzugsweise unter Nutzung eines bereits trainierten Modells oder unter Nutzung eines oder mehrerer Erkennungsservices, in die endliche Menge der entsprechenden Detektoren übersetzt. Die Detektoren sind in einer Ontologie mit einem semantischen Zwilling repräsentiert: Jeder relevante Detektor kann entweder direkt in eine Zielrepräsentation übersetzt werden oder mehrere Detektoren werden miteinander in Beziehung gesetzt, um verschiedene ML-Modelle zu verknüpfen. Auch kann bei Bedarf auf eine Inferenzmaschine zurückgegriffen werden.

Erkennungsalgorithmen des maschinellen Lernens basieren auf statistischen Verfahren. Ihre Ergebnisse sind stets von Unsicherheit geprägt und die Ergebnisse werden mit einer Erkennungswahrscheinlichkeit bewertet. So ist es möglich, dass in einem Bild eine Rose zu 93 % erkannt wird, eine Pflanze zu 12,6 %. Demgegenüber steht die Wissensmodellierung der semantischen Verfahren. Hier wird das modellierte Wissen in der Regel als Grundwahrheit angenommen. Diese unterschiedlichen Sichtweisen müssen bei der Reinterpretation der ML-Ergebnisse durch die Semantik berücksichtigt werden.

Reiz et al. (2022) schlagen für die Verknüpfung dieser zwei Verfahren einen Messwert *Semantic Confidence* (*SC*) für die Verknüpfung von Objektdetektion mit Bounding-Boxen und Semantik vor. Initial wird die Größe eines detektierten Objekts $\vec{o}_{i_{po}}$ als Prozentwert der Fläche berechnet, die von den Begrenzungsrahmen des Objekts abgedeckt wird. Anschließend werden Größe und Wahrscheinlichkeitswert des Detektors (der Detektionswert des neuronalen Netzes) $\vec{o}_{i_{pb}}$ addiert und dann mit dem in der Semantik abgeleiteten Element multipliziert. Im letzten Schritt werden die Werte der Kontextelemente aggregiert und auf 100 % für das meistdetektierte Objekt normalisiert.

$$SC = \left(\vec{o}_{i_{po}} + \vec{o}_{i_{pb}} \right) * S = \vec{o}_i * S = \begin{pmatrix} o_{i_1} \\ \vdots \\ o_i \end{pmatrix} \begin{pmatrix} s_{11} & \cdots & s_{1k} \\ \vdots & \ddots & \vdots \\ s_{i1} & \cdots & s_{ik} \end{pmatrix}$$

⁴Für eine Einführung in die Boxology siehe Kap. 1.

Das Ergebnis ist eine Bewertung der Ergebnisse anhand ihrer Größe und Erkennungswahrscheinlichkeit. Die Ausgaben aus der semantischen Verknüpfung lassen sich somit auf ähnliche Weise nutzen wie eine ausschließliche ML-Lösung. Fallstudie 2 zeigt die Berechnung von *Semantic Confidence* (SC) anhand eines Beispiels.

16.3 Potenziale für Kosteneinsparungen durch die Nutzung hybrider KI

Durch die Nutzung von hybrider KI lassen sich Trainingskosten für das maschinelle Lernen einsparen. Im besten Fall kann auf das Training vollends verzichtet werden. Es gibt bereits eine Vielzahl frei verfügbarer Modelle für verschiedenste Anwendungsbereiche. Deren Ergebnisse lassen sich über eine Semantik miteinander verknüpfen und somit auf die eigene Domäne übersetzen. Teilweise ist das ausschließliche Zurückgreifen auf vortrainierte Modelle nicht möglich. In diesem Fall lässt sich jedoch der Trainingsaufwand für das neuronale Netz durch das Verknüpfen von individuellen und vortrainierten Modellen vermindern. Die Semantik kann auf diese Weise z. B. Subkategorien von detektierten Objekten über die trainierten Modelle hinaus erfassen.

16.3.1 Nutzung von Out-of-the-Box-KI

Zu Beginn der 2000er-Jahre war die Bilderkennung noch primär Gegenstand von Forschung und Wissenschaft. Hier ergab sich initial die Motivation, Vergleichbarkeit zwischen neuen Rekognitionsmodellen zu schaffen. Dieses führte zu großen, annotierten Datensätzen, welche zumeist frei verfügbar zum Trainieren eigener Modelle genutzt werden können (Szeliski 2022, 9 ff.).

Mit steigender Präzision der Algorithmen und größerer und günstigerer Rechenkapazität fanden die Algorithmen des maschinellen Lernens vermehrt Einzug in praktische Anwendungsbereiche. Frameworks für das Aufsetzen von maschinellem Lernen wie Scikit-learn, Tensorflow und Pytorch setzten sich durch, die Nutzerfreundlichkeit stieg durch Schnittstellen wie Keras.⁵ Große Hürden für deren Implementierung verblieben jedoch: Die benötigte Rechenkapazität für das Training solcher Modelle ist immens, die Parametrierung und Selektion der Algorithmen oft komplex und jede Änderung des Modells erfordert die Wiederholung des Trainingsprozesses. Dies führte zu dem vermehrten Teilen von vortrainierten Modellen. Diese können direkt von den entsprechenden Frameworks verwendet werden, sind insbesondere verglichen zu den Trainingsdaten außerordentlich schlank und erlauben mit wenigen Zeilen Code die Implementierung einer entsprechenden ML-Applikation. Es entwickelten sich Marktplätze wie Tensorflow Hub

⁵<https://scikit-learn.org/>, <https://www.tensorflow.org/>, <https://pytorch.org/>, <https://keras.io/>

oder Pytorch Hub. Diese erlauben es, gezielt nach Modellen je nach Anwendungszweck und Domäne zu suchen.

Die letzte Stufe der vereinfachten Nutzung ist das Zurückgreifen auf Software-as-a-Service-(SaaS-)Lösungen. Sämtliche großen Cloud-Provider, wie Microsoft Azure, Amazon Web Services oder IBM Cloud, bieten ML-basierte KI-Lösungen an. Diese können über RESTful-Services oder bereitgestellte APIs in bestehende Applikationen eingesetzt werden.

16.3.2 Frei verfügbare Datensätze für das eigenverantwortliche ML-Training

Die verschiedenen, online verfügbaren Datensätze enthalten in der Regel alle nötigen Informationen für das Training eines eigenen ML-Modells. Dies umfasst die Rohdaten, jedoch insbesondere auch die zu erkennenden Annotationen.

Trainingsdaten mit Annotationen sind verfügbar in verschiedenen Größen und Detailgraden für die verschiedensten Rekognitionsarten, z. B. im Bereich Computer Vision für das Training von Klassifikations-, Detektions- oder Segmentationsaufgaben oder die Objektverfolgung in Bewegtbildern. Ebenfalls vertreten ist die Text- oder Audioerkennung. Es werden jedoch auch speziellere Daten zur Verfügung gestellt, z. B. annotierte Lidaraufnahmen, 3D-Meshes oder Graphen.

Auch die Domänen sind unterschiedlich. Teils entspringen die Daten aus spezieller Anwendungsforschung, siehe z. B. das annotierte Brustkrebsdatensatz auf der Basis von Magnetresonanztomografie (Buda et al. 2021), multidimensionale Lidar- und Videoaufnahmen für das autonome Fahren (Geiger et al. 2013) oder eine Punktverfolgung von Trainingseinheiten (Weitz und Hensley 2022). Es gibt jedoch auch eine Vielzahl von Datensätzen für domänenunabhängige Erkennungsaufgaben, z. B. Coco für die Erkennung von 91 verschiedenen Objekten wie *Katze*, *Bus*, *Buch* (Lin et al. 2014) oder Librispeech, ein Korpus von 8000 englischen, gesprochenen Hörbüchern für das Training von Natural-Language-Modellen (Panayotov et al. 2015).

Das Forschungsteam von Meta betreibt die Webseite *paperswithcode.com*, eine große Sammlung von verschiedenen Datensätzen. Diese sind verlinkt zu den entsprechenden Veröffentlichungen, Benchmarks und eingesetzten Methoden und kategorisiert nach Art der zu erkennenden Daten, der Rekognitionsaufgabe und Sprache. Einen ähnlichen Ansatz verfolgt *huggingface.co* und *kaggle.com* mit einem gitbasierten Ansatz für ML-Modelle und Trainingsdaten. Für die Suche nach vordefinierten Trainingsdaten bieten diese Webseiten einen guten Einstiegspunkt.

16.3.3 Framework für das maschinelle Lernen und deren Modell-Repositories

ML-Algorithmen bauen auf komplexen statischen Verfahren auf. Diese müssen heute jedoch nicht mehr händisch implementiert werden. Es gibt mittlerweile eine Vielzahl von Frameworks, welche komfortabel nutzbare Interfaces für die zugrunde liegenden Methoden zur Verfügung stellen. Diese bieten teils feingranularen Zugriff auf die verschiedenen Konfigurationsparameter (z. B. Tensorflow) oder verstecken die Komplexität zugunsten einer einfacheren Nutzung (z. B. Keras).

Auch unter Nutzung populärer ML-Frameworks sind die Konfiguration und das effektive Training von Modellen aufwendig. Zum einen sind das Aufsetzen und die Konfiguration des Trainingsprozesses nicht trivial, zum anderen benötigt der Trainingsprozess für größere Datensets zumeist signifikante Rechenleistung.

Viele der Frameworks stellen daher Verzeichnisse für das Teilen von vortrainierten Modellen zur Verfügung. Diese bauen oft, aber nicht zwangsweise auf frei verfügbaren Datensets, wie z. B. COCO, auf. Teils sind für einen Datensatz verschiedene Modelle verfügbar. So gibt es in dem Tensorflow Repository 39 Modelle für COCO mit verschiedenen Anwendungszwecken und Ausgaben: Zum Beispiel mobiloptimiert vs. hohe Präzision oder Keypoints vs. Boxen vs. Masken.

Tab. 16.1 zeigt eine Auswahl der populärsten Frameworks und deren Verzeichnisse. Während die Repositorien der Frameworks ausschließlich Modelle spezifisch für die gegebene Software bereitstellen, gibt es mit dem Open-Neural-Network-Exchange (ONNX-)Format eine Initiative zum plattformübergreifenden Teilen von Modellen. OpenVino verfolgt einen ähnlichen Ansatz und stellt eine abstrakte API für das Nutzen von verschiedenen ML-Frameworks zur Verfügung.

Tab. 16.1 Viele ML-Frameworks bieten vorkonfigurierte Modelle an

Framework	Repository
Tensorflow	tfhub.dev/
Pytorch	https://pytorch.org/hub/
Wolfram	https://resources.wolframcloud.com/NeuralNetRepository/
Apache MXNet	–
Scikit-learn	–
FluxML	https://github.com/FluxML/model-zoo/
<i>Diverse</i>	https://github.com/onnx/models
<i>OpenVino</i>	<i>Open Vino Model Zoo</i>

16.3.4 Erkennung als Software as a Service

Vortrainierte Modelle können die Implementation von Rekognitionslösungen deutlich vergünstigen und beschleunigen. Jedoch bindet auch die Nutzung eines komplexen neuronalen Netzes Ressourcen in erheblichem Maße. Dies kann, z. B. bei Anwendungen mit wechselnder Last, die Skalierung erschweren. Noch einfacher ist daher die Nutzung von bereitgestellten KI-Lösungen. Die großen Cloud-Anbieter bieten hierfür vorgefertigte Lösungen an. So gibt es neben Software-Lösungen mit nutzerfreundlicher Trainingsoberfläche auch bereits vortrainierte Erkennungslösungen. Diese können meist entweder unter Nutzung einer API für die gängigen Programmiersprachen eingebunden werden oder als RESTful-Webschnittstelle. Insbesondere von Vorteil ist für den Nutzer die flexible Skalierung, da die Nutzung in der Regel ausschließlich nach der Anzahl der Anfragen berechnet wird und keine eigene Rechenkapazität für das Ausführen der Erkennungsaufgaben benötigt wird. Die vortrainierten Modelle der Cloud-Anbieter sind zumeist zugeschnitten auf die gängigsten Anwendungsfälle der Industrie, z. B. das Klassifizieren von pornografischen Inhalten, das Taggen/Klassifizieren von Bildern oder das Detektieren von alltäglichen Objekten.

16.4 Fallstudie 1: Verknüpfen von individueller und Out-of-the-Box-KI

Zwar gibt es eine große Bandbreite aus frei verfügbaren Trainingsdaten und ML-Modellen. Doch lassen sich immer noch viele Anwendungsszenarien nicht ausschließlich auf Out-of-the-Box-KI aufbauen. Dennoch lässt sich teilweise auch für diese Fälle durch die Verknüpfung verschiedener ML-Modelle Trainingsaufwand einsparen. So lässt sich ein vergleichsweise kleines eigenes ML-Modell mit vortrainierten Modellen kombinieren, um Elemente über das eigene Modell hinaus zu schließen.

Reiz et al. (2021) zeigen diese Art von Verknüpfung anhand eines Anwendungsfalls in der Modedomäne (Abb. 16.3). In der entsprechenden Applikation sollen Kleidungsstücke in Videos erkannt werden, um zielgerichtete Werbung schalten zu können. Die verschiedenen Arten von Kleidung lassen sich jedoch oft nur schwer unterscheiden. Das für diesen Zweck trainierte neuronale Netz nutzt daher nur eine begrenzte Anzahl von vergleichsweise groben Detektoren. Weiter kommt eine Out-of-the-Box-Szenenklassifikation (Zhou et al. 2018) zum Einsatz. Die Semantikkomponente verknüpft die beiden Sichten unter Nutzung eines Zwischenobjekts *Occasion* für die Verringerung der Komplexität. Diese erlaubt nun das Schließen zusätzlicher passender Kleidungsstücke, die deutlich über das eigene trainierte Detektionsmodell hinausgehen.

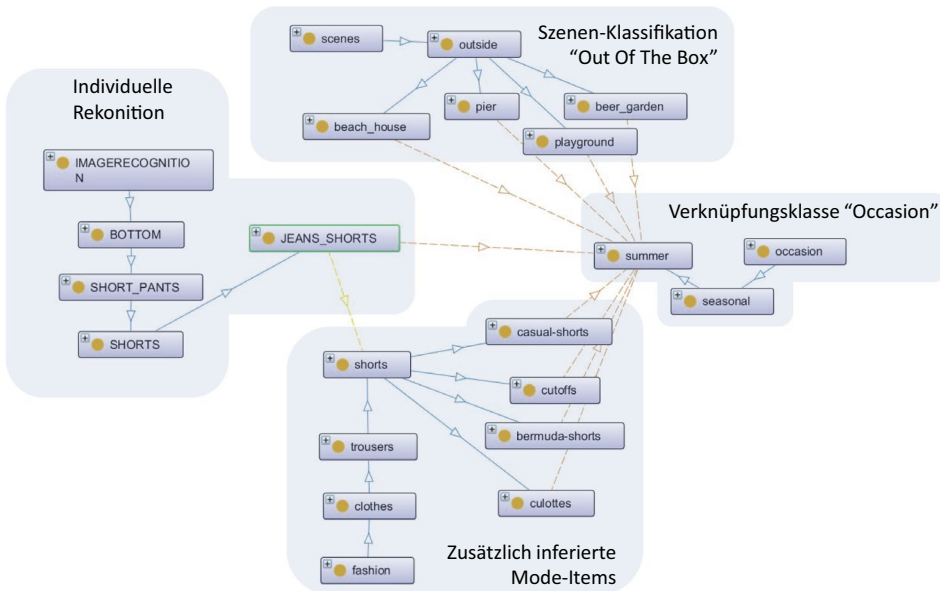


Abb. 16.3 Die Verknüpfung der eigenhändig trainierten Detektion mit Szenen aus bestehenden Datensätzen unter Nutzung einer Verknüpfungsklasse für die Inferierung zusätzlicher Mode-Items (Reiz et al. 2021)

16.5 Fallstudie 2: SemOI – Eine semantische Erweiterung der Open-Images-Detektion

Die zweite Fallstudie stellt die Beispielapplikation SemOI (**semantische Augmentation für Open Images**) vor (Reiz et al. 2022). Die Software baut auf Tensorflow und vortrainierten Modellen von Tensorflow Hub als Rekognitions-Engine auf sowie dem Web-Framework Django für das User Interface. SemOI ist frei lizenziert und verfügbar in Github.⁶ SemOI individualisiert die Ergebnisse einer Out-of-the-Box-ML-Lösung, ohne dass in den ML-Prozess eingegriffen werden muss (Abb. 16.4).

Für die Detektion stehen dem Anwender zwei verschiedene neuronale Netze zur Verfügung: ResNet und MobilNet.⁷ Ersteres zielt darauf ab, eine hohe Genauigkeit zu erzeugen; Letzteres zeigt mögliche Integrationsszenarien für Geräte mit begrenzter Rechenleistung (Wu et al. 2020). Es ist jedoch möglich, die zugrunde liegende Er-

⁶<https://github.com/Uni-Rostock-Win/SemOI> und <https://doi.org/10.5281/zenodo.5005618>

⁷https://tfhub.dev/google/faster_rcnn/openimages_v4/inception_resnet_v2/1, https://tfhub.dev/google/openimages_v4/ssd/mobilenet_v2/1

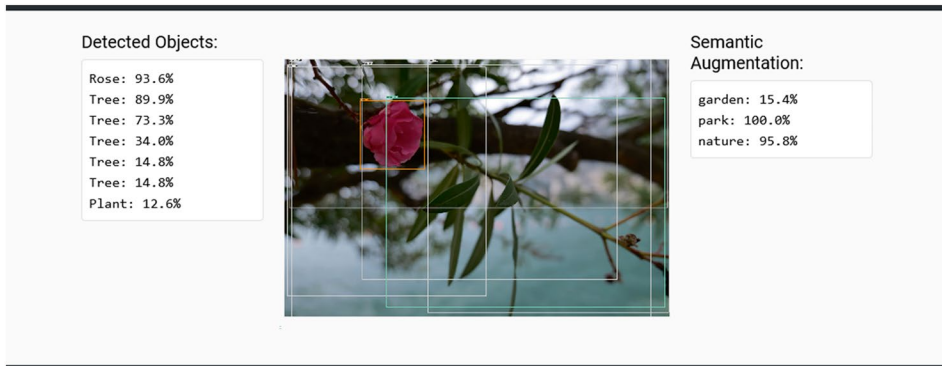


Abb. 16.4 Analysiertes Bild mit Ergebnissen. Links sind die Ergebnisse aus der Bilderkennung, rechts die Ergebnisse aus der semantischen Augmentation (Reiz et al. 2022)

kennungs-Engine mit anderen, standardisierten, frei verfügbaren Modellen und Trainingsdaten auszutauschen.⁸

Abb. 16.6 gibt einen Überblick über die Architektur des Prototyps. Zunächst erkennt das Out-of-the-Box-ML-Modell die gegebenen Objekte. Nachdem eine Erkennung abgeschlossen ist, ruft die Anwendung die semantische Augmentationseinheit mit einer Liste der erkannten Elemente auf. Jeder Objektdetektor hat einen ontologischen Zwilling in der Semantik und kann über das Detektorlabel oder die Open-Images-ID identifiziert werden. Die Detektoren sind mit neuen, kontextualisierten Items verbunden (Abb. 16.6). In der semantischen Augmentationseinheit wird die Anfrage in eine SPARQL-Abfrage übersetzt, die die Ergebnisse aus der Ontologie abrufen und die Kontextelemente zur weiteren Analyse zurückgibt. Anschließend berechnet die Applikation den *Semantic-Confidence*-Wert (Abb. 16.5).

Tab. 16.2 zeigt beispielhaft die Ergebnisse für das analysierte Bild in Abb. 16.4 unter Nutzung der Verknüpfungen aus der Ontologie in Abb. 16.6: Die Detektion erkannte eine Rose, Pflanze und verschiedene Instanzen von Bäumen mit der Konfidenz \vec{o}_{pb} und der Größe \vec{o}_{po} . Initial werden \vec{o}_{po} und \vec{o}_{pb} für jedes detektierte Objekt aggregiert zu \vec{o}_i , dann multipliziert mit den verbundenen Klassen S (Step 1). Das Ergebnis ist eine Liste gewichteter Kontextelemente. Im nächsten Schritt werden die Werte der Kontextelemente aggregiert (Schritt 2) und auf einen Prozentwert normalisiert, der den höchsten SC-Wert als 100 % berücksichtigt.

$$SC = 6,108 \text{ nature} + 6,374 \text{ park} + 0,9801 \text{ garden} \quad \text{Schritt 2}$$

$$SC_{\text{normalized}} = 100 \% \text{ park} + 95,8 \text{ nature} + 15,8 \% \text{ garden} \quad \text{Schritt 2}$$

⁸Der Prototyp ist online unter <http://semoi2.informatik.uni-rostock.de> verfügbar.

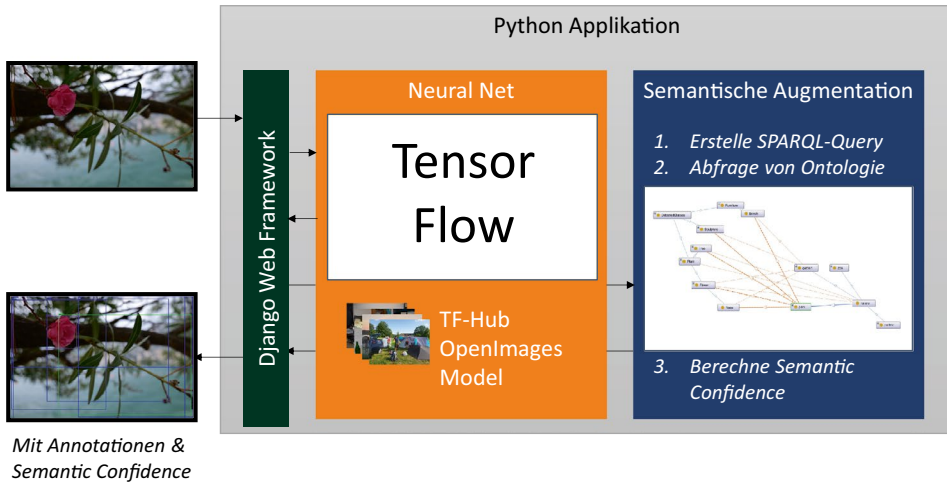


Abb. 16.5 Der Detektions- und Übersetzungsprozess von SemOI (Reiz et al. 2022)

16.6 Ökonomische Betrachtungen

Die vorangegangenen Abschnitte haben gezeigt, welche Komplexität aus technischer Sicht in Anwendungen des maschinellen Lernens und auch hybriden KI-Ansätzen steckt. Dieser Abschnitt soll ergänzend zur technischen Sicht die wirtschaftliche Perspektive von hybriden KI-Anwendungen beleuchten. In diesem Abschnitt werden dazu zunächst generelle Ansätze zur Analyse wirtschaftlicher Aspekte vorgestellt, von denen dann ein ausgewählter Ansatz am Beispiel von SemOI verdeutlicht wird.

16.6.1 Generelle Ansätze zur Wirtschaftlichkeitsbetrachtung von IT

Für wirtschaftliche Betrachtungen können generell unterschiedliche Ansätze verwendet werden, wie beispielsweise, was der Wert eines solchen Ansatzes für ein Unternehmen ist, wie sich die Kosten-Nutzen-Relation darstellt oder wie die Wirtschaftlichkeit des Ansatzes ist. Auch wenn die Begriffe Wert, Kosten/Nutzen und Wirtschaftlichkeit eine sehr ähnliche Bedeutung zu haben scheinen, bezeichnen sie doch recht unterschiedliche Sichtweisen.

Der **Wert** eines hybriden KI-Ansatzes könnte prinzipiell aus unterschiedlichen Perspektiven ermittelt werden, wie z. B. der Wert aus Prozessperspektive, der wahrgenommene Wert durch die Nutzer oder der Wert der getätigten Investition bzw. der dadurch entstehenden organisationalen Transformation. Woraus der Wert dann konkret ermittelt wird, geht weit über den messbaren Nutzen hinaus, wie der Ansatz von Gregor

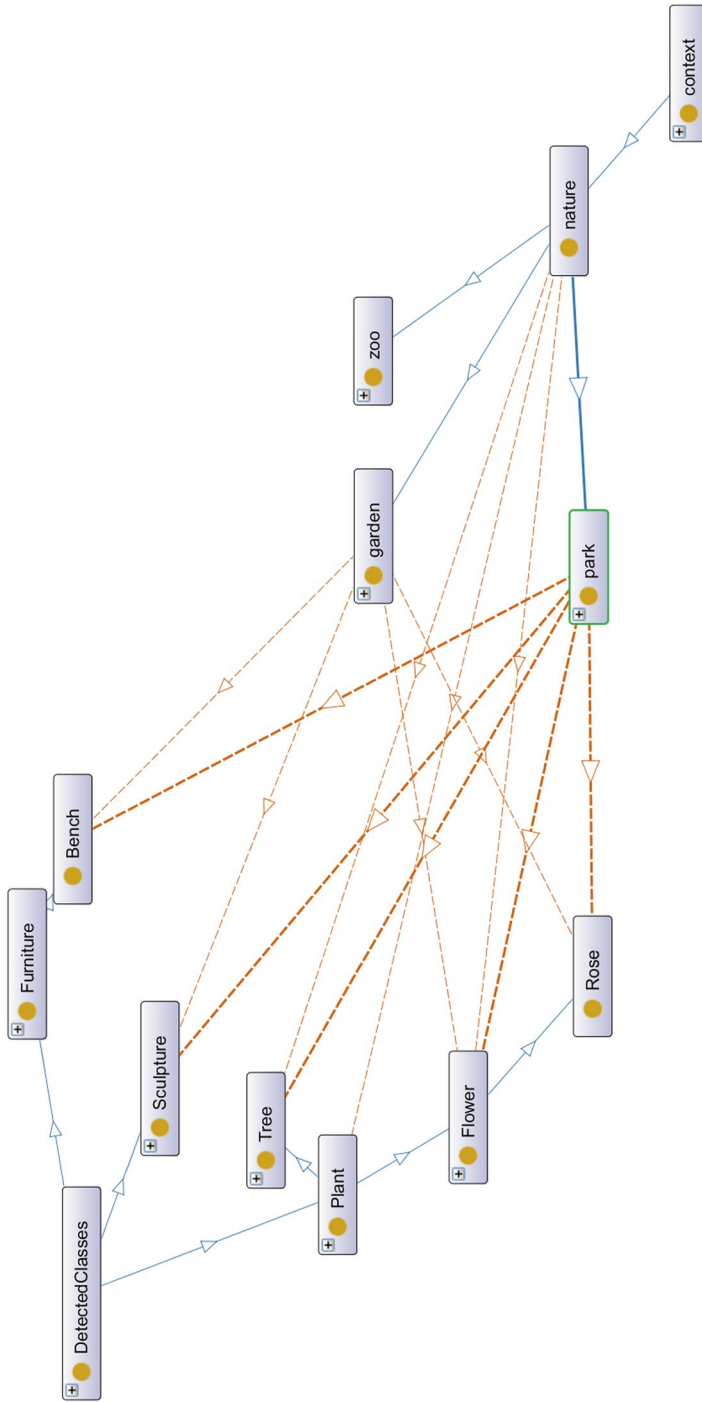


Abb. 16.6 Die Ontologie für die Kontextualisierung von Open Images (Reiz et al. 2022)

Tab. 16.2 Beispiel für die Berechnung der Semantic Confidence (Schritt 1)

Erkanntes Objekt	Konfidenz \vec{o}_{pb}	Größe \vec{o}_{po}	Ergebnis $\vec{o}_i \ S$
Rose	0,936	0,0441	0,980 (Park + Garden)
Plant	0,126	0,588	0,714 Nature
Tree	0,899	0,933	1,832 (Nature + Park)
Tree	0,733	0,615	1,348 (Nature + Park)
Tree	0,340	0,568	0,908 (Nature + Park)
Tree	0,148	0,46	0,608 (Nature + Park)
Tree	0,148	0,553	0,701 (Nature + Park)

et al. (2006) zeigt, die strategische, informatorische, transaktionale und transformative Aspekte betrachten:

- Ein *strategischer Nutzen* liegt vor, wenn eine Investition in IT getätigt wird, um einen Wettbewerbsvorteil zu erlangen und den Marktanteil zu erhöhen, in der Regel in etablierten Geschäftsbereichen. Eines der Merkmale ist, dass die Technologie und Methodik zum Zeitpunkt der Investition auf eine für die Branche neue Art und Weise eingesetzt werden.
- Von einem *Informationsnutzen* kann ausgegangen werden, wenn Verbesserungen in der Informationsinfrastruktur für Kontrolle, Planung oder andere Managementaufgaben erreicht werden, d. h., ein Informationsvorsprung durch die neue IT-Technologie oder -Methode gegenüber der Situation ohne sie besteht.
- Der *transaktionale Nutzen* ist typischerweise mit der Automatisierung oder zumindest Teilautomatisierung von Aufgaben innerhalb eines Unternehmens verbunden. Diese Art von Nutzen ist in der Regel mit der Senkung von Kosten und der Verringerung des Zeitaufwands für Prozesse oder Aufgaben verbunden.
- *Transformationsvorteile* werden durch IT-bezogene organisatorische Umgestaltungen erzielt, die zu neuen Geschäftsmodellen und wesentlichen Änderungen der Wertschöpfung und der Angebote führen. Solche organisatorischen Umgestaltungen können sich in neuen Qualifikationsniveaus, organisatorischem Lernen und industriellen oder betrieblichen Prozessen und Strukturen niederschlagen.

Ein weitverbreiteter Ansatz zur Ermittlung der Wirtschaftlichkeit ist die **Kosten-Nutzen-Analyse** (Mühlenkamp 1994), bei der Kosten und Nutzen einer Maßnahme gegenübergestellt werden. Dieser Ansatz setzt jedoch voraus, dass Kosten und Nutzen bekannt sind bzw. gemessen werden können. Bei den Kosten einer hybriden KI ist dies durch die traditionell bei IT-Kosten übliche Aufteilung in einmalig anfallende Investitionskosten und laufende Betriebskosten möglich. Beispiele für diese Kosten sind:

- *Einmalige* IT-Investitionskosten: grundlegende Kosten (Anschaffungskosten für die Hard- und Software, technische Einrichtung der Infrastruktur, Nebenkosten, Zinsen etc.), Kosten für Informationssysteme und Daten (Analyse/Konzeption der IT-Umgebung, Installation der neuen Hard- und Software, Übertragung bzw. Integration vorhandener Software und Daten, Ausfallkosten durch die Umstellung der IT-Umgebung), Personalkosten (Training von internem IT-Personal, Schulungen von Anwendern und dadurch entstehende Ausfallkosten)
- *Laufende* IT-Betriebskosten: Hardwarekosten (ggf. Miete/Leasing, Energie, Ersatzteile für Reparaturen), Softwarekosten (Lizenzpreise/Update-Garantie für neue Versionen, Aktualisierungspreis zum Bezug von Software-Patches), Wartungsaufwendungen (Kosten für internes IT-Personal, Beseitigung von Störungen, Sicherheitsüberprüfungen etc., Kosten für die Behebung von Störungen durch externe IT-Spezialisten/Servicevertrag), zusätzliche Servicekosten

Für die Bezifferung des Nutzens besteht in der Regel die Herausforderung, die o. g. strategischen, informatorischen, transaktionalen und transformationalen Nutzen messbar zu machen und in konkreten Geldwert umzurechnen. Dies ist bspw. bei strategischem und transformationalem Nutzen nicht trivial und führt oft zu Schätzungen, die die Gefahr von inakzeptablen Vereinfachungen oder Ungenauigkeiten bergen.

Unter dem Begriff **Wirtschaftlichkeit** versteht man im Allgemeinen das Verhältnis von (monetär) quantifizierbaren Kosten zur gemessenen Leistung (Gronau 2019). IT-Lösungen und auch hybride KI-Anwendungen werden dann als wirtschaftlich betrachtet, „wenn die aus der Nutzung des Systems realisierbaren Erträge innerhalb eines bestimmten Betrachtungszeitraums die mit der Systembereitstellung verbundenen Aufwendungen übersteigen“ (Gronau 2019). Im Vergleich zur Kosten-Nutzen-Analyse müssen hier also sowohl ein definierter Zeitraum als auch eine Ertragsrechnung statt einer Nutzenabschätzung vorgenommen werden.

Es kann zwischen der Bereitstellungs- und der Verwendungswirtschaftlichkeit unterschieden werden (Gronau 2019): Die Wirtschaftlichkeit der Bereitstellung beschreibt dabei üblicherweise das Verhältnis der von der bereitstellenden Einheit (z. B. interne IT oder externer Service-Provider) erbrachten Leistungsmenge (z. B. Betriebsstunden der KI-Lösung, Anzahl Transaktionen, bearbeitete Fälle) zu den Kosten der dafür verwendeten Ressourcen (Personal, Miete, Energie etc.). Die Wirtschaftlichkeit der Verwendung fokussiert dagegen auf das Verhältnis der Nutzeneffekte für die Organisation oder den Anwender (z. B. Reduktion des manuellen Aufwands, bessere Qualität der Ergebnisse) zu den aus der Nutzung der IT-Leistungen entstehenden Kosten (Abschreibung, Betriebskosten, Software-Lizenzpreise etc.). Eine verbreitete Methode zur Ermittlung der Wirtschaftlichkeit von KI-Lösungen ist die Erstellung eines Business Case.

Die oben genannten Ansätze stellen nur einen kleinen Teil der in der Literatur diskutierten Verfahren dar. Alle sind prinzipiell für die Bewertung von hybriden KI-Verfahren geeignet, wenn dafür die notwendigen Informationen aus der jeweils betrachteten konkreten Organisation vorliegen, in der die KI eingeführt und genutzt werden soll bzw. wird.

16.6.2 Integrationskosten und Entwicklungsprozesse von hybrider KI und ML

Für den Fokus dieses Abschnitts, die Betrachtung hybrider KI-Verfahren und von Verfahren maschinellen Lernens „in Reinform“ aus ökonomischer Perspektive, wären alle Ansätze aus dem vorangegangenen Abschnitt prinzipiell geeignet. Wir werden uns im Folgenden jedoch nur auf einen Vergleich der Kosten konzentrieren, da zum einen der strategische und transformationale Nutzen bei beiden Verfahren idealerweise gleich sein sollte und zum anderen die Unterschiede zwischen beiden Ansätzen bei den Kosten deutlicher werden.

Auf Basis der im vorangegangenen Abschnitt beschriebenen typischen IT-Kosten lässt sich die nachfolgende Kostenmatrix (Tab. 16.3) motivieren, die auf Müller (2005) zurückgeht und durch Büttner et al. (2014) um Integrationskosten erweitert wurde. Integrationskosten bezeichnen dabei die Kosten, die bei der Integration von IT-Anwendungen (z. B. hybriden KI-Lösungen oder reine maschinelle Lernverfahren) in Organisationen entstehen. Abweichend von Büttner et al. (2014) betrachten wir datenbezogene Kosten separat, weil die Identifikation, Aufbereitung, Integration und Pflege von Daten eine zentrale Voraussetzung für hybride KI- und ML-Anwendungen sind, die bei Nicht-KI-IT-Anwendungen so nicht existiert. Wir unterscheiden also zwischen technischen, organisatorischen und datenbezogenen Kosten. In klassischen Ansätzen würden datenbezogene Kosten zum Großteil unter den technischen Kosten subsumiert.

Die Zeilen der Tabelle bilden die Häufigkeit des Kostenauftretens und die Spalten die Veränderlichkeit der Kosten ab. Beispiele für einmalige und laufende Kosten wurden oben schon gegeben. Fixe Kosten sind unabhängig von der Häufigkeit und Nutzungsintensität der Anwendung einer KI-Lösung und über einen gewissen Zeitraum konstant, während variable Kosten davon abhängig sind und meistens auch mit der Häufigkeit steigen.

Zum besseren Verständnis der datenbezogenen Kosten stellen wir die bei datenseitig durchzuführenden Aufgaben dar, was auch einen Vergleich zwischen symbolischer KI (Ontologieentwicklung) und maschinellem Lernen (Entwicklung von Detektoren) ermöglicht. Tab. 16.4 zeigt für typische Entwicklungsphasen, welche Schritte jeweils bei der Ontologieentwicklung und bei der Entwicklung eines maschinellen Lernverfahrens notwendig sind. In der ersten Entwicklungsphase wird definiert, welche Ziele die geplante KI-Anwendung haben soll und welche Probleme dadurch zu lösen sind, d. h. der Fokus der Anwendung. Dies resultiert auch in einer Abgrenzung, was nicht im Entwicklungsprojekt betrachtet werden soll. Danach erfolgt die Spezifikation der genauen Anforderungen an die KI-Anwendung, was bei der Ontologieentwicklung häufig über sogenannte Kompetenzfragen erfolgt und bei der Entwicklung maschineller Lernverfahren über fachliche Anforderungen und entsprechende messbare Erkennungsqualität. In der Vorbereitungsphase ist für Ontologien neben der Werkzeugauswahl das Identifizieren bzw. Beschaffen des notwendigen Domänenwissens von Bedeutung, was häufig durch Taxonomien aus dem Fachgebiet oder Fachexperten bereitgestellt wird. Bei

Tab. 16.3 Matrix der Integrationskosten

	Fixe Kosten	Variable Kosten
Einmalig auftretende Kosten	Technische Kosten: <ul style="list-style-type: none"> • Anschaffungskosten KI-Anwendung • Anschaffungskosten für Schnittstellen • Entwicklungskosten für KI-Anwendung • Implementierungskosten • Infrastrukturkosten Datenbezogene Kosten: <ul style="list-style-type: none"> • Anschaffungskosten für Daten und Modelle Organisatorische Kosten: <ul style="list-style-type: none"> • Kosten für Veränderungen während Anschaffung, Entwicklung und Implementierung • Kosten für Tests 	Technische Kosten: <ul style="list-style-type: none"> • Kosten für Änderungen an KI-Produkt • Einmalige Lizenzkosten Datenbezogene Kosten: <ul style="list-style-type: none"> • Kosten für Anpassungen an den KI-Modellen Organisatorische Kosten: <ul style="list-style-type: none"> • Kosten für externe Berater • Schulungskosten • Change-Management-Kosten
Laufend auftretende Kosten	Technische Kosten: <ul style="list-style-type: none"> • Kosten für Systemverfügbarkeit • Kosten für Systemsicherheit Datenbezogene Kosten: <ul style="list-style-type: none"> • Kosten für Bereitstellung der KI-Modelle Organisatorische Kosten: <ul style="list-style-type: none"> • Servicecenterkosten 	Technische Kosten: <ul style="list-style-type: none"> • Wartungskosten Anwendung und Schnittstellen • Betriebskosten (Strom etc.) • Kosten für temporäre Lizenzen • Kosten durch Ausfallzeiten Datenbezogene Kosten: <ul style="list-style-type: none"> • Wartungskosten Daten • Kosten für Nutzung der KI-Modelle Organisatorische Kosten: <ul style="list-style-type: none"> • Kosten für Pflege und Verwaltung • Kontrollkosten

ML ist die Plattform zur Bearbeitung der Daten auszuwählen und es sind entsprechende Trainingsdaten zu beschaffen. In der Entwicklungsphase folgen für die Ontologien das Entwickeln der Ontologie auf Basis des Domänenwissens und ggf. das Füllen der Wissensbasis, während bei ML-Ansätzen die Modellentwicklung, inkl. Trainieren des Modells mit den beschafften Daten, erfolgen muss. Die Einführungsphase stellt die Bereitstellung von Software-Tools für den Endanwender bzw. das Integrieren der Ontologie oder des trainierten Modells in bestehende Anwendungen in den Mittelpunkt. Wenn im laufenden Betrieb Wartungsbedarf entdeckt wird, führt dies bei Ontologien meistens zu Ergänzungen von Konzepten, Relationen oder Regeln, während beim ML die Modelle oft nachtrainiert werden müssen.

Tab. 16.4 Entwicklungsphasen für Ontologien und maschinelles Lernen im Vergleich

Entwicklungsphasen	Ontologieentwicklung	Entwicklung maschinelles Lernen
<i>Abgrenzung</i>	Fokus der Anwendung, Definition von Grenzen	
<i>Anforderungen</i>	Spezifikation von Anforderungen, z. B. durch Kompetenzfragen	Spezifikation von Anforderungen, z. B. über fachliche Kriterien und messbare Qualität
<i>Vorbereitung</i>	Werkzeugauswahl, Quellen für Domänenwissen identifizieren	Plattformauswahl, Beschaffung oder Erstellung von Trainingsdaten
<i>Entwicklung</i>	Ontologie entwickeln (Begriffe, Konzepte, semantische Zusammenhänge, Regeln)	Modell auswählen und anpassen oder entwickeln
	Wissensbasis füllen	Modell mit Daten trainieren (z. B. als Klassifikationsmodell)
<i>Einführung</i>	Bereitstellung einer Schnittstelle zu Ontologiepopulation und Nutzung	Bereitstellen der KI-Anwendung, bspw. über einen Zugangspunkt (Endpoint) in der Cloud
<i>Betrieb</i>	Nutzung der KI-Lösung	
<i>Wartung</i>	Anpassungsbedarf aus dem Betrieb (z. B. neue Konzept)	Weitere Trainingsdaten aus Betrieb, Nachtrainieren des Modells

Anhand dieser Entwicklungsphasen (Tab. 16.4) und der unterschiedlichen Kostenarten (Tab. 16.3) können nun genauere ökonomische Betrachtungen vorgenommen werden. Dabei stehen drei Fragen im Mittelpunkt des Interesses:

- 1 Welche Unterschiede bestehen zwischen einem ML-Ansatz ohne Nutzung vor-trainierter Modelle bzw. existierender Datensets und einem Ansatz Out of the Box, d. h. mit Nutzung existierender Datensets und vortrainierter Modelle?
- 2 Welcher Unterschied besteht zwischen hybriden Ansätzen und nachtrainierten KI-Out-of-the-Box-Lösungen?
- 3 Welche Unterschiede bestehen bei den variablen Kosten zwischen einem reinen ML-Ansatz und einem hybriden Ansatz? Hier liegt der Fokus also auf den Kosten während des laufenden Betriebs.

Bei allen drei Fragestellungen ist vor allem der Vergleich der datenbezogenen Kosten relevant und soll daher in den nächsten Absätzen im Fokus stehen. Organisatorische Kosten sind für alle KI-Ansätze vergleichbar, da alle in die betrieblichen Abläufe integriert werden müssen und ähnliche Wartungs- bzw. Kontrolllisten verursachen. Technische Kosten sind auch bei beiden Ansätzen ähnlich, was Anschaffungskosten für Software und Schnittstellen sowie Kosten für Sicherheit und Verfügbarkeit betrifft. Bei ML-Ansätzen ist ein Teil dieser Kosten jedoch aktuell deutlich höher als bei Ontologien, da hö-

here Rechenleistungen erforderlich sind, was Serverbetrieb und Stromkosten deutlich verteuert.

16.6.3 Vergleich der Entwicklungskosten von ML-Ansätzen mit und ohne Out-of-the-Box-Bestandteilen

Für den Vergleich zwischen einem ML-Ansatz ohne Nutzung vortrainierter Modelle bzw. existierender Datensets und ML-„Out-of-the-Box“, d. h. mit Nutzung existierender Datensets und vortrainierter Modelle, sind vor allem die einmalig auftretenden, datenbezogenen Kosten aufschlussreich, da sich hier die Vorbereitungs- und Entwicklungsphase besonders unterscheiden. In der Entwicklungsphase ist die Modellentwicklung besonders wichtig. Hier geht es vereinfacht formuliert darum, für das gegebene Problem geeignete Verfahren des maschinellen Lernens und die entsprechende Konfiguration bzw. Parametrierung zu finden und durch einen Algorithmus umzusetzen. Je nach Art der Daten (Zeitreihen, umfangreiche Dokumente, kurze Texte, Grafiken, Bewegtbilder, Audiodaten etc.), deren Eigenschaften (uni-/multiskalar, uni-/multivariat, Auflösung, Rauschen etc.) gibt es bereits Erfahrungen, welche Art von Verfahren (künstliche neuronale Netze, symbolische KI, approximative Verfahren etc.) geeignet sein könnten. Im Falle einer klar erkennbaren, einfachen Problemstruktur ist die Modellentwicklung dann in wenigen Wochen abgeschlossen. In Fällen unbekannter Verteilung und spezieller Eigenschaften des Phänomens kann dies aber auch viele Monate dauern.

Da konkrete Zahlen zum Aufwand solcher Modellentwicklungen und Implementierungen in der Literatur nicht verfügbar sind, bedienen wir uns zwecks der Konkretisierung eigener Erfahrungen. Das mit Abstand einfachste Entwicklungsprojekt (Diadiushkin et al. 2019) untersuchte die Machbarkeit eines maschinellen Lernansatzes in der Betrugserkennung von Zahlungsverkehr. Die Testdaten wurden generiert, d. h., diese lagen mit genau bekannten Eigenschaften vor. Es wurden nur Standard-ML-Verfahren verwendet. Der Aufwand für die Entwicklung lag bei 20 Arbeitstagen à 8 h durch einen entsprechenden Ingenieur. Ein sehr aufwendiges Entwicklungsprojekt aus dem medizinischen Umfeld (Bej et al. 2021) musste dagegen sowohl das Datenset erst beschaffen bzw. erstellen und in der Modellentwicklung verschiedene Ansätze erproben und anpassen, bevor eine hinreichend genaue und zuverlässige Erkennung erreicht wurde. Insgesamt waren hier etwa drei Jahre (entsprechen etwa 3350 produktiven Arbeitsstunden) erforderlich. In einem weiteren Vorhaben (Görgen 2003) war vor allem der Vorbereitungsanteil aufwendig, da kein Datenset vorlag. Hier wurden etwa 12.000 Versicherungsdokumente digitalisiert, nach Versicherungsunternehmen sortiert und dann in verschiedene Arten des Dokuments (Provisionsabrechnung, Schadensfallmeldung, Beitragsabrechnung etc.) klassifiziert. Dies erforderte zwischen 30 und 55 s pro Dokument mit einem Durchschnittswert von 39 s. Anzumerken ist, dass die Dokumente selbst bereits vorlagen. Muss für ein Datenset auch noch die Beschaffung der Daten erfolgen und nicht nur deren „Labeling“, erhöht sich der Aufwand signifikant. Für eine hinreichende Genauigkeit der Er-

kennung wird je nach Anzahl der relevanten Merkmale eine Anzahl von mindestens 1000 Dokumente pro Kategorie empfohlen.

Die obigen Zahlen machen deutlich, dass Modellentwicklung und Vorbereitung des Datensets substanzielle Kosten verursachen. Bei Out-of-the-Box-Bestandteilen ist ein Großteil der beschriebenen Aufwände nicht mehr erforderlich, was auch die Kosten substanziell senkt.

Vergleich der Entwicklungskosten von hybrider KI mit nachtrainierten Out-of-the-Box-Lösungen

Auch für den Vergleich zwischen hybriden Ansätzen und nachtrainierten KI-Out-of-the-Box-Lösungen sind die Vorbereitungs- und Entwicklungskosten entscheidend. Der wesentliche Unterschied zwischen einer KI-Lösung Out of the Box und einer hybriden KI-Lösung unter Verwendung von Out of the Box ist die Art und Weise, wie eine hinreichende Präzision der Rekognition erreicht wird. Bei Out of the Box müssen die vortrainierten Modelle durch weiteres Training mit spezifischen Datensets aus dem Anwendungsfall weiter ertüchtigt werden, d. h., hier sind das Beschaffen und „Labeln“ von Trainingsdaten und das anschließende Trainieren des Modells erforderlich. Bei hybriden Ansätzen wird das vortrainierte Modell idealerweise nur minimal erweitert oder so genutzt, wie es ist, und die Präzision mittels der ergänzenden Ontologie erreicht. In diesem Fall ist eine Ontologieentwicklung erforderlich. Dieses Vorgehen ist jedoch davon abhängig, wie gut das vorhandene Datenset bzw. das vortrainierte Modell die Anwendungsdomäne abdeckt. Ist die Abdeckung unzureichend, muss das Modell nachtrainiert werden, was aber im Wesentlichen geringeren Umfang erfordert als bei reinem maschinellen Lernen.

Der Aufwand für die hybride KI und nachtrainierte Out-of-the-Box-Lösungen hängt von der Anzahl der Merkmale bzw. Merkmalsausprägungen ab, die für die Rekognition relevant sind. Diese Merkmale und Merkmalsausprägungen werden als Konzepte in der Ontologie repräsentiert und werden entweder bereits von der Out-of-the-Box-Lösung erkannt oder müssen nachtrainiert werden. Für den Aufwandsvergleich unterscheiden wir drei Mengen von Merkmalen (bzw. Konzepten in der Ontologie):

- M_{basis} : Merkmale bzw. Merkmalsausprägungen, die bereits im Out-of-the-Box-Modell enthalten sind, d. h. durch dieses erkannt werden,
- M_{erg} : Merkmale bzw. Merkmalsausprägungen, die noch nicht in der Out-of-the-Box-Lösung enthalten sind und nachtrainiert werden müssen,
- M_{ableit} : Merkmale bzw. Merkmalsausprägungen, die aus anderen Merkmalen bzw. Merkmalsausprägungen hergeleitet werden können. Aus Braut und Bräutigam könnte man beispielsweise Brautpaar herleiten.

Vorbereitungsaufwand: Verwendet man keine hybride KI-Lösung müssen M_{ableit} in der Out-of-the-Box-Lösung nachtrainiert werden, d. h., hierfür sind in der Vorbereitungsphase geeignete Daten zu beschaffen und zu labeln. M_{erg} sind bei beiden Lösungsan-

sätzen nachzutrainieren, da sie ja weder abgeleitet werden können noch im vortrainierten Modell vorliegen. M_{basis} müssen in die Ontologie integriert werden. Der Vorbereitungs- aufwand für die Integration eines Konzepts (= Merkmal bzw. Merkmalsausprägung) in eine Ontologie ist deutlich geringer als das Nachtrainieren eines Merkmals, da nur die Relationen zwischen dem Konzept und den vorhandenen Konzepten bekannt sein muss.

Entwicklungsaufwand: Die Entwicklung bzw. Anpassung einer Ontologie für den hybriden Ansatz und auch die Entwicklung bzw. Anpassung eines Modells für das maschinelle Lernen sind komplexe Prozesse, für die der Aufwand von einer Vielzahl von Faktoren abhängt, wie bspw. der Anzahl der Relationen und Axiome bei Ontologien oder der Gleichmäßigkeit der Verteilung beim maschinellen Lernen. Erfahrungswerte zeigen zudem, dass beide Entwicklungsaufgaben aufwandsmäßig dieselbe Größenordnung annehmen können. Wir werden bei der Vergleichsrechnung daher annehmen, dass sie vom Aufwand her vergleichbar sind.

Abb. 16.7 zeigt einen Vergleich beider Ansätze, wobei die rote Linie den Aufwand für die hybride Lösung und die blaue Linie für den nachtrainierten Out-of-the-Box-Ansatz darstellt. H_0 und O_0 zeigen den Entwicklungsaufwand für die Ontologie bzw. das Modell für den ML-Ansatz, die, wie oben erläutert, für die Modellrechnung als gleich angenommen werden. Beim Vergleich wurde angenommen, dass die weit überwiegende Anzahl der Merkmale (M_{basis}) bereits in der Out-of-the-Box-Lösung enthalten ist und nur in die Ontologie integriert werden muss. Weiterhin wurde angenommen, dass eine geringere Anzahl an Merkmalen abgeleitet (M_{ableit}) werden kann bzw. ergänzt werden muss (M_{erg}).

In vielen Anwendungsfällen, bei denen Datensets und vortrainierte Modelle schon zur Verfügung stehen, macht die Kombination mit symbolischer KI eine KI-Anwendung überhaupt erst möglich, da das Sammeln eigener Daten und Entwickeln eigener Modelle

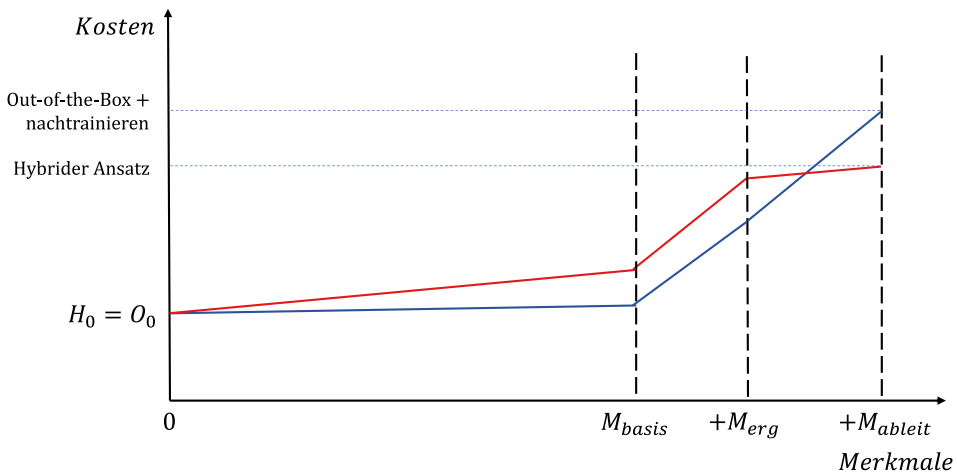


Abb. 16.7 Vergleich des Aufwands für einen hybriden KI-Ansatz mit nachtrainierter Out-of-the-Box-Lösung

wirtschaftlich zu aufwendig wären, die verfügbaren vortrainierten Datensets und Modelle ohne symbolische KI aber zu unpräzise sind.

Vergleich der variablen Kosten von hybridem Ansatz und ML

Am Beispiel SemOI lässt sich nun auch konkret berechnen, welche Kostenvorteile der hybride Ansatz gegenüber einem rein maschinellen Lernansatz bietet. Wir betrachten dazu vor allem laufend auftretende variable Kosten.

Nehmen wir an, wir haben M Bilder und bestimmen die relevanten Objekte in Bild k . Außerdem stehen N Klassifikatoren zur Verfügung, die mit dem ML-Ansatz trainiert wurden, einer für jedes Objekt. Welche Objekte in welcher Szene relevant sind, wird im semantischen Netz weiter spezifiziert. Die Bestimmung der relevanten Objekte in k kann nur mit dem ML-Ansatz oder mit dem kombinierten (hybriden) Ansatz von semantischem Netz und ML erfolgen.

Im ersten Fall (ohne semantisches Netz) müssen zur Identifizierung aller relevanten Objekte im Bild k alle Klassifikatoren angewendet werden, da das semantische Netz keine Informationen darüber liefert, welche Szenen existieren, welche Objekte diese Situationen charakterisieren und welche zusätzlichen Objekte in einer Szene relevant sind. In diesem Fall besteht der Aufwand A_k der Identifizierung relevanter Objekte für Bild k darin, den Ansatz mit allen verfügbaren Klassifikatoren anzuwenden.

Bei Verwendung des semantischen Netzes müssen die Klassifikatoren des ML-Ansatzes zunächst zur Identifizierung der Szene verwendet werden. Die Anzahl der Szenen ist viel kleiner als die Gesamtzahl der Objekte für M und die Anzahl der Klassifikatoren, die zur Bestimmung einer bestimmten Szene erforderlich sind, ist ebenfalls viel kleiner als die Gesamtzahl der Objekte. Im zweiten Schritt sind nur die Objekte relevant, die mit der identifizierten Szene in Verbindung stehen, d. h., es muss nur diese bestimmte Menge von Klassifikatoren für den ML-Ansatz verwendet werden.

Der Aufwand für die Bestimmung der relevanten Objekte in k setzt sich zusammen aus dem Aufwand für die Erkennung der Szene plus dem Aufwand für die Ausführung der Klassifikatoren für die in der erkannten Situation relevanten Objekte. Dies wird in Abb. 16.8 veranschaulicht, wobei e_0 der Aufwand für die Vorbereitung der Szenenerkennung für das Bild k bei der anfänglichen Analyse des semantischen Netzes, e_i der Aufwand für die Ausführung eines einzelnen Klassifikators und N' die Anzahl der erforderlichen Klassifikatoren zur Bestimmung aller relevanten Objekte ist. Wenn für das Bild k alle Objekte (und verfügbaren Klassifikatoren) relevant sind, ist N gleich N' . Dann ist der Aufwand für die Objekterkennung ohne den Einsatz des semantischen Netzes geringer als mit dessen Einsatz.

Abb. 16.8 zeigt die Abhängigkeitsgraphen des Aufwands der Objekterkennung von deren Anzahl mit und ohne Verwendung des semantischen Netzes. Zu Beschreibungszwecken wurde angenommen, dass der Aufwand für die Ausführung eines bestimmten Klassifikators gleich ist, und zwar mit a_{mean} . Dann ist der Aufwand für die Bestimmung aller relevanten Objekte in k ohne Verwendung des semantischen Netzes $A = a_{mean} * N$.

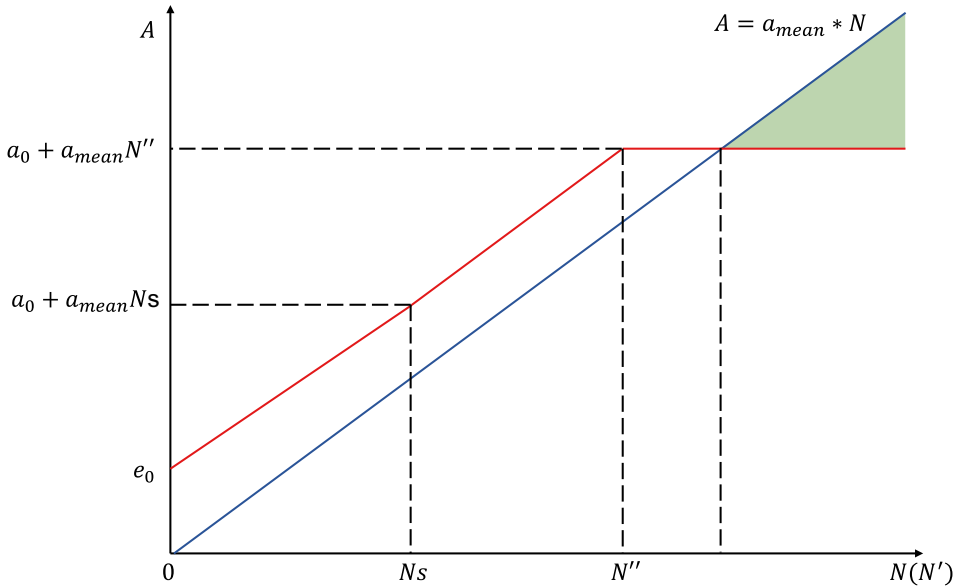


Abb. 16.8 Vergleich des Aufwands in SemOI für die Objekterkennung mit hybridem und rein maschinellem Lernverfahren

Bei Verwendung des semantischen Netzes für Bild k ist ein anfänglicher Aufwand a_0 erforderlich, um das semantische Netz zu initialisieren und N' Objekte auszuwählen, die für die Szenenerkennung verwendet werden sollen. In diesem Fall ist $N' \in [1, N'']$, wobei N'' die Anzahl der angewandten Klassifikatoren für die Objekte ist, die für die Szenenerkennung benötigt werden, die in der Regel nicht mehr als 5–6 beträgt. Es ist zu beachten, dass der Wert von N'' umso kleiner ist, je eindeutiger die Objekte die relevanten Szenen beschreiben. Wenn die Situation nur mit vielen Objekten beschrieben werden kann, wird der Wert von N'' hoch sein (d. h., es muss eine große Anzahl von Klassifikatoren verwendet werden). Gibt es eine Menge von Objekten, die nur für eine bestimmte Szene von Bedeutung sind, wird der Wert von N'' niedrig sein. Der erforderliche Aufwand für die Objekterkennung mithilfe des semantischen Netzes wird dann für $N' < N''$ nach der Formel $A = a_0 + a_{mean}N'$ und für $N' > N''$ nach der Formel $A = a_0 + a_{mean}N''$ bestimmt. N_s ist die Anzahl der Klassifikatoren zur Bestimmung einer Szene und $A_k = a_0 + a_{mean}N'$ der erforderliche Aufwand. Das grün gefärbte Dreieck ist der erwartete Nutzen des hybriden Ansatzes.

16.7 Diskussion und Fazit: Potenziale und Grenzen des Ansatzes

Künstliche Intelligenz hat das Potenzial, bestehende Prozesse grundlegend zu transformieren, neue Geschäftsmodelle und Geschäftsfelder zu erschließen und in einer schnelllebigen Geschäftswelt den entscheidenden Wettbewerbsvorteil zu erlangen. Die Implementation und Anwendung sind jedoch zumeist von großem Investitions- und Ressourcenaufwand geprägt. Das Training benötigt große, zumeist händisch annotierte Datenmengen sowie enorme Rechenkapazitäten. Weiter bedingt jedwede Änderung des neuronalen Netzes ein erneutes Training. Auch das Ausführen der Rekognitionsaufgaben ist ressourcenintensiv und damit teuer. Dies macht die Anwendung von ML-Verfahren bisher insbesondere in schnelllebigen Bereichen fraglich.

Die Kombination von subsymbolischer mit symbolischer KI kann dessen Nachteile mitigieren. Zwar passen die verfügbaren Trainingsdatensets, vortrainierten Modelle oder SaaS-Lösungen oft nicht genau auf den gegebenen Anwendungsfall. Jedoch erlaubt das semantische Netz die Übersetzung der zumeist allgemein gehaltenen ML-Ergebnisse auf die gegebene Domäne und kann größere vorgefertigte Lösungen mit kleineren eigenen ML-Modellen sinnvoll verknüpfen. Dies ermöglicht das schnelle Aufsetzen und Anpassen einer personalisierten KI-Lösung unter Umgehung der größten Kostentreiber dieser Technologie.

Der hier aufgezeigte Ansatz basiert auf der Kontextualisierung von frei verfügbaren Rekognitionslösungen. Dieser Vorteil des gezeigten Ansatzes ist zugleich seine Limitation. Für eine erfolgreiche Übersetzung der Ergebnisse muss die vorgeschaltete ML-Lösung zumindest teilweise relevante Objekte für den gegebenen Anwendungsfall erkennen. Sehr spezifische Szenarien, wie z. B. die Identifikation von schadhafte Bauteilen in einer Produktionsstraße, werden sich mit diesem hybriden Ansatz kaum vereinfachen lassen. Für Anwendungen, die sich jedoch über die Kombination von allgemein verfügbaren Objekten beschreiben lassen, bietet der hier gezeigte Ansatz eine kostengünstige und flexible Architektur für das Erstellen einer KI-Lösung.

16.8 Empfehlungen

Das Training neuronaler Netze ist kostenintensiv und ein möglicher Hinderungsgrund für den Einsatz von ML. Zwar gibt es bereits eine Vielzahl von vorgefertigten Lösungen, diese passen jedoch oft nicht auf den eigenen Anwendungsfall. Die Verbindung dieser direkt nutzbaren ML-Lösungen mit einer Semantik erlaubt eine schnelle und kostengünstige Adaption auf die eigene Domäne.

1. Nicht alle KI-Lösungen benötigen individuell erstellte Modelle mit eigens kuratierten Daten. Eine einzelne oder eine Kombination aus entfernt passenden Lösung lässt sich mittels einer Semantikkomponente an den gegebenen Use Case anpassen.
2. Es gibt eine Vielzahl von nützlichen ML-Verzeichnissen, zum Beispiel speziell für die verwendeten Frameworks (z. B. Tensorflow Hub, Pytorch Hub), oder allgemeinere, externe Quellen wie paperswithcode.com, kaggle.com und huggingface.co. Auch SaaS-Rekognitionslösungen der verschiedenen Cloud-Anbieter lassen sich nutzen.
3. Über einen semantischen Zwilling der ML-Detektoren im semantischen Netz ist dieser mit kontextualisiertem Wissen verbunden. Auf diese Weise lassen sich zwei ML-Verfahren miteinander kombinieren oder komplexe Abhängigkeiten modellieren.
4. Subsymbolische (ML) und symbolische (Semantik-)KI arbeiten fundamental unterschiedlich. Erstere errechnet Wahrscheinlichkeiten, Zweitere nimmt, im Falle semantischer Netze, eine Grundwahrheit an. Die Nutzung von verschiedenen Erkennungsattributen, wie die vorgeschlagene Semantic Confidence, erlaubt die Verknüpfung der unterschiedlichen Sichtweisen.
5. Die Kombination von maschinellem Lernen und Semantik bietet erhebliche ökonomische Einsparpotenziale. Eine objektive Kostenbetrachtung für den gegebenen Anwendungsfall kann die Architekturentscheidung erleichtern.

Literatur

- Bej, S., Davtyan, N., Wolfien, M., Nassar, M. & Wolkenhauer, O. (2021) „LoRAS: an oversampling approach for imbalanced datasets“, *Machine Learning*, Vol. 110, No. 2, S. 279–301.
- Buda, M., Saha, A., Walsh, R., Ghate, S., Li, N., Swiecicki, A., Lo, J. Y. & Mazurowski, M. A. (2021) „A Data Set and Deep Learning Algorithm for the Detection of Masses and Architectural Distortions in Digital Breast Tomosynthesis Images“, *JAMA network open*, Vol. 4, No. 8, e2119100.
- Büttner, D., Fischer, D. & Stelzer, D. (2014) Entwicklung eines Hilfsmittels zur Bestimmung der Kosten der Integration von Informationssystemen: Eine Erweiterung des Ilmenauer Integrationsmodells [Online], Ilmenau, Techn. Univ. Inst. für Wirtschaftsinformatik. Verfügbar unter <http://nbn-resolving.de/urn:nbn:de:gbv:ilm1-2014200098>.
- Diadiushkin, A., Sandkuhl, K. & Maiatin, A. (2019) „Fraud Detection in Payments Transactions: Overview of Existing Approaches and Usage for Instant Payments“, *Complex Systems Informatics and Modeling Quarterly*, No. 20, S. 72–88.
- Geiger, A., Lenz, P., Stiller, C. & Urtasun, R. (2013) „Vision meets Robotics: The KITTI Dataset“, *International Journal of Robotics Research (IJRR)*.
- Görgen, L. (2003) A Comparison of Open Source and Commercial Solutions for Document Classification, Bachelor Thesis.
- Gregor, S., Martin, M., Fernandez, W., Stern, S. & Vitale, M. (2006) „The transformational dimension in the realization of business value from information technology“, *The Journal of Strategic Information Systems*, Vol. 15, No. 3, S. 249–270 [Online]. <https://doi.org/10.1016/j.jsis.2006.04.001>.
- Gronau, N. (2019) Wirtschaftlichkeit von Informationssystemen [Online]. Verfügbar unter <https://wi-lex.de/index.php/lexikon/informations-daten-und-wissensmanagement/informations->

[management/information/wirtschaftlichkeit-von-informationssystemen/](#) (Abgerufen am 27 Februar 2023).

- Iansiti, M. & Lakhani, K. R. (2020) *Competing in the age of AI: Strategy and leadership when algorithms and networks run the world* [Online], Boston, MA, Harvard Business Review Press. Verfügbar unter <https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=5846337>.
- Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L. & Dollár, P. (2014) *Microsoft COCO: Common Objects in Context* [Online]. Verfügbar unter <https://arxiv.org/pdf/1405.0312>.
- Mühlenkamp, H. (1994) *Kosten-Nutzen-Analyse*, München, Wien, Oldenbourg.
- Müller, A. (2005) *Wirtschaftlichkeit der Integration: Eine ökonomische Analyse der Standardisierung betrieblicher Anwendungssysteme*, Wiesbaden, Deutscher Universitätsverlag.
- Panayotov, V., Chen, G., Povey, D. & Khudanpur, S. (2015) „Librispeech: An ASR corpus based on public domain audio books“, 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015): South Brisbane, Queensland, Australia, 19–24 April 2015. South Brisbane, Queensland, Australia, 4/19/2015–4/24/2015. Piscataway, NJ, IEEE, S. 5206–5210.
- Reiz, A., Albadawi, M., Sandkuhl, K., Vahl, M. & Sidin, D. (2021) „Towards More Robust Fashion Recognition by Combining of Deep-Learning-Based Detection with Semantic Reasoning“, Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021). Stanford University, Palo Alto, California, USA, CEUR-WS.
- Reiz, A., Lantow, B. & Sandkuhl, K. (2022) „SemOI2 – Building Adaptive and Cost-Effective Recognition Applications With Semantic Augmentation“, AAAI Spring Symposium: MAKE.
- Szeliski, R (Hg.) (2022) *Computer Vision*, Cham, Springer International Publishing. <https://doi.org/10.1007/978-3-030-34372-9>.
- van Harmelen, F. & Teije, A. ten (2019) „A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems“, *Journal of Web Engineering*, Vol. 18, No. 1, S. 97–124.
- Weitz, A. & Hensley, D. (2022) *GitHub - toinfinityai/InfiniteRep* [Online], Infinity AI. Verfügbar unter <https://github.com/toinfinityai/InfiniteRep> (Abgerufen am 19 Januar 2023).
- Wu, X., Sahoo, D. & Hoi, S. C. (2020) „Recent advances in deep learning for object detection“, *Neurocomputing*, Vol. 396, S. 39–64.
- Zhou, B., Lapedriza, A., Khosla, A., Oliva, A. & Torralba, A. (2018) „Places: A 10 Million Image Database for Scene Recognition“, *IEEE transactions on pattern analysis and machine intelligence*, Vol. 40, No. 6, S. 1452–1464.

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.





Interaktive Entscheidungsfindung unter Einsatz von maschinellem Lernen und Regelbasierten Systemen

17

Knut Hinkelmann , Mengyang Li, Maja Spahic-Bogdanovic 
und Hans Friedrich Witschel 

Kernaussagen

1. Künstliche Intelligenz trägt dazu bei, konsistente Entscheidungen zu treffen.
2. Komplexe Anwendungen erfordern eine Kombination von Technologien.
3. Künstliche Intelligenz ist ein Werkzeug, um die Menschen zu entlasten, aber sie kann sie nicht ersetzen.
4. Maschinelles Lernen stößt an Grenzen bei unvollständigen Daten und sich ändernden Situationen.
5. Die Sicherstellung von Compliance erfordert nachvollziehbare Schlussfolgerungen.

17.1 Einleitung

In diesem Beitrag befassen wir uns mit der Entscheidung zur Zulassung von potenziellen Studierenden zu einem Studiengang. Die Anwendung steht stellvertretend für Entscheidungen, bei denen aufgrund von vorgegebenen Kriterien einigen Menschen der

K. Hinkelmann (✉) · M. Li · M. Spahic-Bogdanovic · H. F. Witschel
Fachhochschule Nordwestschweiz FHNW, Olten, Schweiz
E-Mail: knut.hinkelmann@fhnw.ch

M. Li
E-Mail: mengyang.li@students.fhnw.ch

M. Spahic-Bogdanovic
E-Mail: maja.spahic@fhnw.ch

H. F. Witschel
E-Mail: hansfriedrich.witschel@fhnw.ch

Zugang zu Gütern und Dienstleistungen gewährt und anderen verweigert wird. Weitere Beispiele sind die Gewährung von Beihilfen wie Sozialhilfe oder Stipendien, die Vergabe von Krediten, der Anspruch auf eine medizinische Behandlung oder die Vergabe von Fördermitteln.

Wegen der Komplexität der Entscheidungskriterien ist die Prüfung der Eignung in der Regel äußerst zeit- und kostenaufwendig (Dagar 2021; Hänig et al. 2023; Sarkar 2021). Hinzu kommt, dass die Kriterien teilweise auch von der subjektiven Sichtweise der Fachleute abhängen (Hollenweger 2011), was gelegentlich zu intransparenten und inkonsistenten Entscheidungen führen kann. Da solche Entscheide den Verlauf des Lebens von Personen entscheidend beeinflussen können, ist bei der Entscheidungsfindung besondere Sorgfalt geboten.

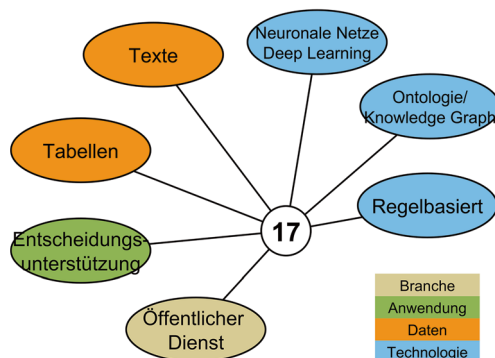
Der Einsatz von künstlicher Intelligenz könnte dabei helfen, transparente und konsistente Entscheidungen zu treffen und gleichzeitig effizienter und kostengünstiger zu arbeiten. Ziel ist es, möglichst viele Entscheide automatisch durchzuführen und menschliche Entscheider nur in unklaren Fällen zu involvieren.

Der Einsatz von künstlicher Intelligenz stellt allerdings Herausforderungen, die wir am Beispiel der Aufnahme in den Masterstudiengang Business Information Systems der Fachhochschule Nordwestschweiz analysieren. Im Studienreglement ist das Aufnahmekriterium in den Studiengang sehr einfach definiert: Aufgenommen wird, wer einen Bachelorabschluss in einem verwandten Gebiet mit guten Noten und Berufserfahrung hat. Ein naives Vorgehen wäre, mit maschinellem Lernen aus den dokumentierten Fällen ein Modell zu trainieren. Dies stößt jedoch an Grenzen:

- **Variabilität:** In der Praxis gibt es eine große Variabilität. Da der Studiengang international ist, kommen die Studierenden aus verschiedenen Ländern mit einem breiten Spektrum an Bachelorabschlüssen und ganz unterschiedlichen Notensystemen. So kommen immer wieder Fälle, die mit vorherigen Fällen nicht vergleichbar sind. Um die Umrechnung von allen Notensystemen lernen zu können, ist eine sehr große Anzahl von Fällen notwendig, die an einer Hochschule nicht verfügbar sind.
- **Compliance:** Es muss sichergestellt sein, dass die verbindlichen Vorgaben eingehalten werden, die sich aus den Gesetzen und der Studienordnung ergeben. Ein verbindliches Kriterium ist, dass der Bachelorabschluss von einer anerkannten Hochschule stammt. Ob eine bisher noch nicht bekannte Hochschule anerkannt ist, lässt sich aus historischen Fällen nicht herleiten, sondern muss explizit überprüft werden, während bei Noten und Berufserfahrung ein Spielraum besteht und Kompensationen möglich sind.
- **Veränderung:** Im Laufe der Jahre wurden die Aufnahmekriterien mehrfach angepasst, wodurch nicht alle Entscheide aus früheren Jahren auf aktuelle Fälle übertragbar sind.
- **Nachvollziehbarkeit:** Ungerechtfertigte Ablehnungen können angefochten werden. Daher muss die Entscheidung transparent, nachvollziehbar und überprüfbar sein.

Um mit der Komplexität der Entscheidung umzugehen, besteht die Möglichkeit, die Entscheidung im Sinne von Divide and Conquer in Teilentscheidungen zerlegen. Für

Abb. 17.1 Thematische Einordnung des Kap. 17



jede dieser Teilentscheidungen kann die jeweils geeignete Umsetzung gewählt werden. Unsere Erfahrung zeigt, dass hybride Intelligenz als Interaktion von maschinellem Lernen, explizit repräsentiertem Wissen und Menschen es ermöglicht, mit begrenzten und unstrukturierten Daten in komplexen Bereichen und bei wechselnden Kriterien konsistente Entscheidungen zu treffen.

Abb. 17.1 zeigt die Einordnung der Lösung in die im Einleitungskapitel vorgestellten Kategorien. Die Anwendung ist eine Entscheidung über die Zulassung zu einem Studiengang an einer Hochschule, was hier als öffentlicher Dienst klassifiziert ist. Als Methoden kommen regelbasierte Systeme und maschinelles Lernen zum Einsatz, die Daten aus Anmeldeunterlagen und einer Tabelle mit den Daten von Anmeldungen der letzten Jahre beziehen.

17.2 Anwendungsfall

Jedes Jahr erhält die Fachhochschule Nordwestschweiz (FHNW) etwa 150 Anmeldungen für den Masterstudiengang Business Information Systems.¹ Im Studienreglement ist das Aufnahmekriterium in den Studiengang sehr einfach definiert: Aufgenommen wird, wer einen Bachelorabschluss in einem verwandten Gebiet mit guten Noten und Berufserfahrung hat. Die Aufnahmekriterien sind in einer Handlungsanleitung konkretisiert. Formal müssen folgende Kriterien erfüllt sein:

- Die Kandidat*innen müssen zwingend einen Bachelor oder einen äquivalenten Abschluss haben.
- Der Abschluss muss von einer anerkannten Hochschule vergeben worden sein.

¹ <https://www.fhnw.ch/de/studium/wirtschaft/msc-bis-business-information-systems>

Sind diese Voraussetzungen erfüllt, werden weitere Kriterien beurteilt:

- Die Durchschnittsnote des Bachelorabschlusses: Erwartet wird eine Durchschnittsnote von mindestens 4,8 laut Schweizer Notensystem. Wurde der Bachelor in einem anderen Land erworben, wird die Note in das Schweizer Notensystem umgerechnet.
- Die Dauer der relevanten Arbeitserfahrung: Erwartet wird mindestens ein halbes Jahr kaufmännische oder technische Berufserfahrung.
- Bachelorabschluss: Erwartet wird ein Bachelor in Wirtschaftsinformatik, Betriebswirtschaft, Informatik oder einem verwandten Gebiet.
- Kommunikationskompetenz in Englisch: Erwartet wird ein Level C1 nach dem gemeinsamen Europäischen Referenzrahmen für Sprachen.²

Diese Kriterien sind nicht binär, sondern können mehr oder weniger gut erfüllt sein. Die Beurteilung, wie gut ein Kriterium erfüllt ist, ist dabei nicht immer eindeutig. Bei der Note gibt es zwar eine klare Maßeinheit zwischen 1 und 6, allerdings ist die Umrechnung ausländischer Noten auf das Schweizerische Notensystem nicht immer eindeutig zu entscheiden, da hinter den Notensystemen unterschiedliche Philosophien stecken. Bei der Arbeitserfahrung beeinflussen neben der Dauer auch die Art der Tätigkeit und der Beschäftigungsgrad die Beurteilung. Da die Bezeichnung von Studienabschlüssen nicht normiert ist und sich bei gleichnamigen Studiengängen die Curricula unterscheiden, ist die Beurteilung der Äquivalenz von Studiengängen eine anspruchsvolle Aufgabe.

Die Note und die Arbeitserfahrung werden nach den folgenden Kategorien bewertet:

- ++ übertroffen,
- + voll erfüllt,
- 0 durchschnittlich,
- nicht erfüllt.

Es wird erwartet, dass die Kandidat*innen in allen Kriterien den Wert + erreichen. Eine 0 kann durch ein ++ in einem anderen Kriterium kompensiert werden. Kandidat*innen, die diese Kriterien erfüllen, können direkt aufgenommen werden. Kandidat*innen, die im Schnitt schlechter sind als + werden nur aufgenommen, wenn noch genügend Studienplätze frei sind.

Das Spektrum der Bachelorstudiengänge, die zum Studium im Master Business Information Systems berechtigen, reicht von wirtschaftlichen bis zu technischen Studiengängen. Neben der Aufnahme in den Studiengang wird daher auch entschieden, ob Studierende Vorleistungen in Form sogenannter Pre-Master erbringen müssen, um fehlende Kompetenzen auszugleichen. Dies erfolgt anhand der Kategorisierung des Bachelorabschlusses.

²<https://www.europaeischer-referenzrahmen.de/>

17.3 Software-Architektur

Das System ist als ausführbares CMMN-Modell des Aufnahmeprozesses (Grudzinska-Kuna 2013; Omg 2013) mit Flowable³ realisiert (Abb. 17.2). Flowable ist ein Workflow-Managementsystem, das es erlaubt, Aufgaben an Personen zu delegieren und externe Anwendungen einzubinden. Die Lösung ist als Prototyp realisiert und mit konkreten Daten getestet.

Die für die Entscheidung über die Aufnahme relevanten Informationen sind in verschiedenen Dokumenten enthalten: ein Formular mit Angaben zur Person, ein Motivationsschreiben, ein Lebenslauf, das Transcript of Record des abgeschlossenen Bachelorstudiengangs und allenfalls Nachweise über Sprachzertifikate oder Zusatzausbildungen. In einem ersten Schritt werden die Anmeldeunterlagen von der Studiengangsassistentin in das System hochgeladen und die relevanten Daten aus den Dokumenten extrahiert und in das System übertragen (Abb. 17.3).

Flowable unterstützt DMN (Decision Model and Notation) der OMG (Omg 2016). Diese Notation wird verwendet, um die Regeln zu automatisieren. Zudem wird ein externer Service zur Extraktion von Informationen aus dem Lebenslauf integriert.

17.4 Umsetzung der Aufnahmeentscheidung

Abb. 17.4 zeigt das Boxology-Modell (van Harmelen und ten Teije 2019) des Aufnahmeprozesses. Nachdem die Daten erfasst sind, werden fünf Teilentscheidungen durchgeführt:

- (a) Überprüfung des Bachelorabschlusses: Es wird geprüft, ob der Bachelorabschluss einem der Studiengänge entspricht, die im Studienreglement als Voraussetzung für die Aufnahme in den Studiengang genannt werden. Diese Prüfung wird durch die Studiengangsassistentin gemacht. Bei anderen Studiengängen analysiert die Studiengangleitung die Äquivalenz anhand des Transcript of Record.
- (b) Anerkennung der Hochschule: Für die Hochschule, die den Bachelorabschluss vergeben hat, wird überprüft, ob sie anerkannt ist. Alle bereits bekannten Studiengänge sind in einem System erfasst, aus dem die Beurteilung automatisch bezogen werden kann.
- (c) Note des Bachelorabschlusses: Die Durchschnittsnote wird, falls aus dem Transcript of Record nicht ersichtlich, bei der Erfassung der Daten berechnet. In diesem Schritt werden ausländische Noten in das Schweizer Notensystem umgerechnet und nach ++, +, 0 und – klassifiziert. Hierfür wird auf eine Formel bzw. bestehende Tabellen zurückgegriffen.

³<https://www.flowable.com/>

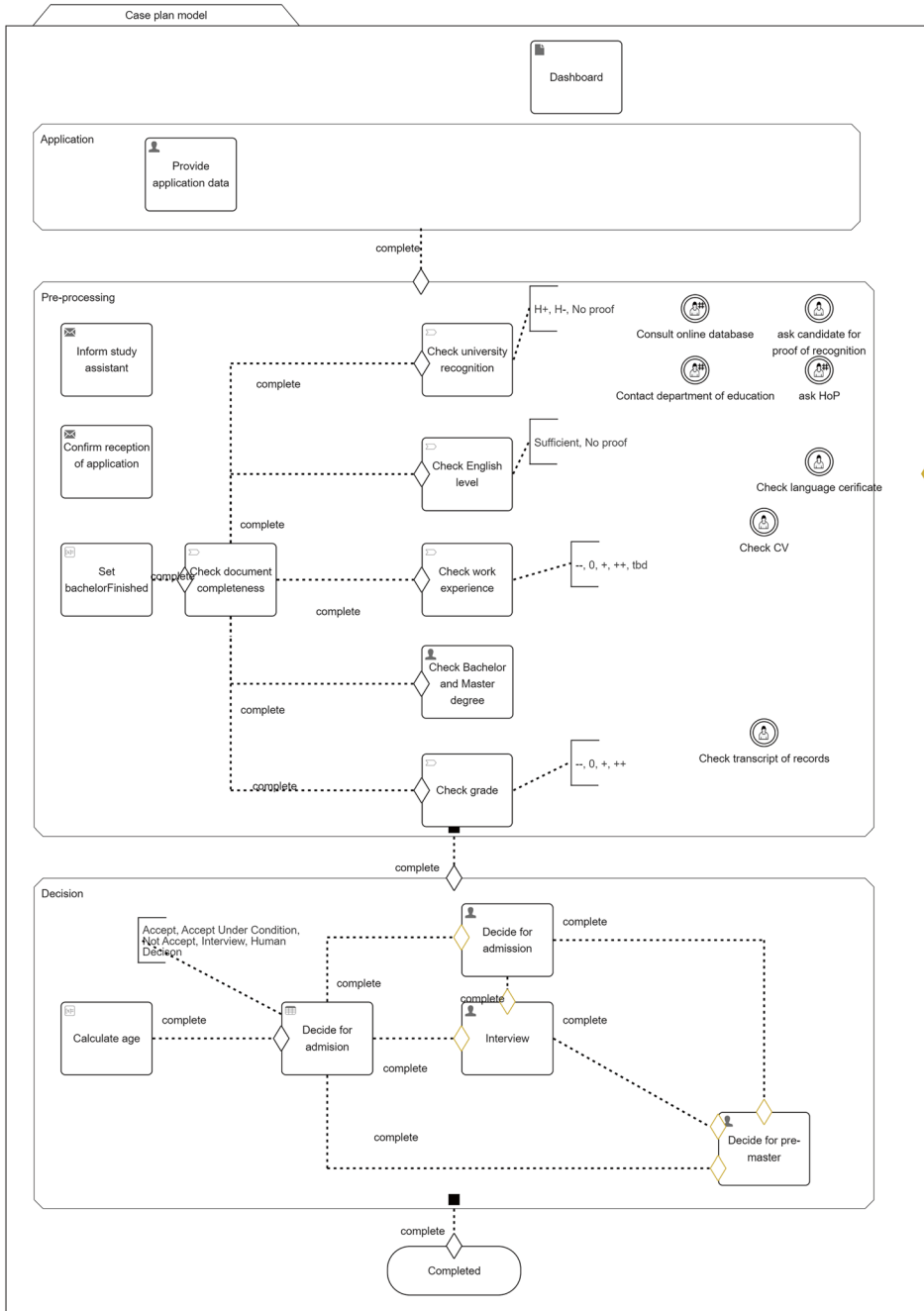


Abb. 17.2 CMMN-Modell des Aufnahmeprozesses

Task Comments People Subtasks Documents History

PERSONAL INFORMATION APPLICATION INFORMATION REQUIRED DOCUMENTS AGB

Person

Title

First name

Last name

Email

Date of birth *

Country of birth

Abb. 17.3 Interface für Antragsdaten

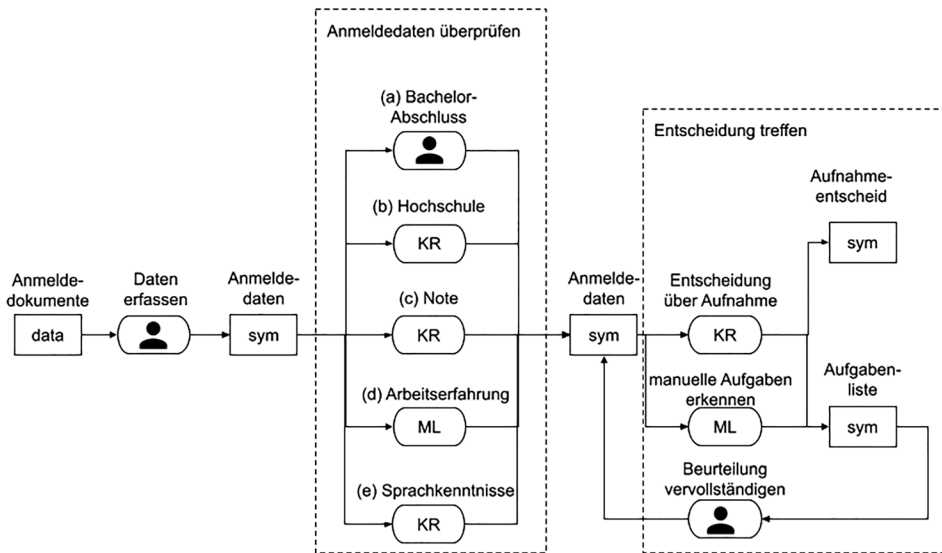


Abb. 17.4 Boxology-Modell des hybriden KI-Systems

(d) Arbeitserfahrung: Aus dem Lebenslauf werden die Art und Dauer der relevanten Arbeitserfahrung extrahiert und nach ++, +, 0 und – klassifiziert. Für die Extraktion wird ein auf maschinellem Lernen basierendes System eingesetzt.

- (e) Sprachkenntnisse: Bei der Übertragung der Daten aus den Anmeldeunterlagen werden auch die vorhandenen Sprachzertifikate eingetragen. Diese werden dann anhand einer Entscheidungstabelle in den gemeinsamen Europäischen Referenzrahmen für Sprachen umgewandelt.

Nachdem die Anmeldeinformationen mit den Informationen aus den einzelnen Teilergebnissen ergänzt wurden, wird der Aufnahmeentscheid getroffen. Dafür wird ein Regelsystem eingesetzt, das als Entscheidungstabelle realisiert ist. Sind die relevanten Informationen für die Entscheidung vorhanden, wird die Entscheidung automatisch getroffen. Fehlen Informationen, wird eine Aufgabe in die Aufgabenliste eingetragen, sodass eine Person die fehlenden Beurteilungen vornehmen kann.

Ein Ziel ist es, einerseits einen möglichst großen Teil des Aufnahmeentscheids zu automatisieren und nur in unklaren Fällen einen Menschen zu involvieren. Andererseits soll auch sichergestellt sein, dass Entscheidungen transparent und nachvollziehbar sind. Daher haben wir mit maschinellem Lernen experimentiert, um zu erkennen, ob eine menschliche Intervention notwendig ist. Dieses gibt neben dem Regelsystem zusätzliche Hinweise für die verantwortlichen Personen, einen Fall näher zu betrachten. Die einzelnen Teilschritte werden in den folgenden Abschnitten beschrieben.

17.4.1 Anerkannte Hochschule

Der Entscheid über die Anerkennung des Bachelorabschlusses wird nicht von der FHNW, sondern von staatlichen Stellen getroffen. Daher ist es die Aufgabe der Studiengangassistenten in vertrauenswürdigen Online-Quellen zu recherchieren, ob die Hochschule, an der der Bachelorabschluss erworben wurde, offiziell anerkannt ist.

Dazu verwendet sie Anabin,⁴ das Infoportal zu ausländischen Bildungsabschlüssen der deutschen Kultusministerkonferenz. Die Klassifikationen H+ und H- kennzeichnen anerkannte bzw. nicht anerkannte Hochschulen. Bei Hochschulen der Kategorie H+/- oder wenn die Hochschule nicht in Anabin enthalten ist, muss genauer recherchiert werden.

Es gibt keine einfachen Regeln, um die Überprüfung zu automatisieren. Auch hilft hier maschinelles Lernen nicht weiter, da man von der (Nicht-)Anerkennung einer Hochschule nicht auf die (Nicht-)Anerkennung einer anderen Hochschule schließen kann. Auch ist die Anerkennung nicht von anderen Daten der Anmeldungen abhängig. Allerdings kann man den Aufwand reduzieren, indem man das Ergebnis der Recherche in Fällen, die in Anabin nicht enthalten sind oder als H+/- klassifiziert sind, in einer Fallbasis speichert. Für zukünftige Fälle, für die Anabin keine Lösung liefert, kann man automatisch nachprüfen, ob die Hochschule in der Fallbasis enthalten ist. Es handelt sich dabei um einen Spezialfall von fallbasiertem Schließen mit einem einfachen Ähnlichkeitsmaß, das lediglich Varianten der Schreibweise berücksichtigt.

⁴<https://anabin.kmk.org>

17.4.2 Note des Bachelorabschlusses

Kandidat*innen mit einer mindestens guten Note werden direkt aufgenommen. Grundlage für die Bewertung ist das Schweizer Notensystem, in dem "gut" der Note 5 entspricht. Die Durchschnittsnote wird auf eine Nachkommastelle gerundet. Eine gute Note entspricht dem Rating +, "sehr gut" oder "exzellent" entsprechen ++, eine befriedigende oder ausreichende Note entspricht einer 0. Ungenügende Noten können eigentlich nicht vorkommen, da man mit einem Durchschnitt unter 4 in der Regel keinen Abschluss bekommen würde. Tab. 17.1 zeigt die Kriterien für das Rating der Note.

Die Noten des Bachelorabschlusses sind im Transcript of Records zu finden. Ist die Durchschnittsnote nicht angegeben, wird sie berechnet. Noten einer ausländischen Hochschule werden in das Schweizer Notensystem umgewandelt, wenn sich die Universität, die den Bachelorabschluss vergeben hat, nicht in der Schweiz befindet.

Die Umrechnung in das Schweizer Notensystem ist der schwierigste Teil. Es gibt verschiedene Methoden für diese Umrechnung, wobei eine häufig angewandte Methode die sogenannte bayerische Formel ist. Diese kann an das Schweizer Notensystem angepasst werden. Sie ist für numerische Notensysteme anwendbar und erfordert neben der Eingabe der umzurechnenden Note die Höchstnote des fremden Notensystems und die niedrigste Note, die zum Bestehen ausreicht. Da jedoch die Höchstnote und die niedrigste Bestehensnote benötigt werden, diese aber nicht immer im Transcript of Records enthalten sind, erfordert dies Kenntnisse über die Notenskala der jeweiligen Universität. Ist die Information nicht verfügbar, recherchiert die Studiengangassistentin online. Ein weiterer Nachteil der bayerischen Formel ist, dass sie auf der Annahme beruht, dass die Notenverteilung linear ist, was nicht unbedingt der Fall ist.

17.4.3 Arbeitserfahrung

Die Beurteilung der Arbeitserfahrung basiert auf Informationen im Lebenslauf. Die Beurteilung der Arbeitserfahrung besteht aus zwei Schritten:

- relevante Berufserfahrung aus dem Lebenslauf extrahieren,
- die Relevanz und Dauer der Arbeitserfahrung beurteilen und als ++, +, 0 oder – klassifizieren (Tab. 17.2).

Tab. 17.1 Rating der Bachelorabschlussnoten

Note n	$n > 5,2$	$5,2 \leq n \leq 4,8$	$4,8 < n \leq 4$	$n < 4$
Rating	++	+	0	–

Tab. 17.2 Klassifizierung von Berufserfahrung

Dauer der Arbeitserfahrung	$d > 2 J$	$2 J \leq d \leq 6M$	$6M < d \leq 1M$	$d < 1M$
Rating	++	+	0	–

Die Beurteilung der Arbeitserfahrung basiert stark auf Erfahrung und Fachwissen, denn die Beschreibung der Tätigkeit ist nicht normiert. Es kommt auch vor, dass Studiengangsleitung und Stellvertretung die Art oder den Umfang der Tätigkeit zunächst unterschiedlich beurteilen.

Die Komplexität der Beurteilung lässt sich reduzieren, indem man zunächst die Tätigkeiten in generische Kategorien gruppiert und anschließend die Relevanz dieser Kategorien beurteilt. Für diese Klassifikation könnte maschinelles Lernen eingesetzt werden. Allerdings ist das Spektrum relevanter Tätigkeiten so groß und die Bezeichnungen sind so vielfältig, dass die Anzahl unserer Beispiele bei Weitem nicht ausreicht, um ein Modell zu trainieren.

Da es jedoch eine Fülle von Online-Diensten zur Analyse von Lebensläufen gibt, ist die Eigenentwicklung eines solchen Algorithmus auch nicht erforderlich. Wir haben uns für den Lebenslauf-Parser von Affinda⁵ entschieden, aber im Grunde genommen könnte jeder Parser für englischsprachige Lebensläufe verwendet werden, der in der Lage ist, die Berufserfahrung in generische Kategorien einzuteilen.

Affinda verwendet die britische Standard Occupation Classification (SOC), um Stellenbezeichnungen in Standardberufsgruppen zu klassifizieren.

Auf Basis der von Affinda erkannten Tätigkeiten und ihrer Dauer hat man eine Basis für die Kategorisierung der Arbeitserfahrung.

Obwohl Affinda nicht alle Arbeitserfahrungen klassifizieren und auch nicht alle Datumsformate erkennen kann, reduziert es den Aufwand erheblich, da man den Lebenslauf nur noch für diejenigen Fälle anschauen muss, für die keine relevante Tätigkeit von mindestens sechs Monaten erkannt wurde.

17.4.4 Sprachkenntnisse

Für diese Entscheidung müssen zwei Aufgaben durchgeführt werden. Erstens sollten Informationen über das Englischniveau eines Bewerbers aus dem Lebenslauf und dem Sprachzeugnis entnommen werden. Zweitens sollte, wenn ein Sprachzeugnis vorliegt, die Note in das CEFR-Niveau umgerechnet werden, um zu entscheiden, ob das Niveau ausreichend ist, wenn nicht, sollten andere englischsprachige Nachweise ausgewertet werden, um das Niveau zu ermitteln.

Bei der Erfassung der Daten werden folgende Informationen zum Dossier hinzugefügt:

⁵<https://www.affinda.com/resume-parser>

- Ist die Bewerberin oder der Bewerber englischer Muttersprachler oder hat einen Bachelorabschluss erworben, der vollständig in englischer Sprache durchgeführt wurde, wird das Englischniveau bereits bei der Datenerfassung auf "ausreichend" gesetzt.
- Liegt ein Zertifikat oder ein Nachweis über Sprachkenntnisse vor, wird die entsprechende Qualifikation in den Anmeldedaten vermerkt.

Im Schritt (e) wird schließlich entschieden, ob das Englischniveau ausreichend ist oder ob ein Mensch diese Entscheidung treffen sollte. Diese Regeln werden mithilfe einer Entscheidungstabelle repräsentiert (Abb. 17.5).

17.4.5 Entscheidung über die Aufnahme

Nach der Beurteilung der Kriterien in den Schritten (a) bis (e) wird über die Aufnahme entschieden. Wenn alle relevanten Informationen verfügbar sind und alle Teilentscheide

F	Certificate / Course Type certificateType string	Grade grade number	Level level string	English Level englishLevelRating string
	-	-	IS IN B2, B2+, C1, C1+, C2, C2+	Sufficient
==	ALTE	-	IS IN level 3, level 4, level 5	Sufficient
==	CISL SF Course	-	IS IN Upper intermediate, Advanced	Sufficient
==	CISL SD	>= 7	-	Sufficient
==	CISL SD Course	-	== Proficient, Advanced, Low advanced, Upper intermediate	Sufficient
==	CISL SD Entrance Score	>= 70	-	Sufficient
==	Cambridge English Scale	>= 160	-	Sufficient
==	BEC Vantage	-	-	Sufficient
==	BEC Higher	-	-	Sufficient
==	CAE	-	-	Sufficient
==	CPE	-	-	Sufficient
==	FCE	-	-	Sufficient
==	GSE	>= 59	-	Sufficient
==	TOEFL iBT	>= 60	-	Sufficient
==	TOEFL iTP	>= 543	-	Sufficient
==	CLB	>= 6	-	Sufficient
==	IELTS	>= 5.5	-	Sufficient
==	TOEIC (L&R)	>= 785	-	Sufficient
==	TOEIC (S&W)	>= 310	-	Sufficient
==	UNicert	-	IS IN II, III, IV	Sufficient
==	PTE Academic	>= 51	-	Sufficient
==	PTE General	-	IS IN level 3, level 4, level 5	Sufficient
==	TEFL	-	-	Sufficient
==	Trinity ISE	-	IS IN II, III, IV	Sufficient
==	Trinity GESE	>= 7	-	Sufficient
==	CAEL	>= 60	-	Sufficient
	-	-	-	Human decision

Abb. 17.5 Beurteilung der Sprachkenntnisse

F	University Recognition		Work Experience		English Level		Grade		Bachelor Degree		Acceptance
	universityRecognitionRating	string	workExperienceRating	string	englishLevelRating	string	gradeRating	string	bachelorDegree	string	acceptance
	IS IN	H-, No proof	-	-	-	-	-	-	-	-	Not Accept
	-	-	==	--	-	-	-	-	-	-	Not Accept
	-	-	-	-	-	-	==	--	-	-	Not Accept
	-	-	!=	++	-	-	==	0	-	-	Not Accept
	-	-	==	0	-	-	!=	++	-	-	Not Accept
	-	-	==	0	-	-	-	-	IS NOT IN	BA, CS, BIS	Human Decision
	-	-	==	tbd	-	-	-	-	-	-	Interview
	-	-	-	-	==	No proof	-	-	-	-	Interview
	-	-	-	-	-	-	-	-	-	-	Interview
	-	-	-	-	-	-	-	-	-	-	Interview
	-	-	-	-	-	-	-	-	-	-	Interview
	-	-	-	-	-	-	-	-	-	-	Accept Under Condition
	-	-	-	-	-	-	-	-	-	-	Accept

Abb. 17.6 Entscheidungstabelle für Aufnahmeentscheidung (Ausschnitt)

automatisch getroffen werden konnten, wird der Aufnahmeentscheid abgeleitet. Wenn einzelne Kriterien nicht geklärt werden könnten, wird eine Person involviert.

Dieser Schritt kann mithilfe von Regeln realisiert werden, die als Entscheidungstabelle implementiert werden. Abb. 17.6 zeigt einen Ausschnitt der Entscheidungstabelle, bei der Spalten und Zeilen ausgeblendet sind. Es wird die erste Regel angewendet, deren Bedingung erfüllt ist. Dies entspricht der Hit-Policy "First" (Omg 2016).

Eine Anmeldung wird abgelehnt, sobald ein Kriterium nicht erfüllt ist. Dementersprechend werden zuerst diejenigen Regeln, die zu einer Ablehnung führen, ausgeführt:

- Kandidat*innen müssen einen Bachelorabschluss haben,
- die Hochschule, die den Bachelorabschluss vergeben hat, hat die Anerkennung H–,
- Arbeitserfahrung oder Note sind mit – eingestuft,
- die Note ist mit 0 eingestuft, aber nicht durch ++ bei der Arbeitserfahrung kompensiert,
- die Arbeitserfahrung ist mit 0 eingestuft, aber nicht durch ++ bei der Note kompensiert,
- der Bachelorabschluss ist nicht äquivalent zu Betriebswirtschaftslehre, Informatik oder Wirtschaftsinformatik (Subject Mismatch) und die Berufserfahrung ist 0.⁶

Danach werden die Fälle geprüft, bei denen weitere Abklärungen zu treffen sind. Das ist dann der Fall, wenn die Beurteilungen in den Schritten (a) bis (e) nicht automatisch durchgeführt werden konnten. Als Ergebnis dieser Regeln wird ein Auftrag an die verantwortliche Person abgeleitet, die in die Aufgabenliste eingetragen wird. Einige Beispiele dafür könnten sein:

⁶Das heißt, ein Subject Mismatch kann durch einschlägige Berufserfahrung ausgeglichen werden.

- Hat der Kandidat bzw. die Kandidatin einen anderen Bachelorabschluss als Betriebswirtschaftslehre, Informatik oder Wirtschaftsinformatik, liegt eine fachliche Diskrepanz vor. In diesem Fall muss geprüft werden, ob der Mismatch auf andere Art ausgeglichen wurde, z. B. Weiterbildungen oder Master in einem verwandten Gebiet.
- Konnte die Arbeitserfahrung oder die Sprachkenntnisse nicht beurteilt werden, wird ein Aufnahmegespräch geführt.

Sollte keine der Regeln zur Ablehnung oder zusätzlichen Beurteilung greifen, kann davon ausgegangen werden, dass alle Kriterien erfüllt sind und der Kandidat bzw. die Kandidatin aufgenommen werden kann.

17.4.6 Überprüfung von Entscheidungen

Um zu erkennen, welche Entscheide von der Studiengangsleitung geprüft werden müssen, haben wir mit maschinellem Lernen experimentiert. Dazu wurden Daten sowohl von abgelehnten als auch von aufgenommenen Bewerbern verwendet, die im Zeitraum von Februar 2022 bis September 2023 (also über drei Semester) gesammelt wurden und insgesamt 321 Bewerbungen umfassten. Für jede Bewerbung wurden in unseren Experimenten folgende Attribute berücksichtigt: (1) *gewünschtes Studienmodell (Vollzeit oder Teilzeit)*, (2) *Bachelorabschluss (z. B. Business Administration, Computer Science)*, (3) *Name der Universität, die den Bachelorabschluss verliehen hat*, (4) *Land dieser Universität*, (5) *Abschlussnote*, (6) *ein Attribut, das angibt, ob die Note eindeutig umgerechnet werden konnte oder nicht*, (7) *Berufserfahrung in Monaten*, (8) *ein Attribut, das angibt, ob die Berufserfahrung eindeutig berechnet werden konnte oder nicht*, (9) *Art des Englischzertifikats*, (10) *Staatsangehörigkeit*, (11) *Wohnort*, (12) *ein Attribut, das angibt, ob ein Visum benötigt wird*, und (13) *eine optionale Bemerkung, die teilweise von der Assistentin erfasst wurde*.

Da wir die Rolle des maschinellen Lernens in diesem Prozess darin sehen zu entscheiden, welche Bewerbungen nicht vom Studiengangsleitenden und seinem Stellvertreter geprüft werden müssen, haben wir ein Klassenattribut für die gegebenen Daten konstruiert, das auf dem Vorhandensein eines vom Studiengangsleitenden oder seinem Stellvertreter eingefügten Kommentars basiert. Ein Kommentar wird in der Regel nur erfasst, wenn eine Bewerbung diskutiert werden muss oder eine Besonderheit aufweist. Daher ist die Vorhersage des Vorhandenseins eines Kommentars gleichbedeutend mit der Vorhersage, ob eine Bewerbung geprüft werden muss. Bei der Auswahl der Attribute (siehe oben) haben wir zunächst auch solche mit eingeschlossen, deren Vorhersagekraft zweifelhaft scheint (beispielsweise das gewünschte Studienmodell), da sich solche Vorhersagekraft manchmal auch überraschend herausstellen kann. Die meisten Attribute schätzten wir aber als hilfreich ein – insbesondere auch die Attribute (6) und (8), welche auf Unsicherheiten hinweisen, die von der Studiengangsleitung nochmals angesehen und beurteilt werden sollten.

Anschließend führten wir Experimente mit verschiedenen Algorithmen für maschinelles Lernen in der Weka Workbench (Frank et al. 2010) durch. Wir verwendeten einige menschenlesbare Modelle, um die den Vorhersagen zugrunde liegenden Muster zu verstehen, und setzten auch einige nicht interpretierbare Algorithmen ein, um ein Verständnis für die Leistungsgrenzen des Ansatzes zu gewinnen. Darüber hinaus arbeiteten wir mit kostensensitiver Klassifizierung (Domingos 1999), um die Klassifikatoren für Fälle zu sensibilisieren, die eine Überprüfung verdienen – wobei wir eine falsch-negative Vorhersage zehnmal stärker bestrafen als eine falsch-positive. Unsere Ergebnisse lassen sich wie folgt zusammenfassen:

- Mit einem **Random-Forest-Modell** (unserem besten Kandidaten) erreichen wir eine Genauigkeit (Accuracy) von 81 %, die Area Under the Curve (AUC) liegt bei 0,72. Betrachtet man die gekennzeichneten Bewerbungen, so findet das Modell nur 11 von 68 diskussionswürdigen Fällen, d. h., es hat einen Recall von 16 %. Dies wäre in der Praxis nicht sehr hilfreich.
- Durch die Anwendung des **kostensensitiven Lernens** erhöht sich die Trefferquote auf 78 % (53 von 68 diskussionswürdigen Fällen werden identifiziert), die Genauigkeit sinkt jedoch von 73 % auf 34 %, d. h., es gibt 101 "Fehlalarme". Selbst wenn man für dieses Verfahren einen Recall von nur etwa 80 % akzeptiert, ist unklar, ob ein Modell mit dieser Performance von praktischem Wert wäre.
- Durch die manuelle Überprüfung einiger falsch-negativer Fälle, die auch bei kostensensitivem Lernen bestehen blieben, konnten wir jedoch einige **zusätzliche Attribute** identifizieren, die man aus den Bewerbungsunterlagen extrahieren könnte mit guten Chancen, weitere diskussionswürdige Bewerbungen zu identifizieren. Diese Attribute umfassen die Information, ob ein Bewerber bereits für einen Master of Science in der Schweiz immatrikuliert war, und die Anzahl der Jahre seit dem Bachelorabschluss.
- Bei der Analyse von falsch-positiven Vorhersagen zeigte sich, dass insbesondere die Attribute „Note“ und „Visum“ zu solch falschen Entscheiden geführt haben. Bei der Note hat sich die Handhabung im betrachteten Zeitraum leicht verändert, was diese Schwierigkeiten teilweise erklären kann (wenn mehr Trainingsdaten ab 2023 vorliegen, sollten daher nur noch diese verwendet werden). Wenn ein Visum benötigt wird, führt dies häufig zu einem Interview, um das Vorhandensein des Visums zu klären. Dies ist aber nicht immer der Fall und die entsprechenden Abweichungen scheinen recht schwer vorherzusagen zu sein.
- Bei der Untersuchung **menschenlesbarer Modelle** war einer der Hauptindikatoren für die Kennzeichnung einer Bewerbung das Vorhandensein des Wortes "bezahlt" im Bemerkungsfeld. Entfernt man das entsprechende Attribut (das nicht viel Sinnvolles beiträgt), so stellt man fest, dass weitere Einträge markiert werden, wenn die Berufserfahrung entweder zu niedrig oder zu hoch⁷ ist, wenn ein Visum benötigt wird oder

⁷Bei Bewerbern mit sehr viel Berufserfahrung wird abgeklärt, ob nicht eher eine Weiterbildung statt einen Master-Studiengangs passend wäre.

wenn die Note unter einem bestimmten Schwellenwert liegt. All diese Kriterien sind intuitiv sinnvoll, da sie oft zu weiteren Nachforschungen oder Rückfragen führen. Sie setzen jedoch voraus, dass eine Bewertung der Berufserfahrung oder der Note bereits stattgefunden hat.

Die **Qualität der Vorhersagen** ist bei allen Verfahren eher gering. Man könnte sie als zusätzlichen Hinweis verwenden, um das Risiko zu reduzieren, dass ein Fall automatisch entschieden wird, für den eine tiefere Analyse sinnvoll wäre.

Als Alternative zum maschinellen Lernen haben wir auch mit der Nutzung von Ontologien und Technologien des Semantic Web experimentiert (Abb. 17.7).

Die Nutzung von Ontologien hat den Vorteil, dass Regeln, die auf Klassendefinitionen basieren, ausdrucksstärker sind als Regeln in Entscheidungstabellen. Das kann man zum Beispiel bei der Beurteilung relevanter Arbeitserfahrung und der Äquivalenz von Studiengängen einsetzen, indem man die Themen in einer Ontologie abbildet und Regeln definiert, die das Vorkommen von relevanten Themen im Curriculum überprüfen. Als nächster Schritt soll die Ontologie in das System eingebunden werden, um zu schauen, ob die Vorteile sich in der konkreten Anwendung tatsächlich als nützlich erweisen.

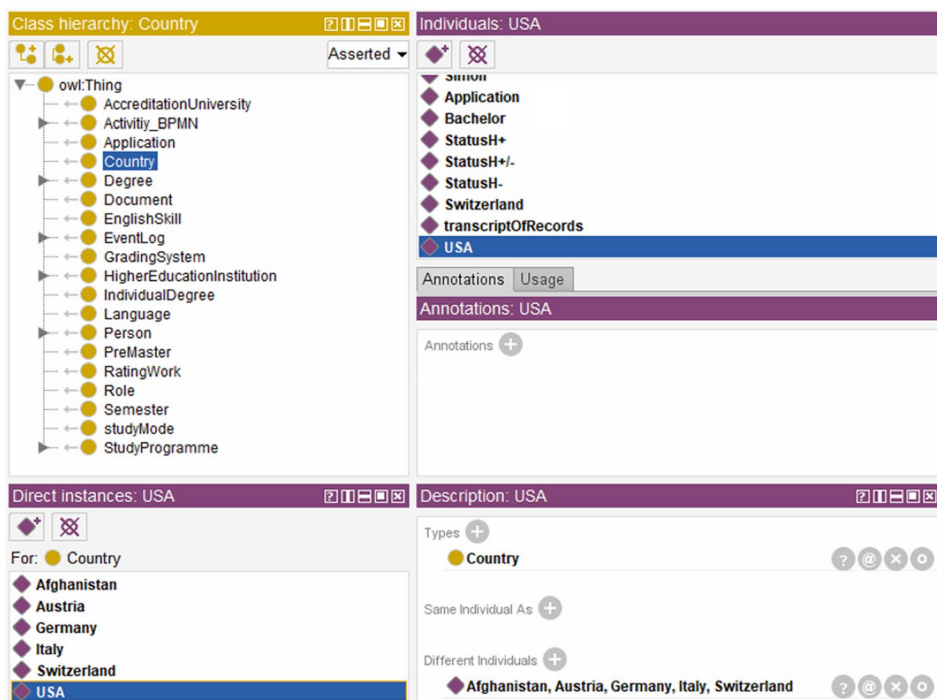


Abb. 17.7 Ausschnitt aus der Ontologie zum Aufnahmeprozess (Eichele 2022)

17.5 Empfehlungen

Für zukünftige, ähnliche Projekte möchten wir folgende Empfehlungen aussprechen:

- Die Aufgabe des KI-Systems besteht darin, zu entscheiden, wann ein Mensch eingebunden werden sollte.
- Um automatisch abschließend zu entscheiden, ob Kandidaten*innen für ein Studium aufgenommen werden sollten, eignen sich Regeln am besten. Maschinelles Lernen ist für diesen Zweck nicht optimal geeignet.
- Menschenlesbare Modelle des maschinellen Lernens können jedoch dabei unterstützen, Aspekte in einer Bewerbung zu erkennen, die besonderer Aufmerksamkeit bedürfen, wie zum Beispiel fehlende Arbeitserfahrung.
- Maschinelles Lernen, insbesondere Large Language Models, zeigt Potenzial bei der Extraktion von Attributwerten (z. B. aus einem Lebenslauf).
- Um Regeln elegant zu erfassen und transparent auszuwerten, empfiehlt es sich, das Domain-Wissen in einer Ontologie abzubilden.

17.6 Fazit

Diese Arbeit beschreibt eine Anwendung für hybride Intelligenz, die auf der Interaktion von Mensch und KI basiert. Die Methoden der KI dienen als Werkzeuge für die Vorbereitung einer Entscheidung. So kann die Beurteilung, ob eine Arbeitserfahrung für das Studium relevant ist, durch KI unterstützt, indem man die Tätigkeiten automatisch klassifiziert und die Klassen nach ihrer Relevanz für den Studiengang bewerten kann. Dies führt zu einer spürbaren Erleichterung der Aufgabe der Studiengangassistenz. Allerdings kann der Mensch nicht vollständig ersetzt werden. Einerseits basieren Teile der Beurteilungen auf Einschätzungen und Erfahrungen, die nicht maschinell gelernt werden können, weil nicht ausreichend Daten vorhanden sind, um alle potenziellen Situationen abzudecken. Ein Beispiel hierfür ist die Umrechnung der Noten. Diese könnten prinzipiell gelernt werden, wozu aber genügend Beispieldaten für jedes Notensystem benötigt würden. Für diese Umrechnung kann man Regeln angeben. Schließlich gibt es Aspekte, deren Umsetzung aus Gründen des Aufwands nicht sinnvoll ist. Ein Beispiel hierfür ist die Überprüfung, ob eine Hochschule anerkannt ist. Es wäre prinzipiell möglich, diese Informationen zu sammeln und auch automatisch zugänglich zu machen. Allerdings können, wie Anabin zeigt, die Vollständigkeit und Korrektheit nicht sichergestellt werden, da es hier immer wieder zu Änderungen kommt. Aus diesen Gründen ist es notwendig, Menschen zu involvieren. Eine Aufgabe für das KI-System ist es dabei, zu erkennen, in welchen Situationen dies geschehen soll. Auch dies wurde in dem beschriebenen Vorgehen realisiert, sowohl durch explizite Regeln als auch unter Nutzung von maschinellem Lernen.

Das System wurde als Workflow mit dem System Flowable umgesetzt, welches es erlaubt, Aufgaben sowohl an Menschen zu delegieren als auch zu automatisieren. Dies hat die Auswahl der Methoden beeinflusst. Insbesondere wurden die Regelkomponenten durch DMN-Entscheidungstabellen umgesetzt, die im Workflow-System bereits integriert sind.

Literatur

- Dagar, A. (2021). A comparative study on loan eligibility. *International Journal of Scientific Research & Engineering Trends*, 7(3), 1646–1649.
- Domingos, P. (1999). MetaCost: a general method for making classifiers cost-sensitive. *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 155–164.
- Eichele, S. (2022). *Enhancement of Event Logs with Domain Knowledge* (K. Hinkelmann & M. Spahic (eds.)) [MSc Business Information Systems]. FHNW University of Applied Sciences and Arts Northwestern Switzerland.
- Frank, E., Hall, M., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I. H. & Trigg, L. (2010). Weka – A Machine Learning Workbench for Data Mining. In O. Maimon & L. Rokach (Eds.), *Data Mining and Knowledge Discovery Handbook* (pp. 1269–1277). Springer US.
- Grudzinska-Kuna, A. (2013). Supporting Knowledge Workers: Case Management Model and Notation (CMMN). *Information Systems in Management*, 2(1), 3–11.
- Hänig, C., Schlösser, M., Hamotskyi, S., Zambaku, G. & Blankenburg, J. (2023). NLP-based Decision Support System for Examination of Eligibility Criteria from Securities Prospectuses at the German Central Bank. In *arXiv [cs.CL]*. arXiv. <http://arxiv.org/abs/2302.04562>
- Hollenweger, J. (2011). Development of an ICF-based eligibility procedure for education in Switzerland. *BMC Public Health*, 11 Suppl 4(Suppl 4), S7.
- Omg. (2013). *Case Management Model and Notation (CMMN), Version 1.0*. Object Management Group OMG.
- Omg. (2016). *Decision Model and Notation (DMN) V1.1*. Object Management Group OMG.
- Sarkar, A. (2021). Machine learning techniques for recognizing the loan eligibility. *International Research Journal of Modernization in Engineering Technology and Science*, 3(12). https://www.irjmets.com/uploadedfiles/paper/volume_3/issue_12_december_2021/17758/final/fin_irjmets1641311178.pdf
- van Harmelen, F. & ten Teije, A. (2019). A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems. In *arXiv [cs.AI]*. arXiv. <http://arxiv.org/abs/1905.12389>

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Kapitel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.



Stichwortverzeichnis

1D-Multistep-Ahead-CNN, 163, 165
3-Schichten-Architektur, 186

A

Abkürzung, 143
ABox, 326
Accuracy, 414
Additives Modell, 285
Ähnlichkeitsgraph, 136, 148
Anomalie, 246
Anomaliedetektion, 246
Anonymisierung, 224
Antonym, 109, 143
Aortenklappenstenose, 200, 201
Apache Kafka, 236
A-priori-Verteilung, 274
A-priori-Wahrscheinlichkeit, 210
A-priori-Wert, 273
Area Under the Curve (AUC), 414
Autoencoder, 131, 153, 158, 162–165
Automat, 38
Automated Machine Learning (AutoML), 176
Automation and Measuring System, 324
Autonomes Fahren, 320, 343
Autonomes Fahrsystem, 319, 320, 322, 326, 341
Autonomous Driving (AD), 320

B

Bag-of-Words, 130
Baumstruktur, 229
Bayes'sches Verfahren, 271

Bayes'sche Modellierung, 271
Bayes'sches Modell, 210
 Hierarchisches Bayes'sches Modell, 282
Bayes'sches Netz, 200, 209
Behauptungsbox, 325
Beschreibungslogik, 30, 324
Blackboard-Architektur, 170, 185
Blackbox, 158, 160
boole'sche Suche, 142
Boxology, 12
Boxology-Modell, 405

C

Case-based Reasoning (CBR), 298
CBOW, 132, 145, 147
Chatbot, 35
ChatGPT, 35
Cloud, 35
Clustering, 127, 363
Clusteringverfahren, 135
CMMN, 405
Codenummer, 227
Community-basiertes Graph-Clustering, 135
Concept Embedding, 233
Controlled Natural Language (CNL), 38
Controller Area Network, 325
Conversational AI, 35
Convolutional Neural Network (CNN), 153, 158, 161
Coreference Resolution, 33
Cyber-physisches
 Produktionssystem, 243, 244
Cyber-physisches System, 244

D

Data Dictionary, 218
 Datengesteuertes Lernen, 333
 Datenkomprimierung, 153
 Datenpunkt, 158
 Datenschutzgrundverordnung (DSGVO), 225
 Datenverfügbarkeit, 157
 DBSCAN, 135, 139
 Decoding, 164
 Deduktive Inferenz, 14
 Deep Learning, 250, 320, 325, 349
 Deep-Learning-Modell, 322, 326
 Deep-Learning-Modul, 322
 Denkmalschutz, 48
 deterministische Variable, 277
 Digital Farming, 154
 Digitalisierung, 227
 Diskrete Wahrscheinlichkeitsverteilung, 280
 Distant Supervision, 34, 35
 Distanz-Ähnlichkeit, 149
 DMN, 405
 Docker, 236
 Down-Sampling, 212
 Dynamisches Bayes'sches Modell, 215

E

Einflussstärke, 212
 ElasticSearch, 133
 EM, 200
 Encoding, 164
 Enrichment, 356
 Entscheidungsbaum, 338
 Entscheidungsunterstützung, 298
 Erdbeobachtungsdaten, 153
 Ertragskarte, 160
 Ertrags- und Wachstumsprognose, 158
 Ertrags- und Wachstumsvorhersage, 154, 156
 Ertragsvorhersage, 156–158
 Erwartungswert, 280
 ETL-Prozess, 28
 euklidischen Distanz, 139
 Event Detection, 33
 Expectation-Maximization-Algorithmus, 200
 Expertensystem, 156
 Expertenwissen, 158, 165, 297, 328

Exponentialfunktion, 280
 Extreme Gradient Boosting (XGBoost), 160

F

facettierte Suche, 49
 Fallbasiertes Schließen, 298
 False Negative, 210
 Faltungsnetz, 158, 163
 Fernerkundung, 156, 157
 Few-shot Learning, 331
 Fine-Tuning, 331
 FOL, 322, 324, 336, 339
 FOL-Formel, 340
 FOL-Regel, 339
 Formale Logik, 331
 Formales Logiksystem, 330
 Frage-Antwort-System, 35
 fremdsprachliche Übersetzung, 143
 Frequentistische Statistik, 273

G

Gauss'sche Multivariate Mixture Modell, 135
 Gauß-Verteilung, 291
 Generatives neuronales Netz (GNN), 3
 GeNIe Modeller, 210
 Gensim, 147
 gerichteter azyklischer Graph (DAG), 143
 Gerichtetes grafisches Modell, 271, 272, 276
 Gewichtetes lineares Modell, 289
 Gradient Boosting, 160
 Gradient Boosting Decision Tree (GBTD), 160
 graph-basiertes Clustering, 135
 GraphDB, 235
 Graph Neural Network, 247
 Größe des Hypothesenraums, 208
 Großes Sprachmodell, 320
 Grundatom, 336
 Grundklausel, 338

H

HDBSCAN, 135
 HD-Karte, 330, 333
 Hidden-Markov-Modell, 288

Hintergrundwissen, 249
Hit-Policy, 412
hochdimensionalen Vektorraum, 130
Hochdimensionaler Datenraum, 208
Homonym, 131
Horn-Klausel, 335, 336
Human in the Loop, 18
hybride semantische Suchfunktion, 128
Hybrides Modell, 333
Hyperonym, 79
Hypothese, 337
Hypothesenraum, 208

I

ICD (International Classification of Diseases), 227
IE, 33
Inductive Logic Programming (ILP), 338
 Neuronales ILP, 342
Induktive Inferenz, 14
Induktives logisches Programmieren, 320
Inferenzalgorithmus, 30
Information Extraction (IE), 33
 Open Information Extraction, 34
Information Retrieval, 349
Informationsextraktionsverfahren, 28
Informationsgehalt, 137
Interpolation, 158, 165
inverse Dokumentenfrequenz (IDF), 147

K

KG, 28, 33, 35, 38, 335
Klappenprothese, 202
Klassifikationssystem, 127
K-nearest Neighbor (kNN), 229
K-Nearest-Neighbour (kNN), 133
Knowledge Engineering, 28
Knowledge Graph (KG), 26
Knowledge Graph, 126
Knowledge Graph Completion, 81
Knowledge Graphen, 143
Kompositum, 107
Konfidenzintervall, 215, 279
Konfidenzwert, 37

Kontextfreie Grammatik, 40
Kontrollierte Sprache, 38
Konzept, 30
Konzeptnormalisierung, 225
Korpus, 233
Kosinus-Ähnlichkeit, 132, 139
Kostensensitives Lernen, 414
Kreuzvalidierung, 212
Künstliche Intelligenz (KI), 170, 376
 Hybride KI, 5
Künstliche Intelligenz, 244, 298

L

Landwirtschaftliche Daten, 153
Langfristige Planung, 334
Language, 324
Large Language Model (LLM), 68, 331
Large Language Model, 28
Latente Variable, 215, 277, 280
Latente Zufallsvariable, 212, 279
Lemmatisierung, 140
Lexem, 86
Lexikalische Ambiguität, 102
LightGBM, 160
Light-Gradient Boosting Machines (LightGBM), 160
Lineare temporale Logik (LTL), 328
Logikprogramm, 335, 336, 339
Logikregel, 322
Logistic Regression, 235
Logistische Funktion, 277, 291
Lokales System, 35
Long-short Term Memory (LSTM), 161
Louvain, 135, 147
LSTM, 131

M

Machine Learning (ML), 170, 224
Machine Learning, 250
Machine Learning Workflow, 175
manuellen Modellierung, 128
Mapping, 30
Mapping-Regel, 325
Markov Chain Monte Carlo (MCMC), 290

- Markov-Clustering (MCL), 135
 Markov-Entscheidungsprozess, 334
 Markov-Kette, 288
 Maschinelles Lernen (ML), 376
 selbstüberwachtes Lernen, 35
 überwachtes Lernen, 34
 unüberwachtes Lernen, 135
 Maschinelles Lernmodell, 320
 MCL, 135
 Medical Subject Headings (MeSH), 229
 Medizincontroller, 227
 Mehrheitsregel, 235
 Mehrwortlexem, 102
 Menschenlesbares Modell, 414
 Merkmalsraum, 226
 Meta Automated Machine Learning (Meta AutoML), 179
 Metrische temporale Logik (MTL), 328
 Metrische temporale Logik, 332
 Modus-Deklaration, 337–339, 342
 Multimodale Knowledge Graph Completion (OW-KGC), 78
 Multiplikatives Modell, 285
 Multistep-Ahead, 163
- N**
- Named Entity Recognition (NER), 33, 78
 Named Entity Recognition, 33, 354
 Namend-Entity-Recognition, 142
 Natural Language Interface, 38
 Natural Language Processing (NLP), 69, 224
 Neuralbasierte ILP-Architektur, 341
 Neural Network, 254
 Neuronaler ILP-Ansatz, 341
 Neuronaler induktiver Regellernansatz, 338
 Neuronales induktives Regellernen, 340
 Neuronales Netz, 2, 158, 163, 323
 NLP, 147
 NLP-Framework, 34
 Noisy-Max-Knoten, 212, 215
 nominale Variable, 285
 Nominalphrasen, 143
 Notrufabfrage, 297
- O**
- Oberbegriff, 126
 Okapi BM25, 357
 Online-Learning, 207
 Ontologie, 3, 28, 49, 126, 143, 145, 155, 164, 181, 230, 323–326, 342, 349, 415
 Operational Design Domain, 337
 OPS (Operationen- und Prozedurenschlüssel), 227
 OPTICS, 135, 140
 OWL, 30
- P**
- Parameterabhängigkeit, 209
 Parser, 410
 Part-of-Speech-Tagger, 40
 Patientenakte, 227
 Planungsmodell, 343
 Poisson-Variable, 282
 Poisson-Verteilung, 280
 POS-Tagger, 40
 Prädikat, 327
 Prädikatenlogik, 336
 Prädikatenlogik erster Stufe, 320
 Prior, 273
 Probabilistische Programmierung, 272, 290
 Probabilistisches grafisches Modell (PGM), 209, 271, 272, 275
 Probabilistisches Programm, 271
 Produktionsregel, 40
 Prompting, 331
 Prozessdaten, 249
 Prozesswissen, 249
 Punktschätzung, 215
 PyMC, 278, 290
 Pyro, 290
 Python, 236
- Q**
- Quantor, 340
 Quasi-Synonym, 140
 Query Expansion, 357
 Question-Answering-System, 372
- R**
- R2RML, 162, 164
 Random Forest, 212, 235
 Random-Forest-Modell, 414
 Randomisierung, 204

Ranking, 134
RDF, 30, 164, 165, 325
RDF-Graph, 40
RDF Knowledge Graph, 164
RDF Triple Language (Turtle), 164
RDF-Triplestore, 32
Realdaten, 158, 161, 163
Reasoner, 30
Recall, 414
Reconciliation, 299
Recurrent Neural Network (RNN), 163
Regelbasierte Diskretisierung, 199
Regelbasierte Methode, 28
Regelbasiertes Schließen, 298
Regelbasiertes System, 320
Regelbasiertes Verfahren, 34
Regel der ersten Ordnung, 339
Regelkonforme Trajektorienvorhersage, 342
Regelsystem, 170, 184, 408
Regelvorlage, 339
RegEx, 37
Regulärer Ausdruck (RegEx), 37
Reinforcement Learning (RL), 334
Rekurrentes neuronales Netz, 160
Rekursives Verfahren, 163
Relation, 30
Relation Extraction (RE), 78
Relation Extraction, 33, 34
Resource Description Framework (RDF), 162, 164
Richtungs-Ähnlichkeit, 133, 139, 149
RL-Modell, 334
RNN, 160, 161
Robustheitsschicht, 333
Rohdaten, 155, 165
Rule-based Reasoning (RBR), 298

S
Satz von Bayes, 209, 271
Scene Graph Generation, 326
Schwach strukturiert, 28
Schwach unstrukturiert, 28
Schwellenwert, 214
Scikit-Learn, 147
Semantic Confidence (SC), 378, 379
Semantic Web, 2, 415
Semantifizierung, 350
Semantik, 30

semantische, 134
semantische Suche, 128
Semantische Technologie, 38
Semistrukturiert, 28
Sensitivitätsanalyse, 212
Sentiment-Analyse, 116
Shallow Parsing, 40
Signal-temporale Logik (STL), 328
SimSet, 130, 134, 137, 140, 142, 145, 147
Simulationsmodell, 159, 160, 165
Simulationssystem, 156, 158, 161, 165
SkipGram, 132, 147
Skolemisierte Formel, 340
Skolem-Normalform, 340
sliding window, 132
Sliding Window, 163, 215
Social-Media-Analyse, 116
Softmax, 284
Soft Voting, 235
Software as a Service (SaaS), 380
Spacy, 147
SPARQL, 32, 38, 164
Spielstärke, 277
Sprachdomäne, 231
Sprachgebrauch, 128
Stammform, 86
Stan, 290
Statistischer Hypothesentest, 273
Statusdaten, 249
Stemming, 140
Strukturelles Wissen, 249
Subject Matter Expert, 356
Support Vector Machine, 235
Support-Vektor-Maschine (SVM), 229
Symbol Grounding, 3
Synonym, 135, 140, 143
Synonyme, 126
Synonymliste, 126
Szenengraph, 327, 328, 342
Szenengraphenerzeugung, 327

T
Taxonomie, 28, 31, 72, 126, 143
Telemedizinische Mitbetreuung, 201, 203
Telemonitoring, 201
Temporale Logik, 320, 328
TensorFlow Probability, 290
Termfrequenz (TF), 147

Terminologie-Service, 33
 Terminologische Box, 325
 terminologisches Wissen, 127
 Textanalyse, 49
 Textbasiertes Dialogsystem, 35
 Textextraktion, 49
 Text-Mining-Modell, 75
 TFIDF, 137
 TF-IDF, 357
 Thesauri, 126
 Thesaurus, 3, 104, 143, 145
 Tokenisiert, 40
 Trainingsdatenbasis, 39
 Trajektorienplanung, 333, 342
 Trajektorienvorhersage, 320, 321, 333, 342, 343
 Trajektorienvorhersagemodul, 334
 Transfer Learning, 156, 158, 161, 163
 Transfer-Lernen, 165, 224
 Transformer-Architektur, 331
 Transformerbasiert, 35
 Transformer-Modul, 340
 Transformer-Netzwerk, 341
 Transkatheter-Aortenklappen-Implantation (TAVI), 200
 Tripel, 30

U

Unified Resource Identifier (URI), 30
 Uniform Resource Identifier (URI), 40
 Unterbegriff, 143
 Unterbegriffe, 126
 Unüberwachtes maschinelles Lernen, 35
 Up-Sampling, 212, 219

V

Varianz, 277, 289, 290
 Variational-Autoencoder, 131
 Variational Inference (VI), 290

Verborgene Variable, 277, 288
 Verifikationssystem, 328
 Verifizieren, 341
 versteckte Bezeichnung, 143
 verwandte Begriffe, 126
 Vokabular, 30
 Volltextsuche, 134
 Vorhersagemodell, 165
 Vorhersagequalität, 158
 Vorhersage und Planung von Trajektorien, 341

W

Wachstums- und Ertragsprognose, 164, 165
 Wahrscheinlichkeitsverteilung, 273, 277, 280
 Wissensbasiertes System, 30, 229
 Wissensbasis, 30, 325, 335, 341
 Wissensgraph, 3, 68, 71, 126, 153, 156, 158, 162, 164, 165, 170, 322, 325, 327, 335, 339, 350
 Wissensingenieur, 128, 143
 Wissensintensiver Prozess, 297
 Wissensrepräsentation, 30
 Word-Embedding, 127, 145, 233
 multi-sense Word-Embedding, 131
 single-sense Word-Embedding, 131
 Word-Embeddings, 49, 131
 Wordnet, 98, 126
 Workflow-System, 417
 Wortnetz, 126, 143

Z

Zeitreihe, 161, 207, 272
 Multivariate Zeitreihe, 215
 Zeitreihendaten, 159, 161, 163, 165
 Zeitreihenanalyse, 215
 Zeitreihenprognose, 153
 Zeitreihenvorhersage, 156
 Zufallsvariable, 211, 277