Blick in die Blackbox

Nachvollziehbarkeit von KI-Algorithmen in der Praxis

www.bitkom.org

bitkom

Herausgeber

Bitkom
Bundesverband Informationswirtschaft,
Telekommunikation und neue Medien e. V.
Albrechtstraße 10 | 10117 Berlin
T 030 27576-0
bitkom@bitkom.org
www.bitkom.org

Verantwortliches Bitkom-Gremium

AK Artificial Intelligence
Big Data & Advanced Analytics

Projektleitung

Dr. Nabil Alsabah | Bitkom e. V.

Projekt-Team

Dr. Gerald Bauer | Fujitsu TDS GmbH
Dr. Andreas Dewes | KIProtect GmbH
Kentaro Ellert | PricewaterhouseCoopers GmbH
Dr. Sebastian Fischer | Deutsche Telekom
Dr. Antje Fitzner | Eucon Digital GmbH
Dr. Bernd Geiger | semafora systems GmbH
Lukas Graner | Fraunhofer-Institut für Sichere Informationstechnologie SIT
Maike Havemann | IBM Deutschland GmbH
Prof. Dr. Marco Huber | Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA und Universität Stuttgart
Janera Kronsbein | Eucon Digital GmbH
Matthias Noch | Atos SE
Nikolai Nölle | Detecon International GmbH
Claudia Pohlink | Deutsche Telekom
Hendrik Reese | PricewaterhouseCoopers GmbH
Andreas Rohnfelder | Fujitsu TDS GmbH
Robin Rojowiec | IBM Deutschland GmbH
Felix Rothmund | Fujitsu TDS GmbH
Nina Schaaf | Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA
Dominik Schneider | Detecon International GmbH
Dr. Horst Stein | Deutsche Telekom
Prof. Dr. Martin Steinebach | Fraunhofer-Institut für Sichere Informationstechnologie SIT
Dr. Susan Wegner | Deutsche Telekom
Dr. Frank Wisselink | Detecon International GmbH

Lektorat

Max Duhr | Bitkom e.V.
Jana Erhardt | Bitkom e.V.

Grafik und Layout

Daniel Vandré

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung und Executive Summary

1 Einleitung und Executive Summary

Nabil Alsabah

Von Suchalgorithmen über Entscheidungsbäume bis hin zu wissensbasierten Systemen: In den letzten sechs Dekaden haben KI-Experten eine Vielzahl an KI-Algorithmen entwickelt. Diese ermöglichen Computerprogrammen, auf nicht einprogrammierte Ereignisse adäquat zu reagieren. Denken Sie nur an Schachprogramme, Routenplaner und Expertensysteme: Der Programmierer muss nicht alle denkbaren Eingaben des Benutzers auflisten. Dank KI sind solche Programme flexibel genug, um eine situationsgerechte Reaktion auf den Input des Nutzers zu generieren.

Die regelbasierte KI ist erklärbar. Der Entscheidungsweg eines klassischen KI-Algorithmus ist transparent. Wir können nachvollziehen, warum sich z. B. ein Entscheidungsbaum für eine bestimmte medizinische Diagnose ausspricht. Deshalb bezeichnet man klassische KI-Algorithmen als White-Box-Verfahren. **Diese Verfahren operieren jedoch an der Spitze des Eisbergs dessen, was mit KI möglich ist.**

Die KI arbeitet dem Menschen zu. Sie nimmt ihm Aufgaben ab, so dass er sich auf wichtigere Aufgaben konzentrieren kann. Dafür ist es wichtig, das Problemlöseverhalten des Menschen dort abzubilden, wo er entlastet werden soll. Doch gerade dieses Verhalten ist oft nicht in Regeln kodifiziert, die man nachprogrammieren kann. Stellen Sie sich vor, man möchte eine Applikation entwickeln, die Großkatzen erkennt. Wie soll man das programmieren, was einen Schneeleoparden, einen Berglöwen oder einen Königstiger ausmacht? Oder wie soll man Gesichtserkennung mittels einer Beschreibung der Gesichtszüge implementieren können? Oder wie können Musik-Streaming-Dienste Ihre Lieblingsmusik im Vorfeld voraussagen?

Menschen lernen nicht nur anhand von Regeln, sondern auch Beispielen. Das Gebiet des maschinellen Lernens ahmt diese menschliche Fähigkeit nach. Die sogenannten Neuronalen Netze sind eine prominente Algorithmenfamilie des maschinellen Lernens. Sie analysieren eine große Datenmenge (z. B. Bilder von Großkatzen). Sie machen jene Merkmale aus, die z. B. Löwen oder Tiger kennzeichnen.

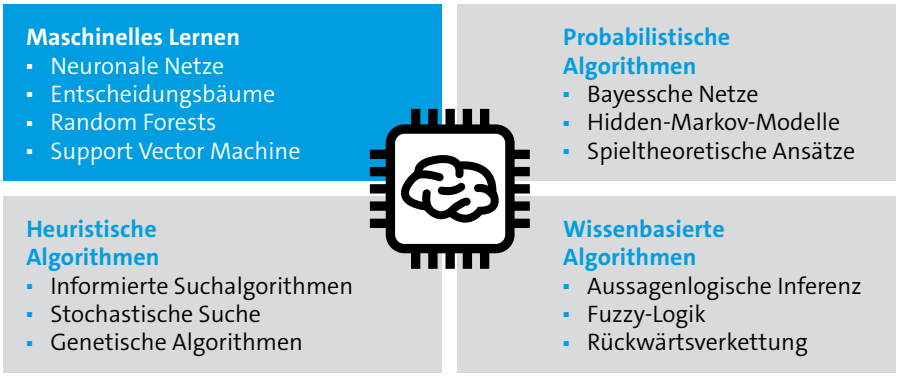


Abbildung 1: Maschinelles Lernen über Massendaten ist Dreh- und Angelpunkt der modernen KI.

Beispieldaten sind wichtig, um Neuronale Netze zu trainieren und das gewünschte Verhalten zu produzieren. In der Trainingsphase bekommen sie gegebenenfalls Feedback vom Trainer, ihren Output passen sie dementsprechend an. Am Ende hat man idealerweise eine Software, die Großkatzen richtig erkennt, einen Roboter, der Gegenstände im Weg meidet, und einen Sprachassistenten, der das gewünschte Lied abspielt. In der Trainingsphase lernt das Neuronale Netz komplexe Zusammenhänge. Diese sind für den Menschen nicht ohne weiteres nachvollziehbar. Deswegen entspricht ihr Verhalten einer sogenannten Black-Box.

Oft brauchen wir uns über die mangelnde Transparenz der Entscheidungsprozesse von Neuronalen Netzen nicht zu kümmern. Denken Sie dabei an automatisierte Filmempfehlungen, maschinelles Übersetzen oder intelligente Staubsauger. Der Entscheidungsprozess des KI-Algorithmus muss weder transparent noch nachvollziehbar sein. Es reicht, wenn das Ergebnis überzeugt.

Es gibt aber Anwendungsfelder, wo Nachvollziehbarkeit von KI-Algorithmen wichtig ist. **Wir brauchen Nachvollziehbarkeit, um Bias in den Trainingsdaten aufzudecken**. Diskriminierung leitet sich aus den Daten ab. Sie reproduziert menschliche Entscheidungen aus der Vergangenheit. Und genau da muss man ansetzen. Man muss also die strukturellen Defizite jener Organisationen angehen, in denen diskriminiert wird/wurde.

Wir brauchen nachvollziehbare KI, um regulatorische Auflagen zu erfüllen. In unserer Publikation **↗ Machine Learning und die Transparenzanforderungen der DS-GVO** haben wir uns ausgiebig mit dem gesetzlichen Rahmen auseinandergesetzt. So kommt die Studie zu dem Ergebnis:

»Datenverarbeitungen unter Einsatz von ML oder KI fallen, soweit personenbezogene Daten betroffen sind, in den Anwendungsbereich der DS-GVO. Das erfasst natürlich auch die Einhaltung der Transparenzgrundsätze (insbesondere Informationspflichten), sowie die Geltung des Verbots mit Erlaubnisvorbehalt.«

Die Industrie braucht unter Umständen nachvollziehbare KI-Algorithmen, um die Robustheit der eingesetzten Software-Lösungen sicherzustellen. In **Kapitel 2** gehen Gerald Bauer, Felix Rothmund und Andreas Rohnfelder von Fujitsu auf konkrete Anwendungen ein, für die Nachvollziehbarkeit im industriellen Kontext erforderlich ist. Ein Fallbeispiel beschreibt eine Textilfabrik, in der eine kamerabasierte KI-Lösung die Stoffqualität kontrolliert. Wird eine Anomalie erkannt, werden je nach Fehlertyp (Risse, Randbeschnitte oder Verunreinigungen) die adäquaten Maßnahmen getroffen. Im Ergebnis werden Verschnitte minimiert und Stoffstücke aussortiert. Bei schwerwiegenden Fehlern wird die Produktion angehalten.

Sowohl bei dieser wie bei vielen anderen Anwendungen wird die KI-Nachvollziehbarkeit in der Industrie mit sogenannten »lokalen Erklärmodellen« hergestellt. Ein prominenter Vertreter dieser Algorithmenfamilie ist LIME *(Local Interpretable Model-agnostic Explanations)*. **Dieses Verfahren identifiziert jene Merkmale (z. B. Farbe, Form, Größe), die für die Entscheidung eines Neuronalen Netzes ausschlaggebend sind.** Hat man z. B. ein Neuronales Netz, das Bananen und Äpfel erkennt, so könnte das Merkmal Form entscheidender sein als das Merkmal Farbe. In **Kapitel 3** erklärt Andreas Dewes von KIProtect die Funktionsweise von LIME sowie von ähnlichen Verfahren.

Kapitel 4 zeigt ein anderes Beispiel für die nachvollziehbare KI. Lukas Graner und Martin Steinebach vom Fraunhofer SIT beschreiben eine Methode der Autorschaftsverifikation. Ihr Anwendungsfall macht jene stilistischen Elemente aus, die typisch oder untypisch für eine Autorschaft sind. Ihr Verfahren generiert nicht nur eine Wahrscheinlichkeit der Übereinstimmung der Autoren in den zu überprüfenden Texten, sondern zeigt, welche Stilelemente zu dieser Einschätzung führten.

Nachvollziehbare KI ist aber auch wichtig, um Cyberattacken gegen Neuronale Netze abzuwehren. Manche Cyberangriffe können die Daten auf eine – für den Menschen – kaum wahrnehmbare Art und Weise verändern. Versteht man aber welche Faktoren für die Entscheidung von Neuronalen Netzen entscheidend sind, so kann man ihre Achillesfersen auch feststellen. In **Kapitel 5** beschreiben Horst Stein, Sebastian Fischer und Claudia Pohlink von der Deutschen Telekom, wie Angriffe (Adversarial Attacks) die Funktionsfähigkeit von ML-Modellen in der Objekterkennung beeinträchtigen können. Konkret erläutern sie, wie die autonome Fahrzeugsteuerung durch falsch klassifizierte Verkehrszeichen und die KI-gestützte Spracherkennung gestört werden kann. Sie empfehlen Maßnahmen, um die Robustheit von ML-Modellen zu stärken.

In **Kapitel 6 und 7** wagen Maike Havemann und Robin Rojowiec (IBM) sowie Nina Schaaf und Marco Huber (Fraunhofer IPA) eine tiefgehende Auseinandersetzung mit den technischen Details der nachvollziehbaren KI. In **Kapitel 8** beschreibt Bernd Geiger von semafora systems, wie man mit klassischen KI-Methoden elektronische Wartungsbücher automatisch in ausführbaren Code umwandelt. Der Beitrag zeigt, dass White-Box-Verfahren auch in komplexen Umgebungen durchaus ihre Existenzberechtigung haben.

In **Kapitel 9** argumentieren Frank Wisselink, Nikolai Nölle und Dominik Schneider (Detecon), **dass wir nachvollziehbare KI brauchen, um gesellschaftliches Vertrauen zu schaffen**. Dabei schlagen

sie auch die Brücke zu ethischen Debatten um die Künstliche Intelligenz. Schließlich geht es in dem Beitrag von PWC in **Kapitel 10** um ein Zertifizierungskonzept für Transparenz und Nachvollziehbarkeit von KI-Systemen.

Das Forschungsgebiet um xAI (Explainable AI) geht bis in die neunziger Jahre zurück. Doch die vier in dieser Publikation beschriebenen Faktoren – gesetzliche Auflagen, Bedürfnisse der Industrie, Schutz vor Adversarial AI und ethische Bedenken – beflügeln die angewandte Forschung in diesem Bereich. Wir hoffen, dass diese Publikation einen ausgewogenen Überblick über die Anwendungsmöglichkeiten nachvollziehbarer KI-Algorithmen geben kann.

2 Erklärbare KI in der Praxis

2 Erklärbare KI in der Praxis

Gerald Bauer, Felix Rothmund, Andreas Rohnfelder

2.1 Einleitung

Spricht man heutzutage von Künstlicher Intelligenz (KI), so sind damit in der Regel lernende Algorithmen gemeint, die aus gewaltigen Datenmengen probabilistische Modelle ableiten. Insbesondere die Methoden des Deep Learning ermöglichen es uns heute, komplexe Probleme zu lösen, die noch vor einigen Jahren unlösbar erschienen.

So erreichen Tiefe Neuronale Netze (engl. Deep Neural Nets) mittlerweile in einigen Bereichen Genauigkeiten, die weit über die kognitiven Möglichkeiten des Menschen hinausgehen [1, 2, 3]. Sie bestehen aus einer Vielzahl künstlicher Neuronen, welche die Lernfähigkeit des menschlichen Nervensystems mathematisch nachbilden. Dabei steigt die Komplexität der verwendeten Modelle immer weiter: Während man im Jahr 2000 noch Neuronale Netze mit 10^2 verbundenen Neuronen anlernte, wurden 2015 bereits Modelle mit mehr als 10^6 Neuronen trainiert [4].

Nicht zuletzt aufgrund der enormen Komplexität der Modelle, ist es nicht mehr ohne weiteres möglich, die zugrunde liegenden Entscheidungsprozesse nachzuvollziehen. Man spricht deshalb vom sogenannten Black-Box-Verhalten. Trotz der erfolgreichen Anwendung dieser Systeme in vielen Bereichen des alltäglichen Lebens, hält sich deshalb eine gewisse Skepsis gegenüber den Methoden des Deep Learning.

Ist diese Zurückhaltung berechtigt? Können wir Neuronalen Netzen vertrauen? Vor allem mit dem Einzug von KI-Modellen in sicherheitskritische Bereiche, wie zum Beispiel im Bereich Industrie 4.0, im Gesundheitswesen oder der Justiz, drängen sich diese Fragen auf. Wenn Entscheidungen von KI-Modellen signifikante Auswirkungen auf das alltägliche Leben haben, sei es bei der Beurteilung der Kreditwürdigkeit oder der Früherkennung von Erkrankungen, müssen wir diesen Systemen vertrauen können. Vertrauen ist die Grundlage für Akzeptanz. Und nur durch die Akzeptanz des Menschen können die Systeme den erhofften Mehrwert bringen.

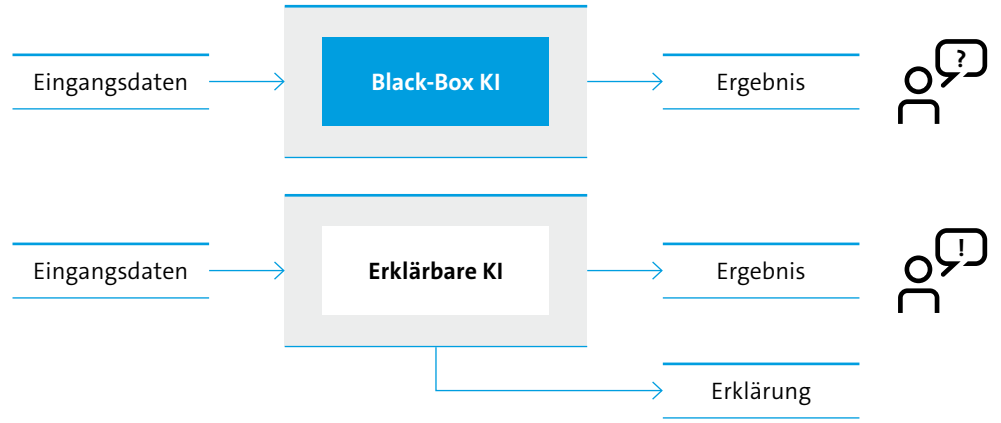


Abbildung 2: Im Gegensatz zur Black-Box-KI liefert die Erklärbare KI neben dem Ergebnis auch eine passende Erklärung.

Um eine positive Erwartungshaltung herzustellen und somit das notwendige Vertrauen zu schaffen, ist ein Verständnis über die Handlungsweise von KI-Modellen unverzichtbar. Es müssen folglich Methoden gefunden werden, die menschenverständlich erklären, auf welcher Grundlage KI-Systeme ihre Entscheidungen treffen. Damit beschäftigt sich das Forschungsfeld der Erklärbaren KI (siehe Abbildung 2).

Im Folgenden wird eine Auswahl von Algorithmen zur Visualisierung von Entscheidungsprozessen in Neuronalen Netzen vorgestellt. Die Methoden werden anhand von KI-Systemen aus der Qualitätssicherung im Textilumfeld und der Medizintechnik erklärt.

2.2 KI-basierte Qualitätssicherung in der Textilverarbeitung

In einer Produktionslinie wird mit Hilfe eines kamerabasierten KI-Systems die Qualität des zu verarbeitenden Stoffes überwacht. Im ersten Schritt werden Anomalien erkannt, also Stoffstücke identifiziert, welche von einem vorgegebenen Muster abweichen. Die gefundenen Anomalien sollen genauer analysiert und durch ein Neuronales Netz verschiedenen Fehlertypen zugeordnet werden. Je nach Fehlertyp (Risse, Randbeschnitte, Verunreinigungen, ...) können dann in weiteren Prozessschritten der Verschnitt minimiert, Stoffstücke aussortiert oder bei schwerwiegenden Fehlern die Produktion angehalten werden.

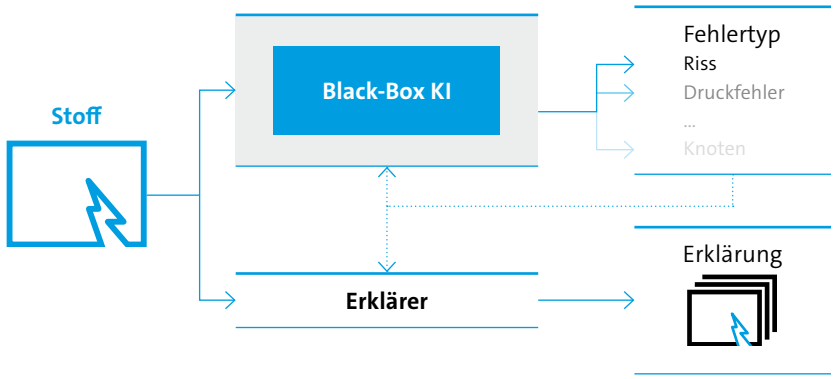


Abbildung 3: Fehlerklassifikation in der Textilverarbeitung

Das KI-System wurde mit verschiedenen Fehler- und Stoffarten trainiert. Es liefert für die vorhandenen Testdaten und den anvisierten Anwendungsfall ausreichend hohe Genauigkeiten. Jedoch ist nicht ersichtlich, warum ein Fehler als solcher erkannt wird. Kann man also sicher sein, dass das Netzwerk gelernt hat, was es hätte lernen sollen? Kann man den produktiven Einsatz des Systems rechtfertigen?

Um diese Fragen zu beantworten, wird das KI Modell um ein Erklärer-Modul erweitert, welches versucht, die Entscheidungsprozesse verständlich zu machen (siehe Abbildung 3).

In der Praxis sind die künstlichen Neuronen eines Neuronalen Netzes in Schichten (engl. layers) organisiert. Die Eingangsdaten wandern Schicht für Schicht durch das Netzwerk, bis in der letzten Schicht die Ausgangsneuronen aktiviert werden. Bei Klassifikationsproblemen entspricht die Aktivierung eines Ausgangsneurons dem sogenannten Confidence Score, also der geschätzten Zugehörigkeitswahrscheinlichkeit zu einer bestimmten Klasse.

In der Bildklassifikation haben sich vor allem sogenannte Faltungsnetze (engl. Convolutional Neural Nets, kurz CNNs) bewährt [5]. Ein als Deep Dream populär gewordenes Verfahren erlaubt es, für jedes Ausgangsneuron eines CNNs iterativ ein Bild zu erzeugen, welches das Neuron besonders stark aktiviert [6, 7]. Man kann so Bilder generieren, die repräsentativ für eine bestimmte Klasse stehen und dementsprechend charakteristische Muster zeigen. Dadurch ist es möglich, einen ersten Einblick zu erhalten, was das Netzwerk gelernt hat.

Im Folgenden (siehe Abbildung 4) sind Visualisierungen zweier Klassenmodelle (jeweils rechts) und entsprechende Beispiele aus dem Trainingsdatensatz zu sehen (jeweils links mit cyan markierten Fehler). Im ersten Beispiel (links: Randbeschnitt) kann man deutlich erkennen, dass das Netzwerk Merkmale gelernt hat, die typisch für den Fehlertyp sind (Anordnung von orthogonalen Linien). Im zweiten Beispiel (rechts: Klebereste) ist das Muster weniger eindeutig.

Dennoch ist eine gewisse Ähnlichkeit zu erkennen. Wir können also davon ausgehen, dass das Netzwerk die Charakteristik des Fehlers korrekt gelernt hat.

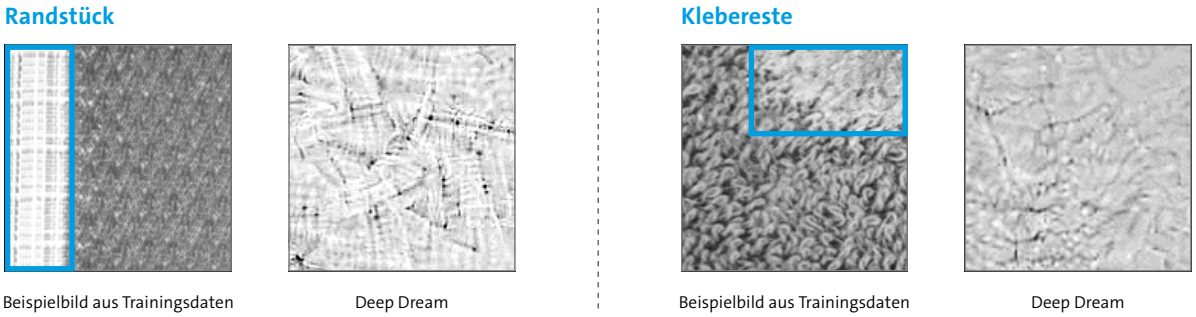


Abbildung 4: Visualisierung zweier Fehlerklassenmodelle mittels Deep Dream

Es gilt allerdings zu erwähnen, dass es sich hierbei nur um ein mögliches Bild handelt, welches eine starke Aktivierung erzeugt. Die Visualisierungen reichen also nicht aus, um das dem Netzwerk zugrundeliegende Klassenmodell zu erklären. Doch obwohl das Verfahren nur einen kleinen Einblick in die Black-Box bietet, kann es in einigen Fällen helfen, gravierende Modellfehler zu erkennen. Hätte man zum Beispiel einen Fehlertypen nur auf einem speziellen Stofftypen trainiert, so hätte das System womöglich gelernt, statt dem Fehler das Muster des Stoffes zu erkennen. Solche Probleme können sich dann in den Visualisierungen der Klassenmodelle zeigen.

2.3 Welche Merkmale sind entscheidend?

Zwar erlaubt die Visualisierung von Klassenmodellen einen ersten Einblick in das Neuronale Netz, jedoch liefert sie keine Hinweise darüber, wie und warum ein Modell im Einzelfall entscheidet. Verschiedene Verfahren erlauben es, Merkmale zu identifizieren, welche besonders stark zur Entscheidungsfindung beitragen. Im Folgenden seien zwei Verfahren kurz vorgestellt.

LIME (Local Interpretable Model-agnostic Explanations) erlaubt es, Merkmale in den Eingangsdaten zu identifizieren, die für oder gegen die Zuordnung einer Instanz zu einer bestimmten Klasse sprechen. LIME approximiert das Black-Box-Modell lokal, das heißt begrenzt im Bereich der Eingangsdaten, indem diese iterativ verändert werden und die Antwort der Black-Box beobachtet wird. Die so erhaltene Approximation gilt also nur für die jeweiligen Eingangsdaten [8].

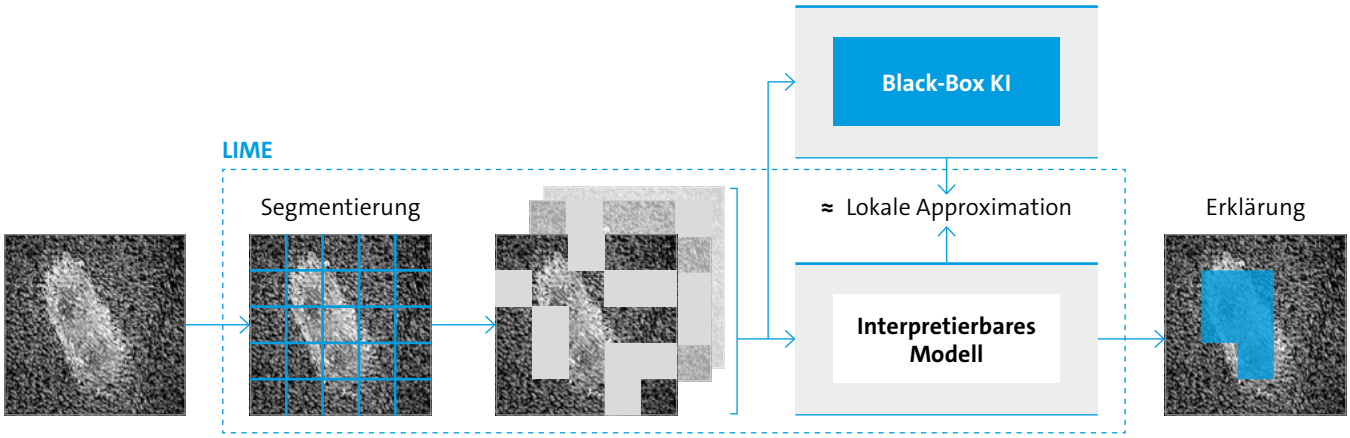


Abbildung 5: LIME am Beispiel eines Textilfehlers. Relevante Bereiche für die Klasse Klebereste sind in der Erklärung farblich markiert.

Für die Klassifikation von Bilddaten, wie im gezeigten Fall, wird das Bild zunächst in kleinere Bereiche unterteilt. Diese werden dann in unterschiedlicher Kombination verdeckt. Für jedes der veränderten Bilder, wird die Klassenzugehörigkeit durch die Black-Box geschätzt. Es wird nun iterativ ein interpretierbares Modell erzeugt, das das Verhalten der Black-Box nachbildet. Dieses Modell erlaubt es nun, Bildbereiche zu identifizieren, welche die Klassifikation unterstützen (Verdeckung verringert Confidence Score) oder gegen sie sprechen (Verdeckung erhöht Confidence Score). Wir können dadurch nachvollziehen, aufgrund welcher Bildbereiche unser Modell beispielsweise eine Verunreinigung durch Klebstoffreste als solche erkennt (siehe Abbildung 5).

RISE (Randomized Input Sampling for Explanation) ist ein anderes verbreitetes Verfahren, das relevante Bildbereiche lokalisiert, welche die Zuordnung eines Bildes zu einer bestimmten Klasse unterstützen [9]. Dabei wird das Eingangsbild mit einer Vielzahl zufällig generierter Masken überlagert. Je nachdem, welche Bildbereiche abgedeckt werden, verändert sich die Antwort der Black-Box. Hieraus lässt sich dann eine sogenannte Heatmap ableiten, welche zeigt, welche Bildbereiche für die Klassifikation ausschlaggebend sind. Im Beispiel unten (siehe Abbildung 6) ist deutlich zu erkennen, dass das System das Stoffstück aufgrund eines Loches als fehlerhaft erkannt hat.

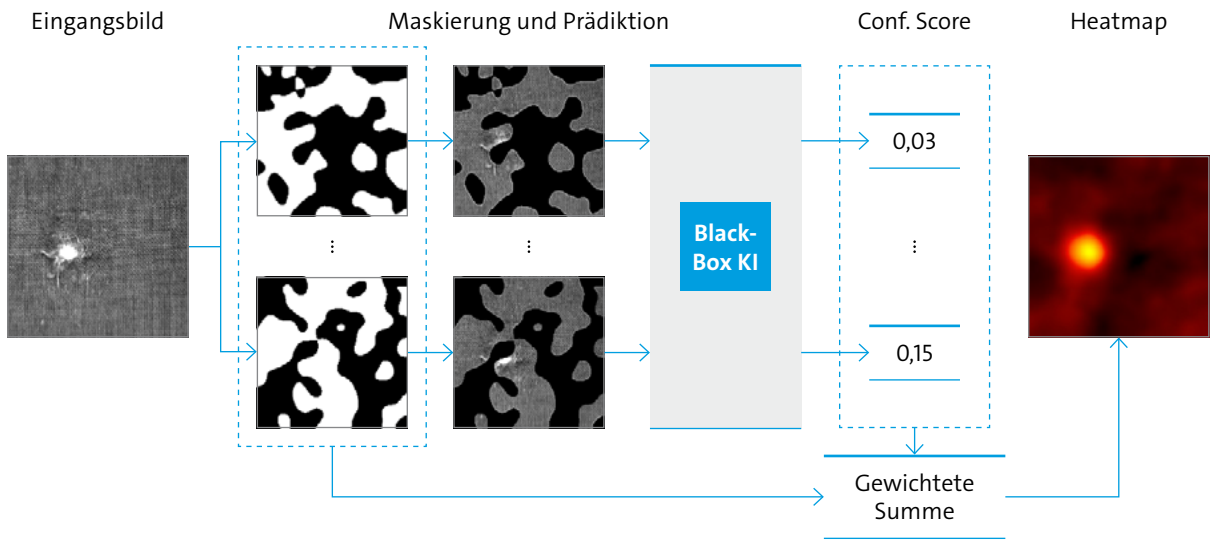


Abbildung 6: RISE Algorithmus am Beispiel eines Textilfehlers. In Anlehnung an [9]

Die gezeigten Ansätze identifizieren relevante Merkmale (Bereiche auf Bildern) für die Entscheidungsfindung und erlauben es so, die Arbeitsweise des Modells besser einzuschätzen. So werden die getroffenen Entscheidungen nachvollziehbar und schaffen damit Vertrauen in das System. Sollten im Betrieb Fehlerkennungen auftreten, so können diese schnell analysiert und interpretiert werden und damit zur weiteren Verbesserung des Systems beitragen.

2.4 Beispiel: Erkennung von Krankheiten basierend auf Genmutationen

Besonders in der behandelnden Medizin ist es unerlässlich, dass Modellvorhersagen nachvollziehbar sind. In einigen Bereichen der Diagnostik haben KI-Modelle bereits die Leistungsfähigkeiten von dermatologischem Fachpersonal übertroffen [3]. Dabei müssen die Systeme möglichst transparent sein, sodass behandelnde Ärztinnen und Ärzte fachlich fundierte Entscheidungen treffen können.

In den Fujitsu Labs in Japan wurde ein Verfahren entwickelt, das es erlaubt, von Genmutationen auf bestimmte Krankheiten zu schließen und das Ergebnis mit wissenschaftlichen Arbeiten aus einer umfassenden Datenbank zu begründen [10]. Da Genmutationen in einer Vielzahl von Variationen auftreten können, und mit unterschiedlichen Symptomen und Faktoren einhergehen, sind die Eingangsdaten für das KI-Modell in Graphen organisiert, die Zusammenhänge zwischen unterschiedlichen Faktoren flexibel darstellen können. Doch Graph-Strukturen sind nicht ohne weiteres als Eingangsdaten für Neuronale Netze geeignet. Durch ein als DeepTensor vorgestelltes Verfahren ist es jedoch möglich, Graph-Strukturen in sogenannte Kern-Tensoren

fixer Größe umzuwandeln und diese dann mit einem Neuronalen Netz zu analysieren [11]. DeepTensor liefert dabei neben der geschätzten Krankheit auch sogenannte Inferenzfaktoren. Diese sind, ähnlich wie die markanten Bildbereiche in den vorherigen Beispielen, Merkmale in den Eingangsdaten, welche entscheidend für die Klassifikation sind (siehe Abbildung 7).

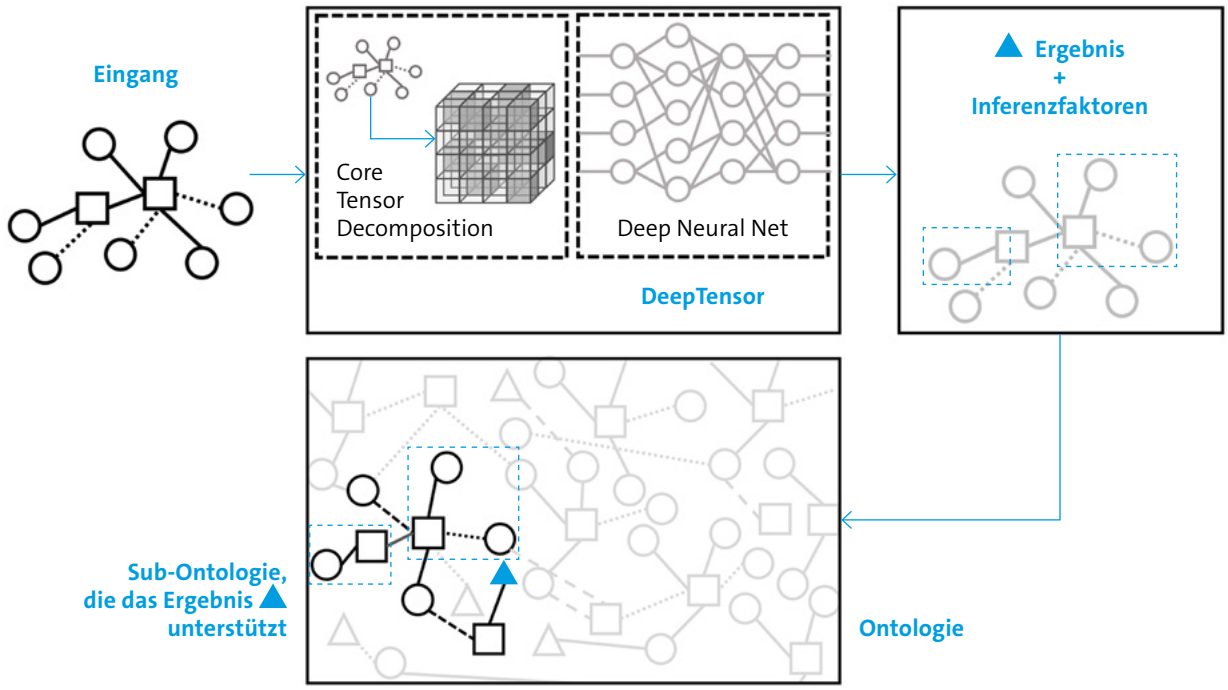


Abbildung 7: Erklärung durch Kombination von DeepTensor und Ontologien

Durch Abgleich der Inferenzfaktoren mit Mustern in einer umfassenden Ontologie, können Sub-Graphen identifiziert werden, welche das Ergebnis unterstützen (siehe Abbildung 7). Als Ontologie versteht man gesammeltes Wissen in Form eines Graphen (engl. Knowledge Graph), der Entitäten (z. B. Krankheit A und Gen 1) miteinander in Verbindung setzt (z. B. Krankheit A folgt auf Gen 1). Die Ontologie wurde aus einer großen Menge an wissenschaftlichen Arbeiten mithilfe von KI-Methoden aus dem Bereich Natural Language Processing (NLP) erstellt. Sie verknüpft wissenschaftliche Arbeiten mit Genmutationen, Krankheiten und andere Faktoren und dient als Basis, um Entscheidungen des Neuronalen Netzes zu erklären.

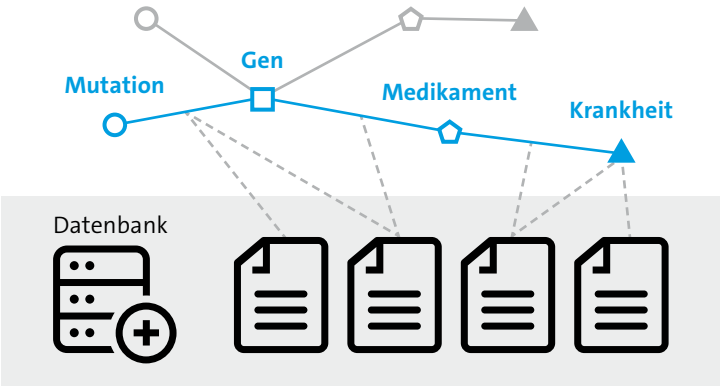


Abbildung 8: Verknüpfung von Genmutationen und Krankheiten mit wissenschaftlichen Arbeiten aus einer Datenbank

Wie in Abbildung 8 zu sehen ist, können somit Genmutationen und andere Faktoren mit Dokumenten in einer Datenbank verknüpft werden, auf deren Grundlage behandelnde Ärztinnen und Ärzte die Vorhersage des Modells entweder annehmen oder ablehnen können.

2.5 Fazit

Die Erklärbarkeit Künstlicher Intelligenz ist bis dato kein abschließend gelöstes Problem. Aber es werden kontinuierlich neue Algorithmen und Verfahren entwickelt, die immer tiefere Einblicke in die Black-Box Neuronaler Netze erlauben. Bei der Entwicklung neuer KI-Modelle können die Methoden der Erklärbaren KI bereits jetzt helfen, schwerwiegende Systemfehler frühzeitig aufzuspüren und in Folge robustere Systeme zu entwickeln.

Damit KI-Systeme das nötige Vertrauen gewinnen können und ihre Vorteile auch in kritischen Bereichen nutzbar werden, ist die menschenverständliche Erklärung ihrer Vorhersagen und Entscheidungen elementar. Denn nur wenn Prozesse nachvollziehbar sind, wird ihnen das nötige Vertrauen entgegengebracht. Nur dann können wir fundamentale Fehlentscheidungen und ihre teils fatalen Folgen rechtzeitig erkennen, sei es in der Medizintechnik, in Produktionsumgebungen im Bereich Industrie 4.0 oder beim autonomen Fahren.

Es ist also davon auszugehen, dass dem Forschungsfeld der Erklärbaren KI weiterhin große Aufmerksamkeit geschenkt wird. Auf lange Sicht werden sich Systeme durchsetzen, die sowohl performant als auch zuverlässig und verständlich sind. Denn letztendlich basiert Vertrauen auf Verständnis.

2.6 Literaturverzeichnis

[1] Silver, David, et al. »Mastering the game of Go with deep neural networks and tree search.« nature 529.7587 (2016): 484.

[2] Assael, Yannis M., et al. »Lipnet: End-to-end sentence-level lipreading.« arXiv preprint arXiv:1611.01599 (2016).

[3] Brinker, Titus J., et al. »Deep learning outperformed 136 of 157 dermatologists in a head-to-head dermoscopic melanoma image classification task.« European Journal of Cancer 113 (2019): 47–54.

[4] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. »Deep learning book.« MIT Press 521.7553 (2016): 800.

[5] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. »Imagenet classification with deep convolutional neural networks.« Advances in neural information processing systems. 2012.

[6] Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. »Deep inside convolutional networks: Visualising image classification models and saliency maps.« arXiv preprint arXiv:1312.6034 (2013).

[7] Mordvintsev, Alexander, Christopher Olah, and Mike Tyka. »Inceptionism: Going deeper into neural networks.« (2015).

[8] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. »Why should i trust you?: Explaining the predictions of any classifier.« Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 2016.

[9] Petsiuk, Vitali, Abir Das, and Kate Saenko. »Rise: Randomized input sampling for explanation of black-box models.« arXiv preprint arXiv:1806.07421 (2018).

[10] Fuji, Masaru, et al. »Explainable AI Through Combination of Deep Tensor and Knowledge Graph.« FUJITSU SCIENTIFIC & TECHNICAL JOURNAL 55.2 (2019): 58–64.

[11] Maruhashi, Koji. »Deep Tensor: Eliciting New Insights from Graph Data that Express Relationships between People and Things.« Fujitsu Sci. Tech. J 53.5 (2017): 26–31.

3 Lokale Nachvollziehbarkeit von ML-Modellen

3 Lokale Nachvollziehbarkeit von ML-Modellen

Andreas Dewes

3.1 Einleitung

ML-Verfahren werden von immer mehr Unternehmen eingesetzt, um Prozesse zu automatisieren und Vorhersagen zu treffen. Die Entwicklung der Systeme erfolgt dabei entweder innerhalb des Unternehmens oder in Zusammenarbeit mit externen Partnern. Um Verfahren des maschinellen Lernens zu implementieren, wird dabei fast immer auf Open-Source Lösungen zurückgegriffen. Insbesondere Programmiersprachen wie Python und R haben hierbei in den letzten Jahren enorm an Bedeutung gewonnen. Open-Source-Bibliotheken wie Tensorflow oder PyTorch machen es hierbei einfach, auch moderne Verfahren wie Deep Learning in wenigen Schritten zu implementieren, was diese Techniken für viele Unternehmen überhaupt erst nutzbar macht.

Ähnlich wie normale Software-Systeme müssen auch ML-Verfahren in betriebliche Prozesse integriert werden, um Nutzen zu schaffen. Und genau wie bei normaler Software können auch hier Probleme auftreten, die dazu führen, dass sich ML-Systeme nicht wie beabsichtigt verhalten. Da die Systeme nicht explizit programmiert, sondern vielmehr durch Daten trainiert werden ist die Testbarkeit und Überwachbarkeit dabei oft sehr viel schwieriger als bei normaler Software. Methoden zur Untersuchung von Robustheit, Sicherheit und Nachvollziehbarkeit spielten bei der Ausbildung von Data-Scientists- und Machine-Learning Experten zudem bisher eine eher untergeordnete Rolle, dementsprechend ist der Wissensstand zu diesem Thema selbst bei erfahrenen Spezialisten oft noch gering. Aktuell wird eine Vielzahl an Lösungen entwickelt, um eine Überwachung und Kontrolle von ML-Verfahren in automatisierter Weise zu ermöglichen und die Entwickler der Verfahren dabei zu unterstützen, diese sicher, robust, fair, nachvollziehbar und transparent zu gestalten. Einige dieser Ansätze sind in kommerzielle ML-Lösungen integriert [1, 2, 3].

Algoneer [4, 5] ist ein vom BMBF im Rahmen des »Prototype Fund« gefördertes Open-Source-Projekt, das zum Ziel hat, eine offene und frei verfügbare Software zu schaffen, mit der ML-Systeme kontinuierlich getestet und auditiert werden können. Die Software soll ermöglichen, ML-Verfahren bereits während der Entwicklung zu testen und diese beim Produktiveinsatz kontinuierlich zu überwachen. Bei der Planung der Software wurde darauf Wert gelegt, dass die Funktionalität ohne größeren Aufwand in bestehende Entwicklungsprozesse eingebettet werden kann. Die Software bietet eine Reihe von Blackbox- sowie Whitebox-Tests, die Datensätze sowie Modelle auf unterschiedliche Eigenschaften testen und die Ergebnisse in einem einfach verständlichen Format aufbereiten. Die Software besteht aus einer Python-Bibliothek, die sich einfach in bestehende ML-Workflows integrieren lässt und über Anbindungen an verschiedene ML-Bibliotheken (Scikit-Learn, Tensorflow) verfügt. Tests von Datensätzen und Modellen lassen sich so datenschutzkonform lokal ausführen und auswerten. Die Ergebnisse der Tests können ebenfalls lokal ausgewertet werden oder an einen zentralen API-Dienst geschickt werden, der

diese speichert und für die Analyse mithilfe einer Web-Software aufbereitet. Dies ist insbesondere für die Entwicklung von ML-Verfahren in Teams sowie für das kontinuierliche Testen im Rahmen eines Continuous-Integration-Workflows relevant, wo Tests automatisiert ausgeführt und Testergebnisse zentral und nachweissicher gespeichert werden sollen. Algoneer implementiert hierbei eine Reihe von Verfahren, die zur Erklärung der Vorhersagen von ML-Modellen genutzt werden können. Wo immer möglich, werden bestehende Open-Source-Bibliotheken genutzt. Um automatisierte Tests zu ermöglichen definiert Algoneer zudem eine Schemasprache zur Definition von Daten- sowie Algorithmen-Schemata, welche verwendet werden können, um Tests spezifisch an Datentypen und einzelne ML-Verfahren anzupassen.

Im Rahmen des Projekts wurde eine Vielzahl an Verfahren für die Erklärung von ML-Modellen untersucht und implementiert. Die folgenden Abschnitte beschreiben anhand von Beispielen einige dieser Verfahren. Die Implementierung der Beispiele ist hierbei ebenfalls als Open Source verfügbar. Erklärungen zu einigen der hier besprochenen Verfahren finden sich in C. Molnars E-Book über interpretierbare Machine-Learning-Verfahren [6]. Zur Erläuterung der Verfahren wird ein von der Universität Porto publizierter Datensatz verwendet [7]. Dieser enthält Daten zur Anzahl der täglichen Fahrradleihen in Porto in den Jahren 2011 und 2012 mitsamt einer Reihe von zugehörigen kategorialen sowie numerischen Attributen, wie z. B. der Temperatur oder Luftfeuchtigkeit. Zur Vorhersage der Anzahl an Fahrradleihen in Abhängigkeit von diesen Attributen trainieren wir ein auf der »random forest«-Methodik basierendes ML-Verfahren. Die Vorhersagen des Modells versuchen wir anschließend mithilfe verschiedener Verfahren zu erklären.

3.2 Kontrafaktische Erklärungen (counterfactual explanations)

Kontrafaktische Erklärungen [17] sind eine sehr einfache Möglichkeit, ML-Modelle nachvollziehbarer zu machen. Die Idee hierbei ist simpel: Ausgehend von einem gegebenen Datenpunkt wird ein neuer Datenpunkt gesucht, der die Entscheidung des Modells signifikant ändert und dabei möglichst nahe am ursprünglichen Datenpunkt liegt. Was genau eine signifikante Änderung darstellt, hängt hierbei vom Modelltyp ab: Bei Klassifikationsmodellen würde man z. B. eine Änderung der vorhergesagten Klasse für den Datenpunkt als signifikante Änderung betrachten, bei Regressionsmodellen kann beispielsweise eine bestimmte Änderung des Vorhersagewertes untersucht werden. In unserem Bike-Sharing-Beispiel könnten wir z. B. untersuchen, welche minimale Änderung der Attributwerte ausgehend von einem gegebenen Datenpunkt eine Erhöhung der vorhergesagten Anzahl an Fahrten um 100 zur Folge hätte. Dies erlaubt uns, ähnlich zu den anderen Verfahren, besser zu verstehen, was die Abhängigkeit des Vorhersagewerts von den Eingabedaten ist. Zur Generierung von kontrafaktischen Beispielen existieren eine Reihe von Verfahren, in der Praxis werden oft heuristische Suchverfahren eingesetzt.

3.3 Partielle Abhängigkeiten (partial dependence plot)

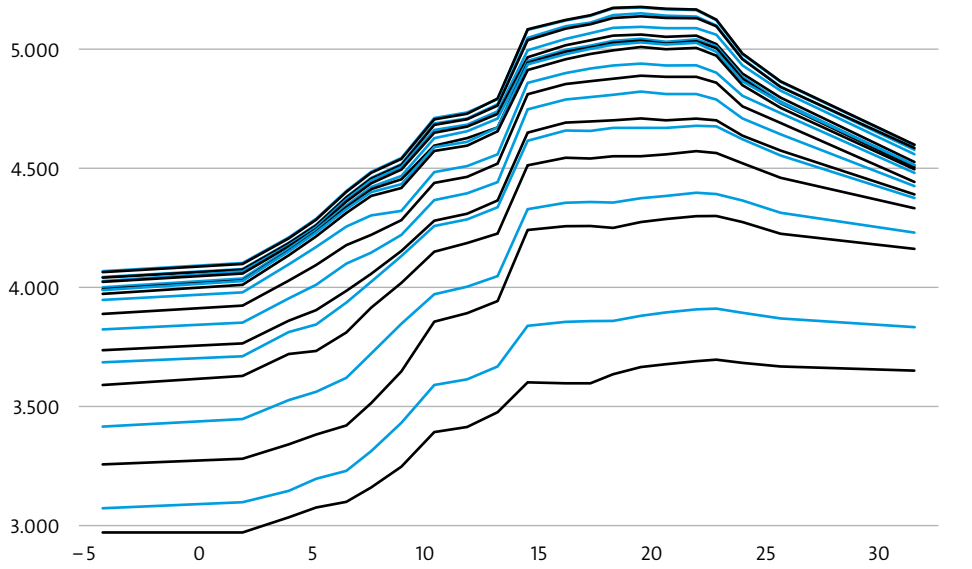


Abbildung 9: Partielle Abhängigkeit der Anzahl der Fahrradleihen von der Temperatur, dargestellt für verschiedene Werte der Luftfeuchtigkeit (je höher die Luftfeuchtigkeit, desto geringer die Anzahl der Fahrradleihen für eine gegebene Temperatur).

Bei diesem Verfahren wird der Einfluss eines einzelnen Attributs auf die Vorhersage eines ML-Modells untersucht [8, 9]. Um mit der Methode für unser ML-Modell beispielsweise den Einfluss der Temperatur auf die Anzahl der Fahrradleihen zu untersuchen, ersetzen wir in sämtlichen Datenpunkten des Testdatensatzes den Wert der Temperatur durch einen künstlichen Wert, der einem plausiblen Temperaturwert entspricht. Wir mitteln dann den Vorhersagewert über all diese synthetischen Datenpunkte und tragen den Mittelwert in einem Diagramm auf. Wir wiederholen diesen Vorgang nun mit einer Reihe von Temperaturwerten und tragen alle so erhaltenen Mittelwerte in unser Diagramm ein. Die resultierende Kurve zeigt den gemittelten Effekt der Temperatur auf die Vorhersage des Modells und erlaubt uns, den vom Modell angenommenen Zusammenhang zwischen der Temperatur und der Anzahl der Leihvorgänge zu untersuchen. Wir können dies zusätzlich für unterschiedliche Werte weiterer Attribute wiederholen, um kombinierte Effekte zu untersuchen. Dies liefert uns grundlegende Aufschlüsse über die internen Zusammenhänge des ML-Modells. Abbildung 9 zeigt exemplarisch die partielle Abhängigkeit der Anzahl an Fahrradleihen in unserem ML-Modell von der Temperatur. Jede Kurve zeigt die Abhängigkeit der Temperatur auf die Anzahl der Leihen für einen spezifischen Wert der Luftfeuchtigkeit. Wie man sieht, steigt in unserem Modell unabhängig von der Luftfeuchtigkeit die Anzahl der Fahrradleihen mit steigender Temperatur zunächst an, sinkt jedoch ab einem

Temperaturwert von ca. 24 °C wieder ab. Im Bereich um 10 °C lässt sich zudem ein starker Anstieg der Leihen feststellen. Die Visualisierung legt nahe, dass unser Modell eine starke Temperaturabhängigkeit aufweist, die Luftfeuchtigkeit jedoch ebenfalls einen großen Einfluss auf die Anzahl der Leihen hat, was sich aus der relativen Verschiebung der einzelnen Kurven in Abhängigkeit der Luftfeuchtigkeit ergibt. Bei der Interpretation der partiellen Abhängigkeiten ist zu beachten, dass es sich bei den ermittelten Werten nur um Mittelwerte handelt, die ermittelte Temperaturabhängigkeit des Modells kann für einzelne Datenpunkte erheblich hiervon abweichen. Zusätzlich ist die Generierung von synthetischen Datenpunkten durch Variation eines einzelnen Attributs wie der Temperatur über einen großen Wertebereich nicht immer sinnvoll, denn die so entstehenden Werte sind in vielen Fällen unrealistisch und haben damit nur einen geringen Aussagewert, da sie vom ML-Modell beim Training nie berücksichtigt wurden und auch in der Realität mit sehr geringer Wahrscheinlichkeit auftreten. In unserem Beispiel enthält der Datensatz beispielsweise sowohl die gefühlte als auch die gemessene Temperatur, welche sehr stark korreliert sind. Variieren wir wie in unserem Beispiel lediglich einen der beiden Werte, erhalten wir Datenpunkte, die sehr unrealistisch sind und beispielsweise eine gemessene Temperatur von 20 °C mit einer gefühlten Temperatur von 0 °C kombinieren. Abbildung 10 zeigt die Korrelation unterschiedlicher Attribute exemplarisch an der Luftfeuchtigkeit und Temperatur sowie an der gefühlten und gemessenen Temperatur. Um solche Abhängigkeiten bei der Untersuchung des ML-Verfahrens besser berücksichtigen zu können, wurde u. a. die Technik der kumulierten lokalen Effekte (accumulated local effects) entwickelt, die wir im nächsten Abschnitt behandeln werden.

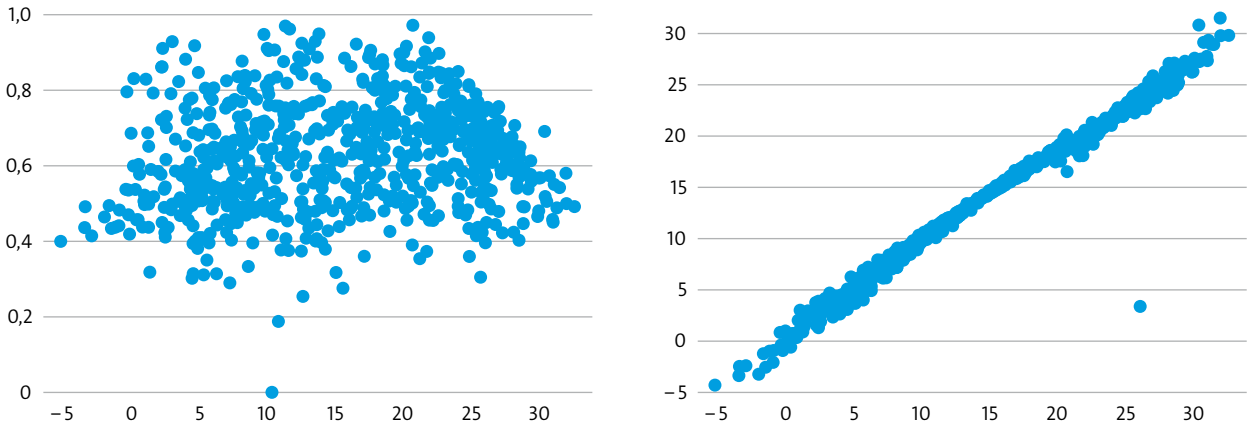


Abbildung 10: Links: Abhängigkeit der Luftfeuchtigkeit von der Temperatur im Beispiel-Datensatz.
Rechts: Abhängigkeit der gefühlten Temperatur von der wirklichen Temperatur.

3.4 Akkumulierte lokale Effekte (accumulated local effects)

Die im vorherigen Abschnitt beschriebene Technik der partiellen Abhängigkeiten hat das Problem, dass teilweise höchst unrealistische Datenpunkte zur Erklärung des Verhaltens unseres ML-Modells herangezogen werden. Beispielsweise werden Datenpunkte mit einer Temperatur von 20 °C und einer gleichzeitigen gefühlten Temperatur von 0 °C generiert, was abseits von Extremwetterlagen höchst unrealistisch ist. Die Technik der kumulierten lokalen Effekte (accumulated local effects) vermeidet dieses Problem, indem realistischere Datenpunkte generiert werden und statt globaler Abhängigkeiten nur lokale Differenzen betrachtet werden [10]. Im einfachsten Fall unterteilt man hierfür den Wertebereich eines gegebenen Attributwerts zunächst in mehrere kleine Intervalle, beispielsweise durch die Bildung von Quantilen. Anschließend werden für jedes dieser Intervalle exemplarisch Datenpunkte ausgewählt, deren Werte für das gegebene Attribut in dem Intervall liegen. Ausgehend von diesen Datenpunkten erzeugen wir neue, synthetische Datenpunkte, deren Attributwerte für das gegebene Attribut entweder auf der linken oder rechten Grenze des Intervalls liegen. Für beide Gruppen von Datenpunkten berechnen wir die Vorhersagen des ML-Modells und bilden anschließend die mittlere Differenz dieser Vorhersagen, was den sogenannten nicht-zentrierten ALE-Effekt in diesem Intervall ergibt. Anschließend summieren wir für jedes Intervall die so ermittelten Werte über das Intervall selbst sowie alle links von diesem liegenden Intervalle auf. Im letzten Schritt ziehen wir von jedem der so entstandenen kumulierten Werte den Mittelwert ab, um zum zentrierten ALE-Wert zu gelangen. Dieses etwas kompliziert anmutende Verfahren produziert ähnlich wie das PDE-Verfahren eine Abschätzung des Effekts eines Attributwerts auf die Vorhersage des ML-Modells in einem gegebenen Intervall. Anders als der PDE-Wert werden für die Berechnung des ALE-Werts aber weitaus realistischere Datenpunkte herangezogen, denn es werden nur synthetische Datenpunkte betrachtet, die nahe an den realen Daten liegen und damit Abhängigkeiten zwischen einzelnen Attributen realistischer berücksichtigen.

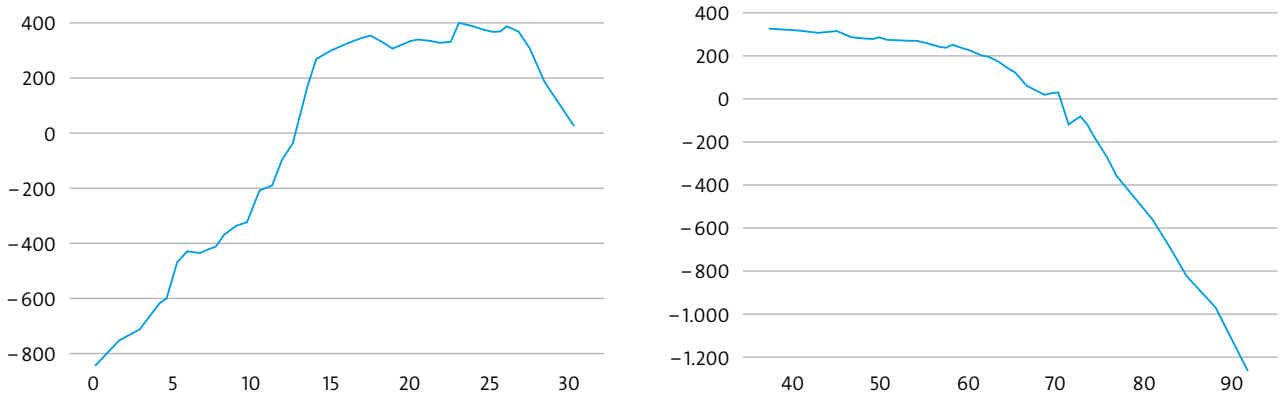
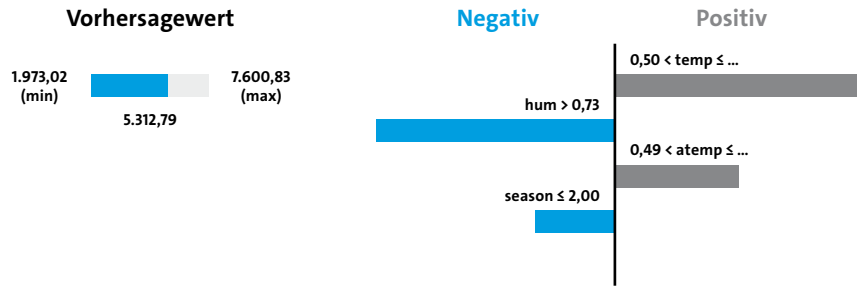


Abbildung 11: Links: Akkumulierter lokaler Effekt der Temperatur auf die Anzahl der Fahrradleihen.
Rechts: Akkumulierter Effekt der Luftfeuchtigkeit.

Abbildung 11 zeigt exemplarisch für unser Modell ALE-Diagramme für die Temperatur sowie die Luftfeuchtigkeit. Der kumulierte Effekt gibt jeweils an, welchen Einfluss der jeweilige Attributwert im gegebenen Wertebereich im Mittel auf die Vorhersage des Modells hat. Beispielsweise erhöht ein Temperaturwert von 25 °C die Anzahl der Fahrradleihen im Vergleich zur durchschnittlichen Vorhersage um ca. 250. Eine Luftfeuchtigkeit von 80 % hingegen verringert die Anzahl der Leihen im Modell um ca. 500. ALE-Werte erlauben uns damit Rückschlüsse auf den lokalen Effekt eines Attributs in einem gegebenen Wertebereich zu erhalten und hierbei auch Korrelationen mit anderen Attributen zu berücksichtigen, um unrealistische Datenpunkte auszuschließen. Genau wie PDP-Werte können ALE-Werte jedoch nur für eine geringe Anzahl an Attributen sinnvoll interpretiert werden, was eine Untersuchung des Einflusses von Kombinationen aus unterschiedlichen Attributwerten erschwert. Um eine solche Untersuchung zu vereinfachen, wurden in den vergangenen Jahren verschiedene Verfahren entwickelt, von denen wir hier LIME sowie SHAP vorstellen.

3.5 Lokale Surrogatwerte (LIME)



Merkmal	Wert
temp	0,61
hum	0,79
atemp	0,58
season	2,00
windspeed	0,15
weathersit	2,00

Abbildung 12: Erklärung einer einzelnen Modellvorhersage mithilfe des LIME Verfahrens.
Links: Vorhersagewert des Modells für die gegebenen Eingabewerte, mit Angabe des Minimum- und Maximumwertes der Modellvorhersagen. Mitte: Einfluss einzelner Attributwerte auf den Ausgabewert. Wirkliche und gefühlte Temperatur wirken sich positiv auf den Vorhersagewert aus, Luftfeuchtigkeit und Jahreszeit negativ. Rechts: Eingabewerte aller betrachteten Attribute für die gegebene Modellvorhersage.

LIME [11, 12, 13] ist ein Erklärverfahren für ML-Modelle, das entwickelt wurde, um auch komplexe und stark nichtlineare Modelle lokal erklärbar zu machen, was mit den in den vorherigen Abschnitten vorgestellten Methoden nur unzureichend möglich ist. Lokale Erklärbarkeit heißt hierbei, dass lediglich das Verhalten des Modells für einzelne Datenpunkte durch LIME erklärt wird, nicht jedoch das globale Verhalten des ML-Modells. LIME wurde von Forschern an der Universität Washington entwickelt und erfreut sich heute großer Beliebtheit, es existiert zudem eine hochwertige Implementierung für Python, die einfach mit bestehenden ML-Verfahren genutzt werden kann.

LIME generiert in mehreren Schritten ein Erklärungsmodell für einzelne Vorhersagen eines Modells. Zunächst werden um den zu erklärenden Datenpunkt herum zufällige Datenpunkte generiert oder aus den Trainingsdaten ausgewählt. Für diese Datenpunkte werden mit dem bestehenden ML-Modell Vorhersagen generiert. Diese Datenpunkte mitsamt Vorhersagen werden anschließend genutzt, um ein erklärbares Modell zu trainieren, welches die Vorhersagen des eigentlichen ML-Modells lokal erklären soll. Hierbei kann z. B. ein interpretierbares Modell wie ein Entscheidungsbaum oder eine lineare Regression genutzt werden. Die Koeffizienten dieses Modells können dann interpretiert werden, um die Entscheidung des ursprünglichen ML-Modells zu erklären. Die grundlegende Idee von LIME ist, dass auch sehr komplexe ML-Verfahren normalerweise lokal linear sind (d. h. Datenpunkte, die sich sehr ähneln, werden üblicherweise auch ähnlich klassifiziert), und diese Linearität kann ausgenutzt werden um das ML-Modell zumindest lokal erklärbar zu machen. LIME kann auf tabellarische Eingabedaten angewandt werden und bietet zusätzlich Anpassungen für spezielle Datenformate wie Bilddaten. Abbildung 12 zeigt exemplarisch die Erklärung für einen einzelnen Datenpunkt des Bike-Sharing-Datensatzes. LIME sagt aus, dass die Temperatur sowie Windgeschwindigkeit für den gegebenen Datenpunkt positiv zum Vorhersage-Ergebnis von 5.300 Fahrrad-Leihen beitrugen, wohingegen die Luftfeuchtigkeit von 73 % negativ zum Vorhersagewert beitrug. Um LIME anzuwenden, müssen verschiedene Parameter gewählt werden, was die Erklärungen teilweise nicht einfach reproduzierbar macht. Um diese Nachteile zu vermeiden, wurde das SHAP-Verfahren entwickelt, welches LIME mit sogenannten Shapley-Werten kombiniert, um noch bessere und reproduzierbare Erklärungen für Modellvorhersagen zu generieren.

3.6 SHAP

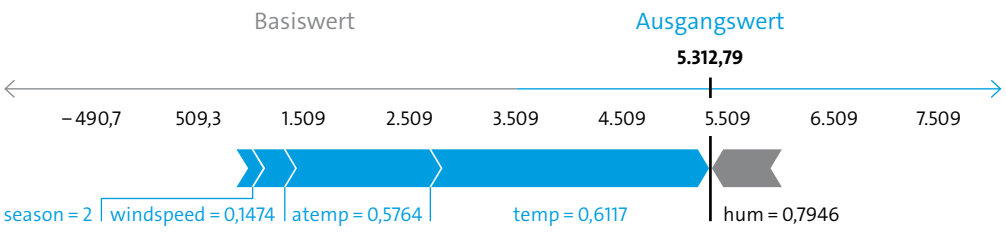


Abbildung 13: Erklärung einer einzelnen Modellvorhersage mithilfe des SHAP Verfahrens. Analog zu LIME wird der Effekt einzelner Attributwerte auf den Vorhersagewert des Modells für einen gegebenen Datenpunkt visualisiert. Wirkliche und gefühlte Temperatur, Windgeschwindigkeit und Jahreszeit haben einen positiven Einfluss, die Luftfeuchtigkeit hat einen negativen Einfluss auf den Vorhersagewert. Unten: Werte der einzelnen Attribute für den gegebenen Datenpunkt.

SHAP [14, 15, 16] ist ein weiteres Verfahren für die lokale Erklärung von ML-Modellen, welches eine Weiterentwicklung verschiedener Konzepte darstellt und u. a. LIME sowie Shapley-Werte kombiniert, um robustere Erklärungen für Vorhersagen von ML-Modellen zu liefern. Genau wie LIME kann SHAP auf generische, tabellenbasierte Daten angewandt werden und bietet zu-

sätzlich spezifische Implementierungen für Datenformate wie Bilddaten. SHAP generiert ähnlich zu LIME ein interpretierbares, lokales Modell eines ML-Verfahrens, welches anschließend zur Erklärung von Vorhersagen des Modells genutzt werden kann. Im Gegensatz zu LIME ist es bei SHAP jedoch nicht nötig, Hyperparameter für das lokale Modell manuell festzulegen, dementsprechend sind die Erklärungen in vielen Fällen robuster und hängen nicht von der Parameterwahl des Benutzers ab.

SHAP ist wie LIME u. a. als Python-Bibliothek implementiert und kann somit leicht genutzt werden. Abbildung 13 zeigt exemplarisch eine Erklärung des Datenpunkts aus dem vorherigen Abschnitt, welche mit der Python-Implementierung von SHAP generiert wurde. Im Gegensatz zu LIME stellt SHAP Erklärungen hier als sogenannte »Force Plots« dar: Für jedes untersuchte Attribut zeigt dieser Force-Plot, wie der Vorhersagewert des Modells durch den gegebenen Attributwert beeinflusst wurde. Ähnlich zu LIME ordnet auch SHAP der Temperatur und der Windgeschwindigkeit einen positiven Einfluss und der Luftfeuchtigkeit einen negativen Einfluss auf die getroffene Vorhersage des Modells zu.

3.7 Grenzen der Erklärbarkeit

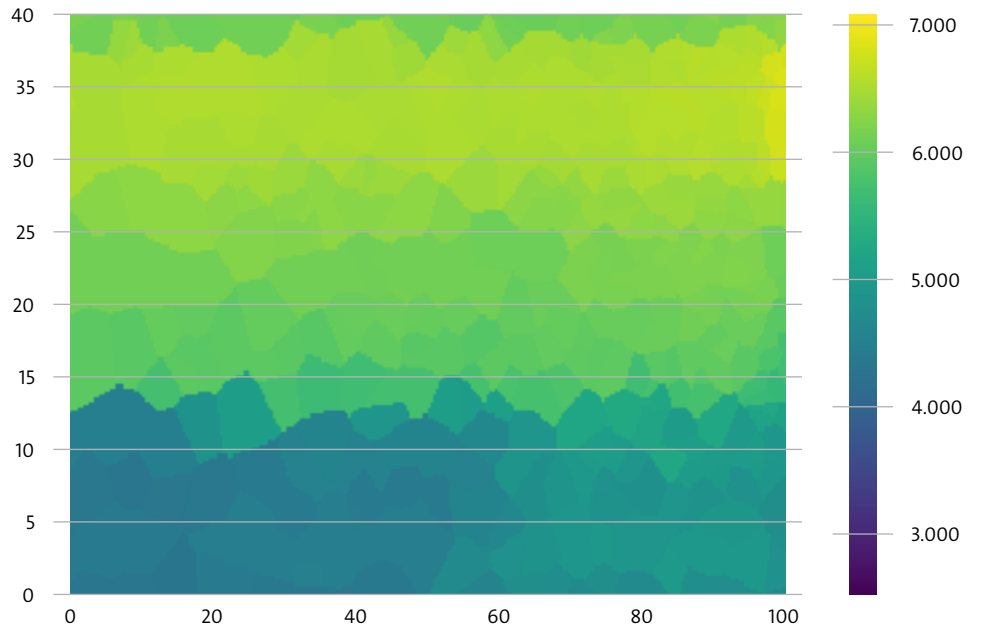


Abbildung 14: Vom ML-Modell vorhergesagte Anzahl an Fahrradleihen für synthetische Datenpunkte, bei denen ausgehend von einem spezifischen Datenpunkt die Attributwerte von Temperatur und Luftfeuchtigkeit variiert wurden.

Die in den vorherigen Abschnitten diskutierten Möglichkeiten, ML-Verfahren zu erklären, können Vorhersagen lediglich lokal nachvollziehbar machen und keine globale Erklärung für alle Vorhersagen liefern. Dies ist nicht überraschend, da Komplexität und Nichtlinearität in vielen Fällen notwendige Eigenschaften von ML-Modellen sind, welche diesen erst ermöglichen, für eine große Bandbreite von Datenpunkten effektiv Vorhersagen zu liefern. Abbildung 14 illustriert dies anhand des für den Beispieldatensatz generierten ML-Modells: Dargestellt werden die Vorhersagen des Modells für synthetische Datenpunkte, bei denen ausgehend von einem spezifischen Datenpunkt die Attributwerte von Temperatur und Luftfeuchtigkeit variiert wurden. Die farbliche Kodierung beschreibt hierbei den Vorhersagewert des Modells. Man erkennt, dass sich das Modell in einzelnen Bereichen annähernd linear verhält, aber auch teilweise eine hohe Nichtlinearität aufweist. Wie durch SHAP und LIME demonstriert, kann ein globales ML-Modell dabei als Kombination aus einer Vielzahl an annähernd linearen, lokalen Modellen interpretiert werden, welche für jeweils sehr kleine Ausschnitte der Daten Vorhersagen treffen. Die Mächtigkeit moderner ML-Verfahren wie z. B. Deep Learning liegt hierbei in der Komplexität und Kapazität ihres Parameterraums, der bei großen Modellen mehrere hunderte Millionen Parameter umfassen kann. Dies erlaubt solchen Modellen eine extrem große Zahl unterschiedlichster Datenpunkte zu verstehen und gleichzeitig Aussagen von Trainingsdaten auf unbekannte Datenpunkte zu verallgemeinern. Die große Anzahl an Parametern ist damit maßgeblich für den Erfolg solcher Verfahren, reduziert aber gleichzeitig die Erklärbarkeit. Mit dem stärkeren Trend zu solchen Verfahren wird es daher schwierig bis unmöglich werden, globale Erklärungen zu generieren, welche die Vorhersagen der Modelle für einen Großteil der möglichen Eingabedaten einheitlich erklären können. Lokale Erklärungen wie sie von LIME, SHAP oder ALE generiert werden, liefern daher zumindest eine begrenzte Möglichkeit, Entscheidungen solcher Verfahren nachvollziehbar zu machen, wenngleich die Erklärungen nur für einen sehr begrenzten Wertebereich des Modells gültig sind. Die Entwicklung neuer Verfahren zur Erklärung von ML-Modellen ist hierbei ein aktiver Bereich der Forschung, dem sich eine Vielzahl an Forschern weltweit widmen. Es ist daher zu hoffen, dass gleichzeitig mit der stärkeren Verbreitung von komplexen ML-Verfahren auch die Ansätze zu deren Erklärung stetig besser und einfacher anwendbar werden (siehe z. B. [18] für eine aktuelle Weiterentwicklung von LIME). Mit Open-Source-Projekten wie Algoneer hoffen wir, hierfür einen Beitrag leisten zu können, indem wir die Anwendung verschiedener Erklärverfahren in der Praxis stark vereinfachen und zudem die Interpretation dieser Erklärungen für den Anwender einfacher gestalten. Eine erste Version der Software-Bibliothek ist bereits online verfügbar [4, 5].

3.8 Literaturverzeichnis

[1] H20 – Open-Source Bibliothek zur Erklärung von ML-Verfahren ↗ https://github.com/h2oai/mli-resources
[2] IBM AI-Fairness 230 Toolkit: ↗ https://github.com/IBM/AIF360
[3] Alibi – Bibliothek für Nachvollziehbarkeit von ML-Verfahren von seldon.io: ↗ https://github.com/SeldonIO/alibi
[4] Algoneer – Webseite ↗ https://algoneer.org
[5] Algoneer – Open-Source Software ↗ https://github.com/algoneer
[6] Christoph Molnar, Interpretable Machine Learning – A Guide for Making Black Box Models Explainable. Selbstpubliziert/Gitbooks (2019) ↗ https://christophm.github.io/interpretable-ml-book/
[7] Bike-Sharing Datensatz der Universität Porto ↗ https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset
[8] Apley, Daniel W. »Visualizing the effects of predictor variables in black box supervised learning models.« arXiv preprint arXiv:1612.08468 (2016).
[9] Friedman, Jerome H. »Greedy function approximation: A gradient boosting machine.« Annals of statistics (2001): 1189–1232.
[10] Zhao, Qingyuan, and Trevor Hastie. »Causal interpretations of black-box models.« Journal of Business & Economic Statistics, to appear. (2017).
[11] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. »Why should I trust you?: Explaining the predictions of any classifier.« Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM (2016).
[12] LIME Python-Bibliothek ↗ https://github.com/marcotcr/lime
[13] Algoneer – LIME Beispiel – Jupyter Notebook ↗ https://github.com/algoneer/algoneer/blob/master/examples/bike-sharing/third-party/lime.ipynb
[14] Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. Scott M. Lundberg, Bala Nair, Monica S. Vavilala, Mayumi Horibe, Michael J. Eisses, Trevor Adams, David E. Liston, Daniel King-Wai Low, Shu-Fang Newman, Jerry Kim & Su-In Lee
[15] Consistent Individualized Feature Attribution for Tree Ensembles. Scott M. Lundberg, Gabriel G. Erion, Su-In Lee
[16] SHAP Python-Bibliothek ↗ https://github.com/slundberg/shap
[17] Wachter, Sandra, Brent Mittelstadt, and Chris Russell. »Counterfactual explanations without opening the black box: Automated decisions and the GDPR.« (2017).
[18] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. »Anchors: High-precision model-agnostic explanations.« AAAI Conference on Artificial Intelligence (2018).

4 Interpretierbare Verifizierung von Autorschaft

4 Interpretierbare Verifizierung von Autorschaft

Lukas Graner, Martin Steinebach

4.1 Einleitung

In diesem Kapitel soll zuerst ein Beispiel für die Notwendigkeit der Interpretierbarkeit von Ergebnissen des maschinellen Lernens vorgestellt werden. Danach stellen wir einen Ansatz zum Erreichen dieser Interpretierbarkeit vor. Auf technischer Ebene beschäftigen wir uns dabei mit der Autorschaftsverifikation. Ein Einsatzszenario ist die Prüfung einer Autorenschaft in einer Gerichtsverhandlung. Hier genügt es nicht, ein Ergebnis der Analyse vorzulegen. Es muss auch nachvollziehbar sein, wie dieses Ergebnis erreicht wurde und wie zuverlässig es ist. Der Anwender des Verfahrens muss dem Richter potenziell als Gutachter eine Erklärung der Ergebnisse liefern und diese gegen Einwände der Gegenseite verteidigen. Dies ist nur möglich, wenn über ein abstraktes Ergebnis im Sinne einer prozentualen Übereinstimmung auch eine nachvollziehbare und detaillierte Darstellung geliefert werden kann, wie dieses Ergebnis zustande kam.

Als Beispiel, wie Analysen durch erklärende und interpretierbare Ergebnisse unterstützt werden können, soll hierbei der Vaterschaftstest dienen, der ebenfalls vor Gericht erörtert werden kann. Hierbei werden unter streng geregelten Umständen DNA-Proben zweier Personen entnommen und miteinander verglichen. Der Vergleich betrachtet dabei Marker in der DNA und leitet davon Muster ab. Da die Muster von beiden Elternteilen vererbt werden, müssen Teilbereiche der Muster jeweils bei Elternteil und Kind identisch sein. Das Ergebnis eines Vergleichs basiert dementsprechend auf einer Messung übereinstimmender Mustersequenzen und kann nur eine statistische Aussage über die Wahrscheinlichkeit der Vaterschaft geben. Diese ist allerdings oft weit über eine Fehlerrate von einem Promille genau. Vor Gericht kann nicht nur die Wahrscheinlichkeit angegeben werden, sondern auch eine visuelle Darstellung der DNA-Ähnlichkeiten von Kind und Eltern, eine Darstellung der Häufigkeit der gefundenen Muster, die auf eine Vaterschaft hinweisen sowie die Formel zur Herleitung der Wahrscheinlichkeit. Dementsprechend wird dem Verfahren großes Vertrauen entgegengebracht.

In den folgenden Abschnitten soll nun erörtert werden, wie auch ein Ergebnis, welches auf Basis von maschinellem Lernen gewonnen wurde, in einer vergleichbaren Klarheit vor Gericht verteidigt werden kann. Unser Verfahren ermöglicht es, die Autorschaft eines Dokumentes einer Person zuzuordnen, wenn eines oder mehrere Dokumente, die nachweislich von dieser Person stammen, als Referenz vorliegen. In der Praxis kann es sich hierbei um eine Beweisführung bezüglich des Verfassens von beispielsweise Drohbriefen, Bekennerschreiben oder Lösegeldforderungen handeln. Der Nachweis der Autorschaft hat hier jeweils signifikante Konsequenzen und muss daher vor Gericht eindeutig nachvollziehbar sein. Da es sich hierbei um eine technische Diskussion handelt, in der besonders die Nachvollziehbarkeit detailliert betrachtet

werden soll, gehen wir in allen übrigen Punkten von einem idealen Szenario aus: Es liegen Dokumente vor, die nachweisbar vom vermuteten Autor stammen und ein zu prüfendes Dokument ist nachweisbar unverändert.

4.2 Autorschaftsverifikation

Die Disziplin der Autorschaftsverifikation (nachfolgend »AV« genannt) beschäftigt sich fundamental mit der Fragestellung, ob zwei gegebene Dokumente von ein und demselben Autor verfasst wurden. Hierbei zeichnet sich die AV durch den zentralen Aspekt aus, dass die Antwort auf diese Fragestellung ausschließlich durch eine Analyse auf Textebene geschieht. Metadaten wie etwa ein in einer Datei hinterlegter Autorenname oder auch visuelle Eigenschaften wie (Hand-)Schriftart fließen somit nicht in die Analyse mit ein. Stattdessen werden stilistische Merkmale in Betracht genommen, zum Beispiel Verwendungen von bestimmten Worten oder Phrasen, wobei hierbei unter anderem die Schwierigkeit besteht, Schreibstil von Thematik zu unterscheiden, da letztere nicht individuell abhängig vom Autoren ist. Die AV wird zumeist in der Forensik und bei gerichtlichen Verfahrensfällen eingesetzt. Mit der zunehmenden Digitalisierung in der heutigen Zeit etablieren sich allerdings in verwandten Feldern wie der Computerlinguistik immer neue Anwendungsgebiete. So kann AV zum Beispiel im Bereich kommerzieller Onlineplattformen genutzt werden, um multiple Benutzer-Accounts beziehungsweise Fake-Accounts aufzuspüren. Ebenfalls können Übersetzungsservices von AV profitieren, etwa um automatisiert zu bewerten zu welchem Grad die übersetzten Texte den Schreibstil der ursprünglichen Texte beibehalten.

4.3 Umfeld

Wo früher noch händische Analysen gemacht wurden (vgl. mit Autorschaftsanalyse der unter einem Pseudonym verfassten »Federalist Papers« in den 1960er Jahren [1]), werden heute automatisierte Verfahren vorgestellt, die insbesondere Maschinelles Lernen mit einbeziehen und vielversprechende Ergebnisse bei der Erkennung der Autorschaft erzielen. Dabei werden im Grunde zwischen drei Disziplinen mit verschiedenen Annahmen und Voraussetzungen unterschieden, namentlich die Autorschaftsverifikation, -attribution und das -clustering. Beim Clustering geht man von mehreren unbekannten Dokumenten aus, die nach ihren zugehörigen Autoren gruppiert (geclustert) werden sollen, sodass sich für jeden Autor eine Gruppe aus von ihm/ihr geschriebenen Dokumenten ergibt. Bei der Attribution wird eine Reihe an Dokumenten sowie die Kenntnis über deren Autorschaft vorausgesetzt. Ausgehend von einem unbekannten Dokument ist nun zu überprüfen, von welchem der bekannten Autoren dieses verfasst wurde. Dies reflektiert einen »Closed-Set«-Ansatz, wo aus der Autorenmenge immer einer als Antwort gewählt wird, auch wenn es in Wirklichkeit keiner derer war. Dies ist problematisch, wenn nicht von vornherein ausgeschlossen werden kann, dass kein anderer, noch unbekannter Autor der wahre Autor ist. Um sich von dieser Problematik zu lösen, muss ein »Open-Set«-Ansatz also die weitere Antwortmöglichkeit »ein Autor außerhalb der Kandidatenmenge« in Betracht

ziehen. Im Extremfall ist dabei nur ein Kandidat gegeben, sodass sich dadurch die Fragestellung ergibt: »Hat der Kandidat das Dokument verfasst oder nicht?«. Diese Fragestellung spiegelt nun das Wesen der AV wider.

4.4 Verfahren

In der Abteilung Media Security und IT Forensics des Fraunhofer SIT haben wir ein vielversprechendes AV-Verfahren entwickelt, welches sowohl wettbewerbsfähige Erkennungsgenauigkeiten erzielt, gleichzeitig aber auch mehrere Ansätze zur Nachvollziehbarkeit der gemachten Entscheidungen bereitstellt. In diesem Abschnitt wird dieses Verfahren genauer beschrieben.

Den zentralen Aspekt unseres Verfahrens stellt ein Neuronales Netz dar. Wir verwenden hierbei ein sogenanntes Siamesisches Netz. Es enthält zwei Subnetze, die jeweils einen Text in einen Stil abbildenden Vektor transformieren. Die resultierenden beiden Vektoren werden anschließend über ein Ähnlichkeitsmaß verglichen, sodass abhängig von der errechneten Ähnlichkeit und einem erlernten Schwellenwert entschieden wird, ob es sich um gleiche Autoren bei den Eingangstexten handelt oder nicht. Die Subnetze sind dabei als sogenannte Convolutional Neural Nets (CNN) konstruiert. Wie diese Texte zu Stilvektoren transformieren, wird im Folgenden näher erklärt.

Da Neuronale Netze auf numerischen Daten operieren, sind sie nicht in der Lage ohne weiteres mit Texten umzugehen. Daher werden diese zuerst als Sequenzen von Texteinheiten (also Wörter, Satzzeichen, etc.) betrachtet. Anschließend wird jede bekannte Texteinheit auf einen Vektor (also Punkt in einem Raum) abgebildet. Dieses Verfahren wird als Embedding bezeichnet und hat die Eigenschaft, dass ein Neuronales Netz selbst lernt, wie es die Einheiten anordnet. Am Ende liegen dabei ähnliche Texteinheiten nah beieinander, wie etwa die Wörter *»Deshalb«, »Deswegen«* und *»Somit«*.

Nach dem Embedding liegt ein Text also in Form einer Vektoren-Sequenz vor. Auf diesen kann nun das CNN operieren, welches im Grunde bestimmte Muster ausfindig macht. Solche Muster repräsentieren dann meist häufig wiederkehrende zusammenhängende Wortsequenzen und – da wir uns im Kontext von AV befinden – werden vom CNN selektiv so gelernt, dass sie stilistische Merkmale reflektieren. Wird ein neuer Text untersucht, so sucht das CNN darin nach den gelernten Merkmalen und zeichnet auf ob und wie häufig diese auftreten. Diese Aufzeichnung wird schließlich dann in Form eines Stil-Vektors ausgegeben, sodass jeder Wert darin ein stilistisches Merkmal reflektiert. Werden also zwei Texte mit einem ähnlichen stilistischen Spektrum durch das CNN transformiert, so sollten sich auch ähnliche Stilvektoren daraus ergeben, welche dann durch das Ähnlichkeitsmaß des Siamesischen Netzes verglichen werden können. Hierbei ist anzumerken, dass unser Verfahren aufgrund seiner siamesischen Architektur keine individuellen Stilmerkmale von Autoren erlernt, sondern solche, die zu einer generell guten Unterscheidung zwischen zwei Autoren verhelfen.

4.5 Interpretierbarkeit

Existierenden AV-Verfahren, darunter auch dem State of the Art, fehlen ausreichende Konzepte zur Nachvollziehbarkeit. Damit solche Verfahren also bei Gericht Anwendung finden können, müssen Interpretierbarkeit und Erklärungen der Ergebnisse ermöglicht werden, wie etwa beim Vaterschaftstest. Ein wesentlicher Bestandteil von Interpretierbarkeit bei maschinellem Lernen ist neben der Transparenz, also der Kenntnis von Architektur und Lernalgorithmen, das Bereitstellen von sogenannten »Post-hoc«-Erklärungen [2]. Diese versuchen auf visueller, textueller oder beispielgebender Basis verständlich zu machen, wie Entscheidungen auf getesteten Daten zustande gekommen sind. Im Kontext unseres AV-Verfahrens soll also anhand der zwei zu überprüfenden Dokumente erklärt werden, warum sich das Verfahren für oder gegen eine übereinstimmende Autorschaft entschieden hat.

Für die »Post-hoc«-Erklärung unseres AV-Verfahrens ziehen wir mehrere Ansätze heran. Zum einen sind wir daran interessiert, wie die resultierenden Stil-Vektoren von Texten zu vergleichen sind. Mithilfe von Techniken zur Dimensionsreduzierung können wir diese visuell darstellen und vergleichen.

Zum anderen ist es essenziell, zu ermitteln, welche Stilmerkmale, also textuelle Muster, überhaupt vom CNN betrachtet wurden. Diese können dann anschließend in den ursprünglichen Dokumenten markiert werden. Somit lässt sich zeigen, wo ein Muster vom neuronalen Netz gefunden wurde und welche genauen Texteinheiten dazu beigetragen haben. Als Grundlage dafür verwenden wir Aktivierungswerte des CNNs, sowie sogenannte Saliency [3] und Class Activation Maps [4].

4.6 Praxisbeispiel

In diesem Abschnitt beschreiben wir den praktischen Einsatz unseres Verfahrens. Wir beschreiben dazu erst das Material, mit dem getestet wurde, und dann die Ergebnisse der Durchläufe.

Um unser Verfahren sowohl auf Performanzmaße wie Genauigkeit, als auch unsere vorgestellten Interpretierungsmöglichkeiten zu testen und zu evaluieren, benötigen wir Datensätze an Texten mit Wissen über die echten Autoren. Für die folgenden Experimente haben wir einen AV-Datensatz konstruiert, der mehrere AV-Fälle enthält. Diese bestehen wiederum aus mehreren Dokumenten eines bekannten Autors, sowie aus einem zu überprüfenden Dokument. Der Datensatz wurde zusammengestellt aus E-Mails, die von Mitarbeitern des Enron-Konzerns versendet wurden und im Rahmen einer Ermittlung öffentlich zugänglich gemacht wurden [5].

Nun wird das vorgestellte AV-Verfahren auf den beschriebenen Datensatz angewendet. Dazu durchläuft es zunächst jeweils eine Trainingsphase, sodass das unterliegende Neuronale Netz auf einem getrennten Teil des Datensatzes lernt. Anschließend wird pro AV-Fall im Test-Set eine Übereinstimmung oder Nicht-Übereinstimmung der Autorschaft von den trainierten Modellen vorhergesagt. Danach betrachten wir die Ergebnisse der Verifikation unter dem Aspekt

der Interpretierbarkeit. Diese ist insbesondere deshalb von Bedeutung, da sich zeigt, dass allgemein die Fehlerrate bei der AV noch hoch ist. So liegen AV-Verfahren, darunter auch der State of the Art, bei unserem Datensatz in über 20 % der Fälle falsch [6].

Im Folgenden stellen wir daher exemplarisch Ergebnisse unserer »Post-hoc«-Erklärungen vor. Abbildung 15 zeigt entsprechend unseres ersten Interpretierungsansatzes die Stilvektoren zweier exemplarischer AV-Fälle, einer mit (»J-Fall«) und einer ohne übereinstimmende Autorschaft (»N-Fall«). Dabei wird deutlich, dass beim N-Fall die Vektoren der einzelnen Dokumente des bekannten Autors (blaue Kreise) eine zusammenhängende Ansammlung bilden, während das zu überprüfende Dokument abseits liegt. Im J-Fall hingegen liegen alle Dokumente nah beieinander, was somit auf eine übereinstimmende Autorschaft hindeutet.

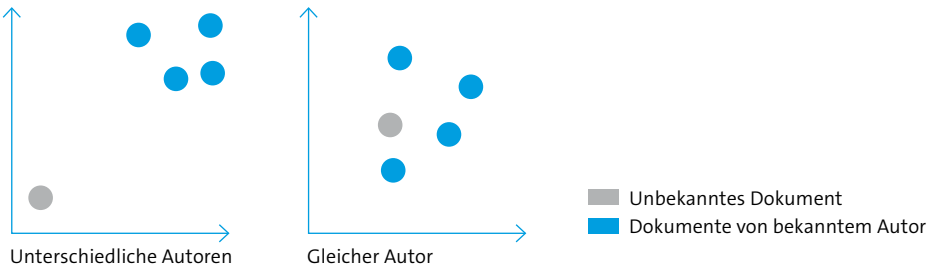


Abbildung 15: Die Stilvektoren zweier AV-Fälle, visualisiert mithilfe der Dimensionsreduzierungtechnik t-SNE [7]. In beiden Fällen bilden die Vektoren der Dokumente der jeweiligen bekannten Autoren (blau) eine dichte Gruppierung. Das unbekannte Dokument (grau) des linken Falls wurde von einem anderen Autor verfasst und liegt entsprechend distanziert zu der Gruppierung des anderen Autors. Im rechten Fall dagegen handelt es sich um ein Dokument des bekannten Autoren und mischt sich dementsprechend unter die Gruppierung.

Für die beiden Fälle aus Abbildung 15 betrachten wir nun, welche Muster für die Entscheidung des Neuronalen Netzes am relevantesten waren. Mithilfe der Saliency und Class Activation maps ermitteln wir zunächst die Textmuster, die durch die Stilvektoren überhaupt abgebildet werden. Im Modell des Enron-Datensatzes sind es unter anderem folgende (da über hundert Muster erlernt wurden, nennen wir hier nur ein paar Beispiele, die aus unserer Sicht interessante Funde darstellen): *», and«* (ein Komma gefolgt von dem Wort *»and«*, dieses markante Muster bei einer Aufzählung wird auch als Oxford-Komma bezeichnet [8]), *»i * [this, that, those, these]«* (das Wort *»i«* gefolgt von einem beliebigen Wort und schließlich gefolgt von einem Demonstrativpronomen), oder *»please let me know«* (eine Phrase, die bestimmte Autoren häufig in ihren Emails nutzen, andere jedoch gar nicht).

In Abbildung 16 wird der Entscheidungsprozess eines Falls des Enron-Datensatzes visuell dargestellt. Jedes Muster des Stilvektors wird hierbei als ein Teil der farbigen Balken dargestellt, wobei die Breite die Relevanz reflektiert. Blau hinterlegte Merkmale (wie beispielsweise *»that this is«*) haben für die Entscheidung für eine übereinstimmende Autorschaft beigetragen, etwa, wenn diese ähnlich häufig verwendet wurden in den Dokumenten. Grau hinterlegt sind hingegen diese, die dagegensprechen. In diesem Falle überwiegen die blauen Merkmale, sodass die Entscheidung rechts von dem Ausgangspunkt, also zugunsten von übereinstimmender Autorschaft, getroffen wurde. Je weiter entfernt vom Ausgangspunkt, also dem Schwellwert θ, die Entscheidung liegt, desto sicherer ist sich das Verfahren.

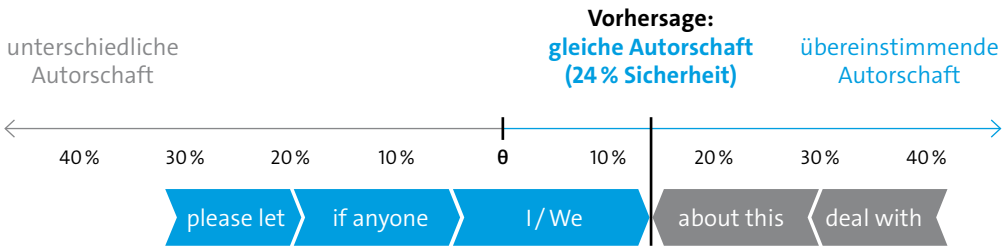


Abbildung 16: Der Entscheidungsprozess für ein AV-Fall verdeutlicht als Zusammenspiel aus mehreren Merkmalen. Merkmale beeinflussen die finale Entscheidung ausgehend vom Schwellenwert θ entweder in Richtung übereinstimmender Autorschaft (blau, nach rechts) oder unterschiedliche Autorschaft (grau, nach links). Sie sind dabei abhängig ihrer Relevanz skaliert, sodass relevante Merkmale mehr zur Entscheidung beitragen und visuell hervorgehoben sind.

Mithilfe der Aktivierungswerte des CNNs können nun auch die einzelnen Merkmale in den Texten markiert werden. So wird direkt in den ursprünglichen Texten visuell erklärt, welche Stellen relevant für die Entscheidung des Verfahrens waren. Ein entsprechender Ausschnitt aus dem Fall aus Abbildung 16 wird in Abbildung 17 gezeigt. Hier sind die relevantesten Muster farbig im Text markiert. Es ist klar zu erkennen, dass deutlich mehr Merkmale für eine übereinstimmende Autorschaft sprechen (blau). Abbildung 18 zeigt wiederum einen N-Fall, bei dem stattdessen besonders die Muster herausstechen, die gegen eine gleiche Autorschaft sprechen (grau).

Das unbekannte Dokument

Rick Shapiro has offered to present. I think that we can work up some presentation for 45 minutes. [...] Please let me know if you can do this and if there are any issues. Sorry about all of this confusion regarding residential customers. I intend to take vacation this Friday, all of next week, and Dec 31. Unfortunately, Harry Kingerski, Bob Frank, Jeff Dasovich,and Christi Nicolay will also be on vacation. Does everyone have access to the O Drive / Public Affairs directory? If you don't have access, please let me know. [...] but a Trust pays the bills. I told him that we would look into this and get back by end of business Thursday. [...] not with their respective areas. I will ask Linda Noske to set up this meeting. If anyone hears anything else about the market delay, please let Jean Ryall know immediately. As of right now, it is my impression that we all agree there is no delay scheduled? Fix Capacity Auction process, including credit matters. It would be good to make sure that we have a running tally. We need all of our customers to call a the Governor, b the key Legislative members, and c their trade associations. We can release very quietly to influential persons Wal-Mart, Raytheon, and Northrop. It is also critical that we try and contact someone from Boeing to get their read of the bill and its impact.

Ein Dokument des bekannten Autors

Agreeing to that, the idea is to review the Bingaman language and see if our groups could agree to that language. I think that internally Enron is already there? John Shelk, if you could pls send out the language to everyone on this e-mail to get agreement that would be very helpful. If anyone has any problems, please let me know. Everyone should be working to get customers focused on messaging to the Governor's office and b trade associations. Once I receive the list, we'll make a decision on how to proceed with that group. Hedy Bev, who should support the amendment? Don't we need a member to carry the provision? If not, please let me know – Joe sent me a copy. He is interested in any analysis that we may have provided related to the Simon Properties deal with the Master Meter issue. If anyone has any old files on this, please send his way. We probably need to have a review of our rights and obligations so that we can properly book the deal. After we get Joe our info, we can regroup to determine next steps. If anyone has any questions, please call. Jim Joe – I forwarded the call from Phil to Bill Bradford and let Kevin Presto know. If you hear anything else, pls let me know. Also, he would like us to let our friends in the industry know about this upcoming meeting. [...]

Abbildung 17: Ausschnitte zweier Dokumente eines AV-Fall mit übereinstimmender Autorschaft. Eine Reihe an markanten Stilmerkmalen werden in beiden Dokumenten auffällig ähnlich häufig genutzt. (blau), wie etwa Satzanfänge mit den Personalpronomen »I« und »We« oder das spezifische Wort-Bigramm »that we«. Auch wenn es Muster gibt, wie »about this«, die so unterschiedlich häufig genutzt werden und somit zwei Autoren andeuten (grau), sind diese deutlich in der Minderzahl. Es wird somit eine übereinstimmende Autorschaft vorhergesagt.

Das unbekannte Dokument

Please let me know your thoughts on this. Who is going to put our presentation together? I will give it if you and Bob feel that is the right thing to do but I just do not have time to put the presentation together. That is time consuming and good for you. If you work out it is ok to go shop and put it on the credit card. She is taking your car in today to be fixed. By the way, there is 50 at Mom's that is for you because David flew home from here last night, and we had to order [...]. I'm glad to have a light week, but it leaves for a lot of boredom and down time. Well, I'll talk to you sometime this week and have fun tonight. I will be out of town Wednesday thru Friday so just let Cindy know if anyone can play. I have not felt well since I gave blood on Friday. Dad I think the best way to handle this is for you to get with Mary Joyce and her team and work out the details. I think the last item may be of some concern. If there are issues that can't be resolved between Mary and you I will gladly get involved. I will do whatever you would like to do. Dad I wish I could be there to see her face too. Sorry I will be unable to see you this weekend. It is not regulation 101, it is very complex and leading edge, but get the risks and rewards aligned much better for shareholdes and customers. Doesn't fit Enron's view of dereg, but Oregon [...]

Ein Dokument des bekannten Autors

The problem was exacerbated by a heat wave across the West, which forced California to compete with other states for scarce electricity, he said. But power generators have complained that the price caps [...]. Hidalgo said the state avoided blackouts only because of last-minute import s from the Bonne 07ville Power Administration, the federal agency that markets government-produced hydroelectric power in the Pacific Northwest. The state went into a Stage 2 power alert, the next-to-last level before blackouts are ordered. The alert was canceled in late afternoon. The blackouts would have been the first in California since May 8. The price fluctuates and is tied to the production costs of the least-efficient plant operating in California during a »power alert« declared by the Independent System Operator, which runs the state's power-transmission grid. When there's no alert, prices can't exceed 85 percent of the cap that was established during the latest alert. Until Monday, the maximum price held steady at about 101 a megawatt-hour in California. Because of a steep drop in the price of natural gas, which fuels many California power plants, suppliers knew the cap would fall. The ceiling for California fell to about 77 at 3 p. But out-of-state suppliers can withhold supplies, and on Monday it was the out-of-staters that were pulling back, Hidalgo said. The Bee's Dale Kasler can be reached at [...]

Abbildung 18: Ausschnitte zweier Dokumente eines AV-Fall mit nicht übereinstimmender Autorschaft. Jedes der beiden Dokumente enthält eine Reihe an markanten Stilmerkmalen, die nur individuell häufig genutzt werden, nicht aber von dem anderen (grau). Dies sind beispielsweise im linken Dokument die Konjunktion »but« nach einem Komma oder im rechten Dokument der Satzanfang mit »The«. Da nur sehr wenige markante Merkmale übereinstimmend sind, wird keine übereinstimmende Autorschaft vorhergesagt.

4.7 Zusammenfassung

Die Verifikation der Autorschaft ist eine Disziplin in NLP und maschinellem Lernen, welche unter anderem in gerichtlichen Gutachten eingesetzt wird. Dies erfordert eine Darstellung der Ergebnisse, die auch für Laien intuitiv nachvollziehbar ist. Wir stellen daher eine Methode der Autorschaftsverifikation auf Basis neuronaler Netze vor, welche insbesondere darauf ausgelegt sind, interpretierbare Ergebnisse zu liefern. Die Merkmale, die besonders stark für oder gegen eine Autorschaft sprechen, werden herausgearbeitet und im Text farblich hervorgehoben.
Die Darstellung zeigt also insbesondere die Stellen im Text, die sehr typisch oder untypisch für eine Autorschaft sind und liefert so einen schnell erfassbaren Anhaltspunkt, wie wahrscheinlich eine Übereinstimmung der Autoren in den zu überprüfenden Texten zu den Referenzen zu finden sind.

4.8 Literaturverzeichnis

[1] Mosteller, Wallace (1963), »Inference in an authorship problem«.
[2] Lipton (2016), »The mythos of model interpretability«.
[3] Simonyan, Vedaldi, Zisserman (2014), »Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps«.
[4] Selvaraju, Cogswell, Das, Vedantam, Parikh, Batra (2017), »Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization«.
[5] Der Datensatz »Enron Email Dataset« ist frei verfügbar auf ↗ **https://www.cs.cmu.edu/~enron/**.
[6] Halvani, Graner (2018) »Rethinking the Evaluation Methodology of Authorship Verification Methods«.
[7] Maaten, Hinton (2008), »Visualizing data using t-SNE«.
[8] Ritter (2005), »New Hart's Rules: The handbook of style for writers and editors«.

5 Adversarial AI: Wie können wir Gefahren für KI-Anwendungen durch feindliche Angriffe erkennen und ihnen entgegenwirken?

5 Adversarial AI: Wie können wir Gefahren für KI-Anwendungen durch feindliche Angriffe erkennen und ihnen entgegenwirken?

Horst Stein, Sebastian Fischer, Claudia Pohlink

5.1 Einleitung

Die Entwicklung zum Einsatz von Künstlicher Intelligenz (KI) in vielen Bereichen unseres täglichen Lebens ist unaufhaltbar. Angefangen von Sprachassistenten im Wohnzimmer bis hin zum Einsatz von KI in der Diagnose von Krankheiten oder der Steuerung komplexer industrieller Anlagen: KI trägt dazu bei, unser Leben komfortabler, sicherer und einfacher zu gestalten. Eine wesentliche Voraussetzung dabei ist jedoch, dass KI im Einsatz auch sicher ist und diese mächtige Technologie sich an ethischen Maßstäben orientiert. Dazu trägt Transparenz und Nachvollziehbarkeit der Vorgehensweisen und Entscheidungen durch KI maßgeblich bei.

Die aktuelle Diskussion um Verantwortung und Sicherheit von KI konzentriert sich hauptsächlich auf ethische Prinzipien wie Diversität, Vermeidung von Diskriminierung, Datenschutz und Fairness. Ursachen von Diskriminierung können z. B. Verzerrungen (Bias) in den Trainingsdaten sein, durch die dann Vorurteile in datengetriebenen Entscheidungssituationen bis hin zum Effekt der systematischen Benachteiligung bestimmter gesellschaftlicher Gruppen entstehen können ([1]). Zusätzlich zu diesen – sehr wichtigen – Fragestellungen gibt es jedoch weitere Aspekte, durch die KI Schaden anrichten kann, auch wenn das ursprüngliche Anwendungsszenario ethisch verantwortungsvoll ist. Diese Art möglicher feindlicher Angriffe (adversarial attacks) auf die KI verändern Daten in böswilliger Absicht und täuschen die KI-Algorithmen, sodass diese falsche Ergebnisse liefern oder falsche Entscheidungen treffen.

Unter KI wird in diesem Beitrag eine spezielle Gruppe von Machine Learning Verfahren – das Deep Learning mit Deep Neural Networks – verstanden, die Klassifikationsaufgaben besonders erfolgreich lösen. Diese Art von Neuronalen Netzen bestehen aus mehreren Schichten von Neuronen. Sie lernen ausschließlich datengetrieben mittels Trainingsdaten, die die komplette Bandbreite an Eingaben bestmöglich repräsentieren. Mittels des Gradientenverfahrens wird im überwachten Lernen (Supervised Learning) mit Neuronalen Netzen ein Modell gelernt, dass neue Beispieldaten bestmöglich klassifiziert.

Feindliche Angriffe auf Anwendungen der KI erzeugen ein ernstes Bedrohungspotential für die Vertrauenswürdigkeit und Transparenz von KI-Algorithmen. Im Folgenden werden einige Anwendungsfälle, der technische Hintergrund von feindlichen Angriffen sowie einige mögliche Maßnahmen zur Reduzierung des Risikos skizziert.

5.2 Beispiele – Was sind feindliche Angriffe?

Sind KI-Systeme, mit denen wir tagtäglich in Kontakt kommen, vertrauenswürdig und robust gegen Störungen oder feindliche Angriffe?

Autonomes Fahren ist für eine Mehrheit der Bundesbürger, zumindest in bestimmten Situationen, wünschenswert (Bitkom 2018 [2]). Einerseits versprechen sie sich weniger Unfälle und mehr Sicherheit für alle Verkehrsteilnehmer, gleichzeitig sorgen sie sich aber um technische Fehlfunktionen, Hacker-Angriffe auf die Fahrzeuge sowie den Datenschutz. Die Autoindustrie führt auch in Deutschland bereits Tests mit autonomen Fahrzeugen in Innenstädten durch, wie beispielsweise Volkswagen in Hamburg ([3]), wobei die Fahrzeuge mit modernster Technik ausgerüstet sind, wie Laser-Scannern, Radarsensoren und Kameras. Diese Technik produziert kontinuierlich große Datenmengen. Entscheidend für automatisiertes Fahren ist die intelligente Verknüpfung und Bewertung aller Daten – durch entsprechende Software und KI. Auf Basis der Sensordaten identifizieren spezielle KI-Programme wie Neuronale Netze (Deep Learning) in Milli-Sekunden Verkehrsobjekte wie Verkehrsteilnehmer, Verkehrszeichen oder Fahrbahnmarkierungen.

Eine spezielle Form von Angriffsverfahren – sogenannte Adversarial Attacks (feindliche Angriffe) – bedrohen die Funktionsfähigkeit von KI-Algorithmen zur Bild- und Objekterkennung. Diese könnten autonomen Fahrzeugen große Probleme bereiten, wenn beispielsweise Verkehrszeichen falsch klassifiziert werden und folglich zu einer unerwünschten Fahrzeugsteuerung führen.

Ein anderes erfolgreiches Anwendungsfeld von KI ist die **Spracherkennung**, die beispielsweise in Speech-to-Text-Systemen wie Apple Siri, Google Assistant oder Amazon Alexa vorkommt. Abbildung 19 zeigt wie ein Angreifer die Entscheidungen einer KI verändern kann, in dem er über das Antwortverhalten eines Modells C ein eigenes Modell C‘ lernt.

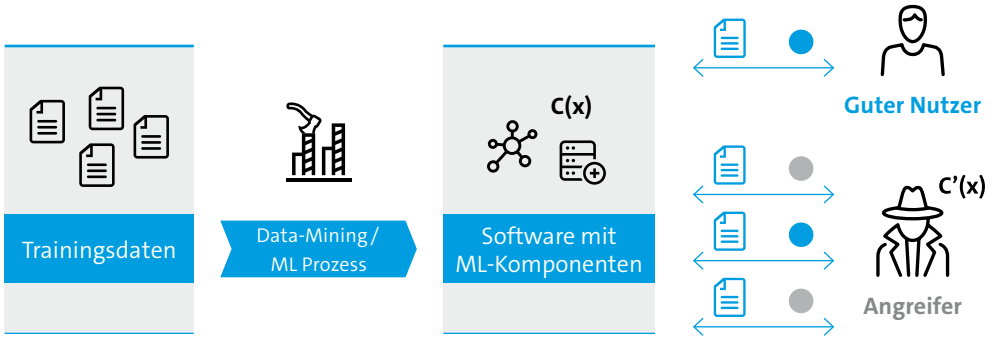


Abbildung 19: Prozess des feindlichen Angriffs auf KI: ein Angreifer, der sich mit dem Modell C(x) befasst, kann durch aktives Lernen zu seinem eigenen Verständnis des Modells als C’(x) gelangen. (Quelle: Sethi, Kantardzic, Ryu, 2017, »Security Theater«: On the Vulnerability of Classifiers to Exploratory Attacks)

Ein weiteres Anwendungsfeld von KI ist **Cybersicherheit**, d. h. die Abwehr von Angriffen auf die Netzwerkinfrastruktur und Computersysteme (z. B. Phishing Attacken, Botnet Attacken, Denial of Service (DoS) Attacken, Spam). Bei der Beobachtung des Netzverkehrs entstehen große Datenmengen, die Rückschlüsse auf typische Angriffsmuster erlauben und frühzeitig vor charakteristischen Anomalien warnen. Zu diesem Zweck werden KI-Verfahren eingesetzt, die ebenfalls anfällig gegenüber feindliche Attacken sein können und beispielsweise Spam oder schadhafte Inhalte nicht mehr zuverlässig ausfiltern können.

Auch wenn es bislang noch keine großen feindlichen Attacken auf KI-Systeme gab, sondern die meisten Attacken von Forschern erzeugt und publiziert worden sind, so ist die Gefahr sehr real. Zum einen wird die zunehmende Verbreitung von KI in verschiedensten Anwendungsfeldern die Durchführung solcher Attacken einfacher machen, zum anderen werden die erforschten Angriffspunkte durch die detaillierte Beschreibung auch leichter reproduzierbar. Die konkrete Motivation für feindliche Angriffe unterscheidet sich je Anwendungsfeld, bei Cyberangriffen steht z. B. die Installation eines Bots im Vordergrund, in Angriffen auf Bilderkennungssysteme wird z. B. unberechtigter Zugang beabsichtigt.

5.3 Hintergrund – Wie funktionieren feindliche Angriffe?

Worum konkret handelt es sich bei derartigen feindlichen Angriffen, die die Entscheidungen der Algorithmen bewusst täuschen wollen?

Bei feindlichen Angriffen auf Bilder und Fotos werden bestimmte Muster oder Pixel über das eigentliche Bild oder den Gegenstand gelegt, die das Neuronale Netz überlisten, d. h. zu einer falschen Klassifikation verleiten. Für die menschliche Wahrnehmung sind diese Muster nicht sichtbar, die Bilder sind nicht zu unterscheiden.

Abbildung 20 zeigt ein Foto einer Verkehrssituation mit Fußgängern (links oben) und der Segmentierung bzw. Klassifikation rechts mit Personen. In der Mitte links ist die überlagerte Störung (universal noise) und rechts die entsprechende Klassifikation ohne Personen.
Unten links ist die Kombination Foto und Störung, und rechts die Klassifikation dargestellt.

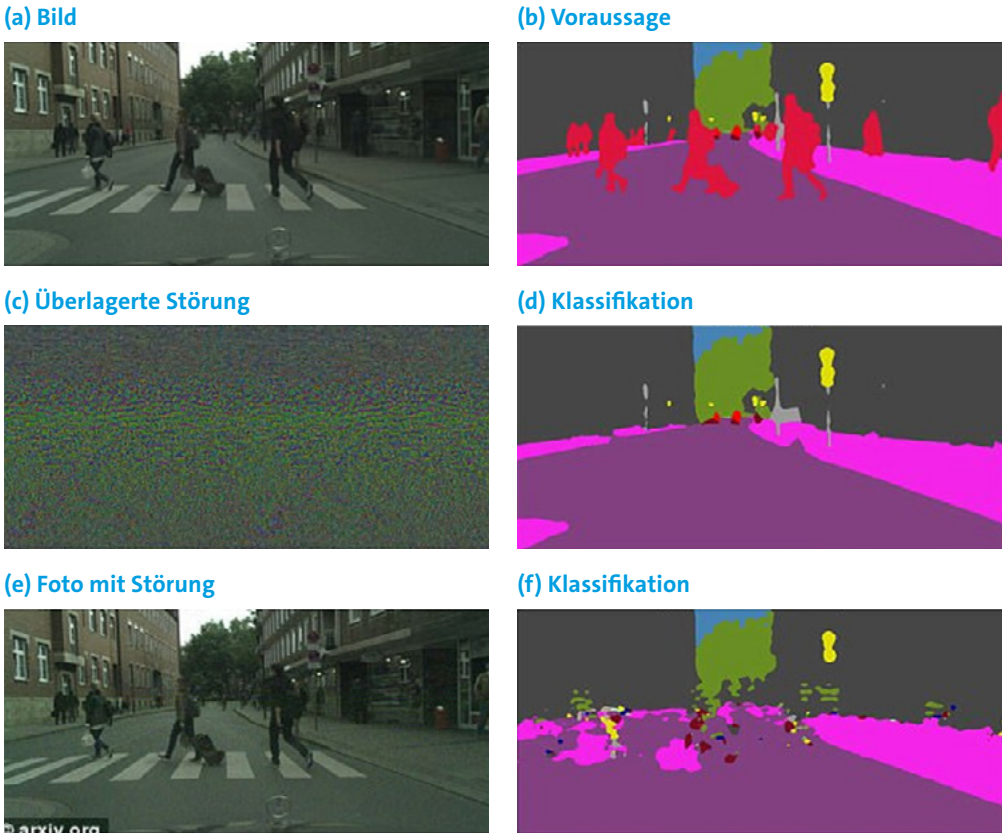


Abbildung 20: Durch universale Störungen aus dem Bild entfernte Personen. (Quelle: Jan Hendrik Metzen et al. [4])

In diesem Fall wird von einer **digitalen Attacke** gesprochen, da der Angreifer direkten Zugriff auf die tatsächlich in das Modell eingegebenen Daten besitzt. In einer realen Umgebung kann dies vorkommen, wenn ein Angreifer eine Bilddatei auf einen Webdienst hochlädt und die Datei absichtlich so manipuliert, dass sie falsch gelesen wird. Beispielsweise können Spam-Inhalte als Bilddatei in sozialen Medien gepostet werden, wobei die Bilddatei manipuliert wurde, um dem Spam-Filter zu entgehen.

In einer anderen Forschungsstudie wurden Verkehrsschilder systematisch verändert, die in ähnlicher Form im Alltag als Graffitis zu finden sind. Abbildung 21 zeigt links ein Stoppschild mit realem Graffiti. Das rechte Foto zeigt ein absichtlich manipuliertes Stoppschild, das den menschlichen Betrachter nicht in der Klassifikation irritiert, bei getesteten Deep Learning Systemen zur Bilderkennung jedoch als (falsche) Klassifikation eine Geschwindigkeitsbegrenzung (»Speed Limit 45 miles«) ergab [5].

Abbildung 21: Besprühtes (links) und manipuliertes (rechts) Verkehrsschild, das falsch klassifiziert wurde (Quelle: Eykholt et.al. 2018)

In diesem Fall wird von einer **physischen Attacke** gesprochen, hier haben die Gegner keinen direkten Zugriff auf die digitale Darstellung des Modells. Stattdessen wird das Modell mit Eingaben gespeist, die von Sensoren wie einer Kamera oder einem Mikrofon kommen. Der Gegner kann Objekte in der physischen Umgebung platzieren (manipuliertes Verkehrsschild), die von der Kamera gesehen wird.

Carlini und Wagner (2018 [6]) erzeugten feindliche Angriffe für die Erkennung von gesprochener Sprache als Audiosignal, indem sie spezielle Störmuster mit Audioströmen kombinierten. Abbildung 22 zeigt oben links die Wellenform einer Sprachaufzeichnung, in der Mitte die Verarbeitung durch ein Neuronales Netz (in diesem Fall Mozillas DeepSpeech Implementierung), und oben rechts das Analyseergebnis in Textform. Durch Zugabe geringfügiger Verzerrungen (Wellen) in der Mitte, wird unten ein Signal erzeugt, dass das Neuronale Netz zu einer komplett anderen Textausgabe (unten rechts) veranlasst.

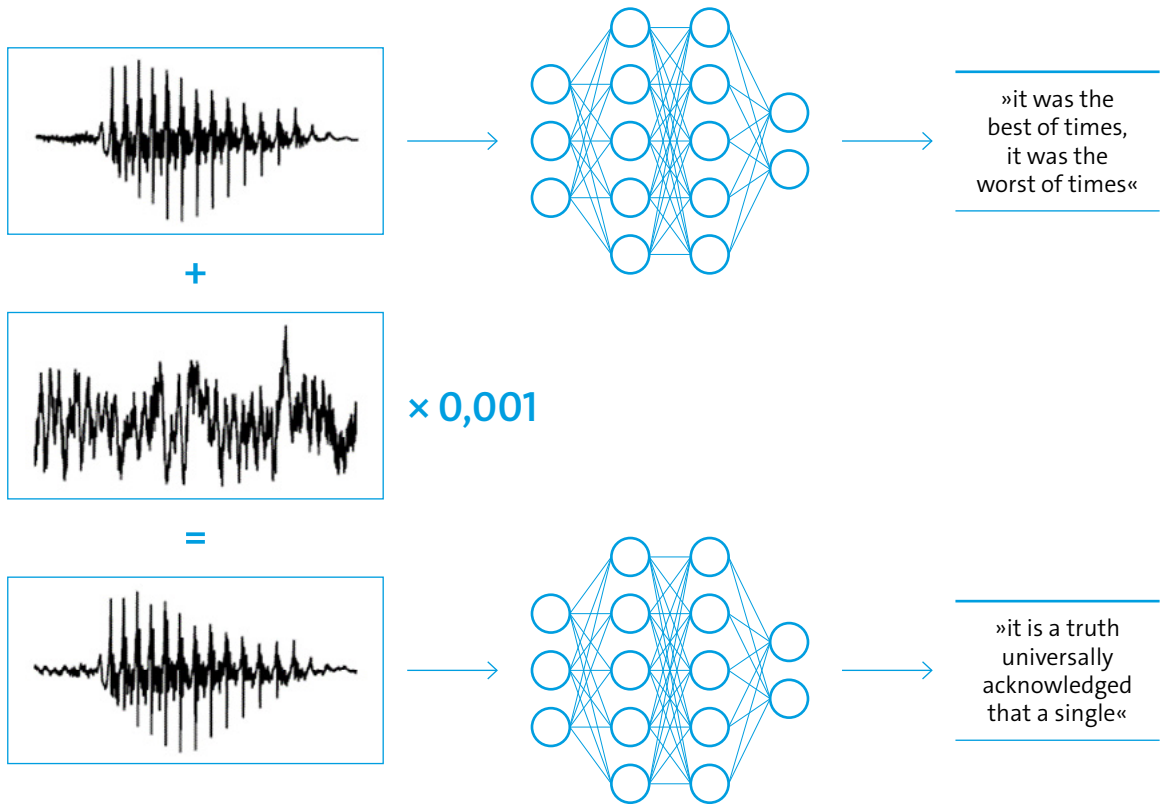


Abbildung 22: Feindliche Angriffe bei Audio Signalen in der automatischen Spracherkennung (Quelle: Carlini und Wagner 2018)

In diesem Fall hatten die Forscher vollständige Kenntnis der Architektur und der Modellparameter des Neuronalen Netzes, was als **White-Box-Angriff** bezeichnet wird. **Black-Box-Angriffe** sind Attacken, bei denen der Angreifer keine Kenntnis der Modellparameter und der Architektur des Neuronalen Netzes hat. Diese Unkenntnis erschwert die Erfolgswahrscheinlichkeit von Angriffen erheblich, macht sie aber nicht unmöglich wie verschiedene Untersuchungen zeigen[7]. Es wurde beobachtet, dass sich feindliche Beispiele zwischen verschiedenen Modellen übertragen lassen ([8] **»Übertragbarkeit«**) und verwendet werden können, um diese im Black-Box-Szenario zu erstellen.

In diesem Zusammenhang können weitere Risiken der Informationssicherheit (BSI 2019 [15]) entstehen, die durch die Nutzung von fremdem Wissen (z. B. vortrainierte Modelle) innerhalb offener Lieferketten (Software-Komponenten) durch den Austausch über Entwicklerplattformen (wie Github) erfolgen.

Über die Ursachen der Anfälligkeit von Neuronalen Netzen bei feindlichen Attacken herrscht in der Forschung keine Einigkeit, Qiu et al. (2019 [16]) haben einen Überblick zu verschiedenen Erklärungsansätzen zusammengestellt.

Zusammenfassend existieren also bereits unterschiedlichste Angriffsformen für verschiedene Anwendungsfelder des Machine Learning, die erhebliche Risiken durch falsche Entscheidungen bergen.

5.4 Lösungen – Wie kann das Risiko durch feindliche Angriffe reduziert werden?

Welche Möglichkeiten haben wir, Angriffe zu erschweren und damit die Sicherheit von KI-Systemen sicherzustellen?

Zur Abwehr von feindlichen Angriffen auf KI werden verschiedene Verfahren eingesetzt, die die Anfälligkeit gegen diese messen, die Robustheit der Verfahren verbessern und damit das Risiko von Fehlklassifikationen reduzieren sowie die Transferierbarkeit von Black-Box Attacken einschränken.

Ein wirkungsvolles Verfahren ist beispielsweise **Adversarial Training** ([9]), bei dem die Trainingsdaten um feindliche Beispiele erweitert werden. Diese feindlichen Beispiele werden von den Entwicklern des KI-Modells selbst generiert und korrekt klassifiziert, sodass die Anwendung des trainierten Modells eine korrekte Klassifikation von anderen feindlichen Beispielen ermöglicht. Diese Anreicherung mit feindlichen Beispielen kann auf unterschiedliche Art erfolgen. Eine Methode nutzt Generative Adversarial Networks ([10]), bei denen zwei Neuronale Netze gegeneinander laufen und neue feindliche Beispiele erzeugen.

Ein anderes Verfahren ist **SafetyNet** ([11]). SafetyNet besteht aus dem ursprünglichen Modell (Classifier) und einem Detektor, der den internen Zustand der späteren Aktivierungsschichten im ursprünglichen Modell untersucht. Wenn der Detektor feststellt, dass ein Beispiel feindlich ist, wird der feindliche Datensatz (Probe) zurückgewiesen. Auf dieser Basis haben Lu et al. (2017) die Anwendung SceneProof für Fotos erzeugt, die feststellt, ob ein Foto real oder eine Fälschung, also ein feindliches Beispiel ist.

Ein drittes Beispiel ist das **Adversarial Logit Pairing** ([12]), das eine Erweiterung des Adversarial Training darstellt. Beim Adversarial Logit-Pairing wird ein Modell trainiert, das die Ähnlichkeit zwischen den Logit-Aktivierungen (d. h. die letzte Ebene des Neuronalen Netzes mit den Rohwerten für die Klassifikation) des Modells für ungestörte und feindliche Beispiele desselben Bildes zum Ziel hat, wodurch empirisch trennscharfe Klassifikationen erzielt werden.

Für die Deutsche Telekom ist die Vertrauenswürdigkeit von KI-Anwendungen von großer Bedeutung. Eine Basis sind die Leitlinien für die Entwicklung und den Einsatz von KI ([13]). In Zukunft werden technische Verfahren zur Prüfung von KI auf Robustheit, Sicherheit, Einhaltung der Privatsphäre und Diskriminierungsfreiheit ein zentraler Baustein für die Akzeptanz von KI durch die Kunden und Bürger sein. Verschiedene Unternehmen entwickeln zurzeit Ansätze hierzu.

Die Telekom Innovation Laboratories entwickelt in einer Kooperation mit Sicherheitsforschern der Ben Gurion Universität in Israel ([14]) eine Test- und Evaluationsumgebung für KI-Modelle. Hier stehen Fragen der Robustheit von KI-Modellen gegenüber feindlichen Angriffen im Fokus, d. h. eine mögliche Überprüfung der Modelle hinsichtlich ihrer Anfälligkeit für feindliche Angriffe sowie eine Messung des Grades der Betroffenheit im Zusammenspiel mit zu identifizierenden anwendbaren Gegenmaßnahmen. Weitere Schwerpunkte der Evaluationsumgebung sind die Untersuchung des ausreichenden Schutzes der Privatsphäre (Privacy) und die Vermeidung von Verzerrungen (Bias) durch die Anwendung des KI-Modells.

Perspektivisch soll ein Werkzeugkasten entstehen, der eine Bewertung von KI-Modellen hinsichtlich Robustheit, Konformität bzgl. Schutz der Privatsphäre und Freiheit von diskriminierenden Verzerrungen ermöglicht. Entsprechende Verfahren zur Reduzierung und Abwehr von möglichen Gefahren können hierüber direkt evaluiert werden. Ein solcher Werkzeugkasten kann dann in verschiedene Produktvisionen einfließen. Denkbar wäre zum Beispiel eine Art »Virenscanner« für KI-Systeme, der im Live-Betrieb mittels bestimmter Verfahren ermittelt, ob das System von feindlichen Angreifern kompromittiert wurde. Eine weitere Produktausprägung könnte ein Self-Service-Tool für Machine Learning Entwickler sein, die vor Inbetriebnahme ihres KI-Systems sicherstellen wollen, dass die verwendeten Modelle vorher definierten Mindeststandards hinsichtlich Robustheit genügen. Als Ergebnis könnte der Entwickler dann einen verbindlichen Report über die Modellgüte erhalten, der dann auch im Sinne einer rechtlichen Nachweispflicht von offizieller Stelle bestätigt würde. Der Werkzeugkasten würde die bereits existierenden Richtlinien für KI-Systeme um konkrete Funktionalitäten sinnvoll ergänzen. Robustheit, Transparenz und Nachvollziehbarkeit sind zentrale Anforderungen an vertrauenswürdige KI und umfassen künftig ein großes Aufgabenspektrum für Unternehmen und Forschung.

5.5 Zusammenfassung

KI wird erfolgreich in verschiedenen Anwendungsbereichen eingesetzt. Wichtige Voraussetzungen für die Akzeptanz von KI sind Diskriminierungsfreiheit, Einhaltung der Privatsphäre sowie Robustheit und Sicherheit. Dies betrifft insbesondere die Korrektheit der Ergebnisse auch unter Berücksichtigung von Störfaktoren wie Angriffen von außen. Ein Angriffsformat auf die Funktionsfähigkeit von KI wird Adversarial Attacks (feindliche Angriffe) genannt. Diese feindlichen Angriffe können die KI zu falschen Ergebnissen und Entscheidungen bringen. Beispiele aus der Bilderkennung beim autonomen Fahren, der Spracherkennung und der Cybersicherheit illustrieren die Wirkung solcher Angriffe. Es existieren bereits viele verschiedene Arten von solchen Angriffen, wie digitale vs. physische Angriffe, White-Box- vs. Black-Box-Attacken oder das Thema der Transferierbarkeit von feindlichen Angriffen. Verschiedene Verfahren zur Identifikation und Abwehr von feindlichen Angriffen bzw. zur Stärkung der Robustheit der KI-Modelle werden aktuell durch Unternehmen wie die Deutsche Telekom entwickelt. Diese Lösungen werden künftig Grundlage für den Einsatz und die Akzeptanz von KI sein, denn bei vielen der KI-Anwendungsfelder wären die negativen Auswirkungen feindlicher Angriffe unvertretbar. Nur mit der Sicherheit, dass KI korrekte, unmanipulierte Ergebnisse liefert, werden wir langfristig entsprechendes Vertrauen bei den Nutzern und in der öffentlichen Diskussion erreichen.

5.6 Literaturverzeichnis

[1] D21 (2019), Denkimpuls Digitale Ethik: Bias in algorithmischen Systemen – Erläuterungen, Beispiele und Thesen ↗ **https://initiatived21.de/app/uploads/2019/03/algomon_denkimpuls_bias_190318.pdf**, abgerufen 10.7.2019

[2] Bitkom (2018) Autonome Autos: Hoffnung auf mehr Sicherheit und Umweltschutz, ↗ **https://www.bitkom.org/Presse/Presseinformation/Autonome-Autos-Hoffnung-auf-mehr-Sicherheit-und-Umweltschutz.html**, abgerufen 10.7.2019

[3] Spiegel Online (2019), VW fährt in Hamburg jetzt autonom ↗ **https://www.spiegel.de/auto/aktuell/hamburg-volkswagen-testest-autonomes-fahren-in-deutscher-grossstadt-a-1251825.html**, abgerufen 10.7.2019

[4] Metzen et al. (2017), Universal Adversarial Perturbations Against Semantic Image Segmentation, IEEE International Conference on Computer Vision

[5] Eykholt et al.(2018), Robust Physical-World Attacks on Deep Learning Visual Classification, ↗ **https://arxiv.org/pdf/1707.08945.pdf**, abgerufen 10.7.2019

[6] Carlini/Wagner (2018), Audio Adversarial Examples: Targeted Attacks on Speech-to-Text, ↗ **https://arxiv.org/pdf/1801.01944.pdf**, abgerufen 10.7.2019

[7] Zhao et al.(2018), Generating natural adversarial examples, ↗ **https://arxiv.org/pdf/1710.11342.pdf**, abgerufen 10.7.2019

[8] Szegedy et al.(2014), Intriguing properties of neural networks. International Conference on Learning Representations, ↗ **https://arxiv.org/pdf/1312.6199.pdf**, abgerufen 10.7.2019

[9] Kurakin et al.(2017), Adversarial Machine Learning at Scale, ↗ **https://arxiv.org/pdf/1611.01236.pdf**, abgerufen 10.7.2019

[10] Xiao et al.(2018), Generating Adversarial Examples with Adversarial Networks, ↗ **https://www.ijcai.org/proceedings/2018/0543.pdf**, abgerufen 10.7.2019

[11] Lu et al. (2017), SavetyNet ↗ **https://arxiv.org/pdf/1704.00103.pdf**, abgerufen 10.7.2019

[12] Kannan et al. (2018), Adversarial Logit Pairing ↗ **https://arxiv.org/pdf/1803.06373.pdf**, abgerufen 10.7.2019

[13] Deutsche Telekom (2018), ↗ **https://www.telekom.com/de/konzern/digitale-verantwortung/details/ki-leitlinien-der-telekom-523904**, abgerufen 10.7.2019

[14] Ben Gurion University, ↗ **https://cyber.bgu.ac.il/**, abgerufen 10.7.2019

[15] Bundesamt für Sicherheit in der Informationstecknik (2019), Deutsch-französisches IT-Sicherheitslagebild, ↗ **https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/DE-FR-Lagebild/de-fr_Lagebild.pdf?__blob=publicationFile&v=7**, abgerufen 10.7.2019

[16] Qiu et al. (2019), Review of Artificial Intelligence Adversarial Attack and Defense Technologies, ↗ **https://doi.org/10.3390/app9050909**, abgerufen 10.7.2019

6 Implementierung algorithmischer Fairness und Nachvollziehbarkeit für branchenübergreifende KI-Anwendungen

6 Implementierung algorithmischer Fairness und Nachvollziehbarkeit für branchen-übergreifende KI-Anwendungen

Maike Havemann, Robin Rojowiec

6.1 Einleitung

Ein System der Künstlichen Intelligenz (KI) ist eine Anwendung, die sich basierend auf historischen Daten und Erfahrungen weiterentwickeln und lernen kann. Dies führt zu einer neuen Reihe von operativen und kulturellen Paradigmen. Vor allem durch die Veränderung der Systeme und durch neue Daten können potenzielle Risiken und Probleme entstehen (z. B. gelernter Rassismus oder unfaire Jobvergabe nach Geschlecht). Diese Probleme entstehen meist durch den menschlichen Bias in den Daten, der während der Optimierung des jeweiligen Modells noch verstärkt wird. Deshalb müssen die Vorhersagen der Modelle verstanden, kontrolliert und gesteuert werden können. Auch benötigen die Algorithmen Daten aus Produktionssystemen, um weiter lernen zu können, was wiederum neue Überlegungen zu Datensicherheit, Betriebsrichtlinien und DevOps mit sich bringt. Besonders herausfordernd ist an dieser Stelle, dass die Entwicklung und Nutzung von KI in unterschiedlichen Bereichen und von unterschiedlichen Teams erfolgt. Data Scientists besitzen tiefgreifendes Wissen in Statistik und Mathematik, ohne welches eine Entwicklung und Implementierung von KI-Systemen schlichtweg nicht möglich wäre. Für die Anwender ist jedoch die Sichtweise auf die Systeme nicht nur von der mathematischen Korrektheit, sondern auch der Nachvollziehbarkeit der Ergebnisse und Transparenz der Entscheidungen geprägt. Schließlich hat bei weitem nicht jeder Anwender ein tiefes Verständnis von den aktuellen Algorithmen und muss sich auch gegenüber Dritten für Entscheidungen rechtfertigen können. Diese Lücke für ein sich ständig weiterentwickelndes System zu schließen, ist aufwendig und erfordert neue Fähigkeiten und Werkzeuge. [1]

6.2 Automatisierte Bias-Reduzierung

Schon immer steht die menschliche Subjektivität bei Entscheidungen in Bezug auf Fairness beziehungsweise Diskriminierung in der Diskussion. Da historische menschliche Entscheidungen als Trainingsmenge die Grundlage für maschinelle Lernmodelle, zum Beispiel in den Themengebieten Recruiting, Credit-Scoring und Strafverfolgung bilden, sind darin möglicherweise auch Vorurteile enthalten. So entsteht eine Verzerrung (Bias) der Daten und somit der Folgeentscheidung. Gerade in den letzten Jahren, in denen der Mega-Trend KI immer mehr skaliert, ist es zu einem Anstieg der Beiträge über algorithmische Fairness in der Literatur des maschinellen Lernens und des Data Mining gekommen, wobei die Grundprinzipien durch Erkennung,

Schätztheorie und Informationstheorie definiert wurden. Es gibt zwei Hauptbegriffe von Fairness bei der Entscheidungsfindung: Gruppenfairness und individuelle Fairness. Gruppenfairness im weitesten Sinne unterteilt eine Bevölkerung in Gruppen, die durch geschützte Attribute definiert sind (z. B. Geschlecht, Nationalität oder Religion) und strebt danach, dass einige statistische Maße gruppenübergreifend gleich sind. Individuelle Fairness hingegen sucht nach ähnlichen Personen, die ähnlich behandelt werden. Die Überprüfung auf Gruppenfairness ist eine relativ einfache Berechnung statistischer Kennzahlen. Die Überprüfung auf individuelle Fairness ist jedoch komplexer, da es viele geschützte Attribute mit vielen Werten geben kann und die Bewertung von Daten mithilfe von Modellen kostenintensiv ist. Die verbreitetste Art von Bias-Reduzierungs-Algorithmen sind solche, die sich mit der Nachverarbeitung von Daten beschäftigen, da vor oder bei der Datenerhebung zumeist keine Einflussnahme möglich ist. Die Methodik der Nachbearbeitungsalgorithmen besteht darin, dass eine Teilmenge von Proben genommen und ihre vorhergesagten Klassen geändert werden, um eine Gruppenfairness-anforderung zu erfüllen. Die vorhergesagten Klassen sind hierbei Entscheidungen, z. B. binärer Art, über eine Jobeignung. In diesem Fall gäbe es zwei Klassen: Positiv (Bewerber ist geeignet) oder Negativ (Bewerber ist nicht geeignet).

Neuere Ansätze schlagen anstatt der zufälligen Auswahl und Anpassung von Vorhersagen das Trainieren eines Bias-Detektors vor, welcher Proben findet, deren Modellvorhersage sich ändert, wenn die geschützten Attribute geändert werden. Alle anderen Eigenschaften bleiben dabei konstant. So werden Beispiele ausgewählt, die individuelle Fairness-Probleme haben oder wahrscheinlich haben werden. Auf diese Weise können sowohl Gruppen- als auch individuelle Fairness gemeinsam angesprochen werden, da diese in direktem Zusammenhang stehen. [2]

Für das Trainieren der Erkennung von individuellem Bias gibt es die Möglichkeit, systematisch den Entscheidungsraum eines Blackbox-Klassifikators zu untersuchen, um Testproben zu erzeugen, die eine erhöhte Wahrscheinlichkeit haben, verzerrt zu werden. Die Methode verwendet zwei Arten der Suche: (a) eine globale Suche, die den Entscheidungsraum so ausleuchtet, dass verschiedene Bereiche abgedeckt werden, und (b) eine lokale Suche, die Testfälle erzeugt, indem sie die Werte nicht geschützter Merkmale einer bereits gefundenen, individuell verzerrten Stichprobe intelligent stört. Das Prinzip zur Reduzierung des Gruppen-Bias arbeitet mithilfe von Veränderungen der Label-Ausgaben des Klassifikators. Folglich sind Nachbearbeitungsalgorithmen erforderlich, insbesondere solche, die einen Klassifikator als komplette Blackbox behandeln können. Im folgenden Diagramm sind verschiedene Algorithmen zur Erkennung von individuellem Bias aufgeführt und mit dem Bias aus dem ursprünglichen Datensatz verglichen [3]:

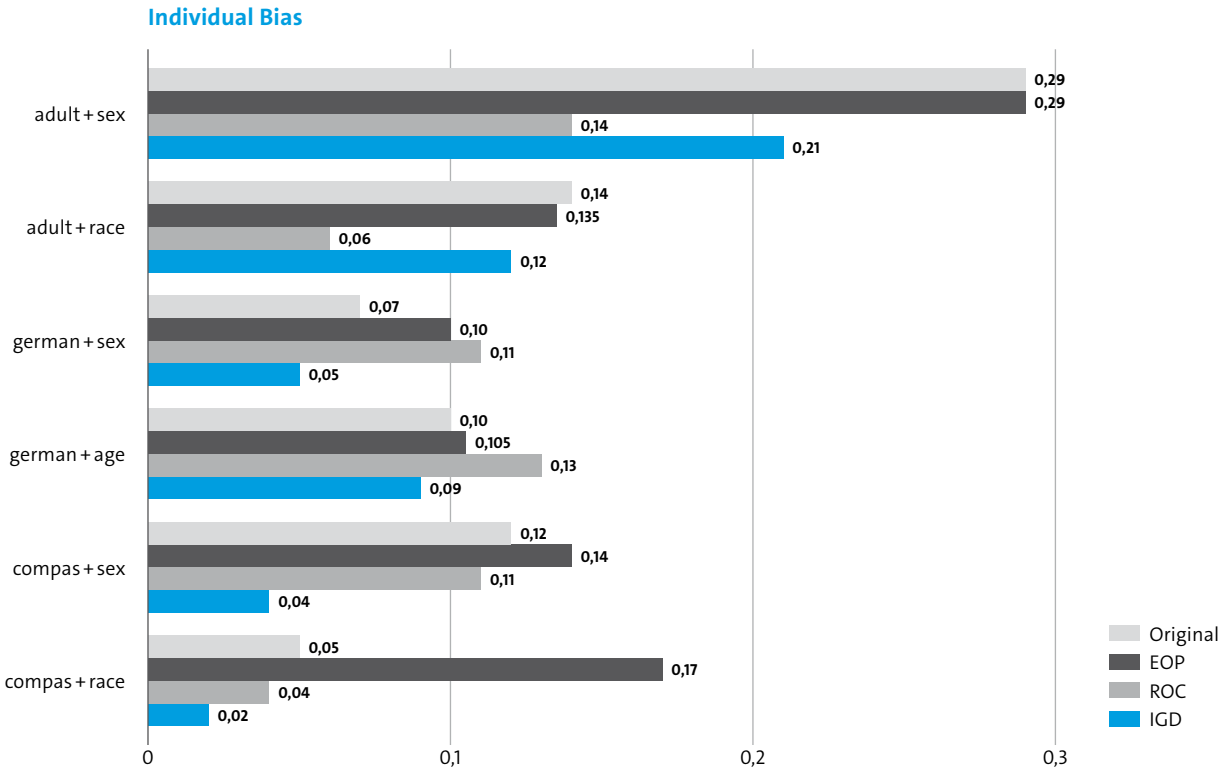


Abbildung 23: Individueller Bias Korrektur mit verschiedenen Attributen. Die Datensätze UCIAdult (adult), UCI Statlog German Credit (german) und ProPublica COMPAS (compas) wurden dazu herangezogen und die jeweiligen Attribute (sex, race und age) überprüft

Die Algorithm-ROC (reject option classification) und EOP (equalized odds post-processing) sind gängige Formen der Bias-Verminderung. ROC sorgt dafür, dass bei Klassen mit einer geringen Konfidenz des Algorithmus die jeweils andere Klasse (binäres Klassifikationsproblem) ausgewählt wird. Dafür muss ein Schwellwert festgelegt werden. EOP hingegen berechnet die Wahrscheinlichkeiten, mit denen ein Beispiel in die jeweilige Klasse kommen müsste und korrigiert nach der Vorhersage solche, bei denen eine andere Zuweisung wahrscheinlicher ist. Der individuelle Bias wird durch die Kennzahl *discrete impact,* also der Auswirkung auf die Vorhersage, gemessen (kleiner ist besser) [4].

Weiterhin ist auf dem Diagramm zu erkennen, dass die Merkmale *Rasse, Geschlecht* und *Alter* unterschiedliche Einflüsse auf den Bias haben. Der mit roten Balken dargestellte Bias-Detektor für *Individual and Group Debiasing (IGD)* stellt hierbei den aktuellen Stand der Technik dar. Seine Besonderheit liegt darin, dass die Bestimmung des Individuellen Bias dafür genutzt wird, den Gruppenbias anzupassen. Dadurch kann der Einfluss verschiedener Attribute auf die Klassenzuordnung genauer gesteuert werden. Dies wird auch in Abbildung 23 erkennbar, in welcher dieser Algorithmus für jede Kombination von Attributausprägungen einen deutlich geringeren Bias zulässt als vorherige Verfahren. Ähnliche Ergebnisse können auch für den Gruppenbias

erzielt werden. Zusammengefasst werden vier Schritte im Algorithmus durchlaufen, bis alle Ausgaben eines Klassifizierers auf Bias korrigiert wurden:

1. Trainieren eines beliebigen Klassifikators
2. Erstellung eines Validationsdatensets und Bestimmung des individuellen Bias aller seiner Beispiele
3. Training eines individuellen Bias-Detektors (ebenfalls ein Klassifizierer, der anhand der gegebenen Attribute und der Vorhersage bestimmt, ob ein Bias vorliegt)
4. Anwendung des Bias-Detektors auf ungesehene Beispiele zur Laufzeit

Die Anwendung erfolgt durch die Vorhersage des Bias-Detektors bei Beispielen der weniger privilegierten Gruppe und durch die entsprechende Korrektur zu einer anderen Klasse, falls der Detektor einen Bias erkennt. Alle anderen Beispiele bleiben unverändert, auch die der Start-privilegierten Gruppe. Bezogen auf die *Accuracy* per Klasse (ausbalanciert) zeigt der IGD-Ansatz analog zu Abbildung 23 konstant bessere Werte als alle anderen Verfahren und weniger Unterschiede zwischen den einzelnen Attributen (siehe Abbildung 24) [5].

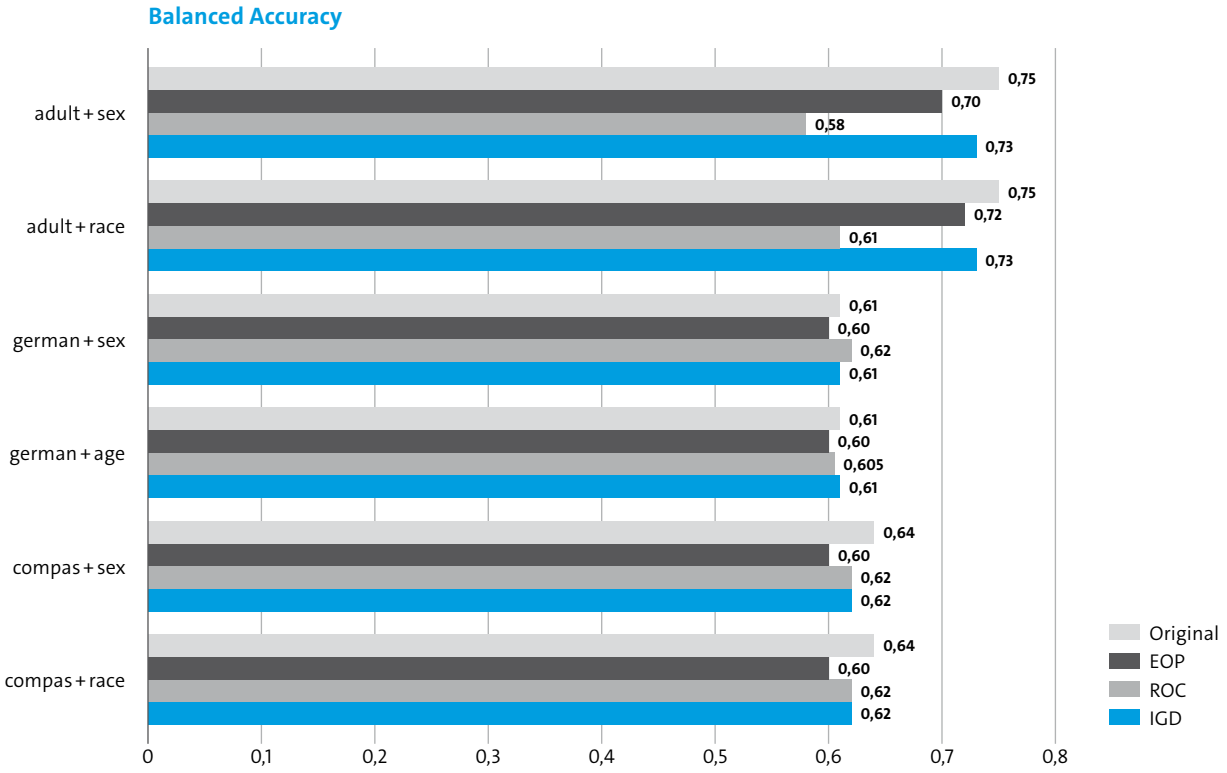


Abbildung 24: Ausgewogene Genauigkeit der verschiedenen Verfahren zur Bias-Reduzierung

6.3 Nachvollziehbarkeit der Entscheidungen eines Modells mit MACEM

Ein Großteil der aktuellen Algorithmen und Modelle, gerade aus dem Bereich *Deep Learning,* besteht aus komplexen verschachtelten Funktionen, die keine einfache Interpretation der Entscheidungsfindung zulassen. Die Werte zur Berechnung verschiedener Ausgaben werden während des Trainingsprozesses vielfach angepasst und die Zusammenhänge zwischen den einzelnen Gewichten und Merkmalen sind nicht ohne weiteres rekonstruierbar. Daher wurden in den letzten Jahren viele Versuche unternommen, die Ausgaben durch Analyse und Veränderung der Eingaben von außen besser zu verstehen und transparenter darzustellen.

Aktuell erzielt das *MACEM (Modell Agnostic Contrastive Explanations for Machine Learning Classification Modells)* in dieser Aufgabe die besten Ergebnisse. Der Algorithmus selbst ist für strukturierte Daten, genauer für kategorische und wertbasierte Angaben, ausgelegt. Da sich üblicherweise in der Verarbeitung von unstrukturierten Daten nach einer Menge von Vorverarbeitungsschritten strukturierte Inhalte für ein Training ergeben, sollte er auch auf solche Daten anwendbar sein. Ein weiterer Mehrwert dieses Algorithmus ist die Tatsache, dass er nur die Eingabe in das zu erklärende Modell und die Vorhersage benötigt, um eine erklärende Struktur zu generieren. Dies ist im Gegensatz zu anderen Verfahren vorteilhaft, weil weder das Modell selbst auseinander genommen noch Teile des Modells analysiert werden müssen. Dadurch kann jede Art von Modell durch den Algorithmus erklärt werden [6].

Zunächst wird X als Datenbereich definiert mit den Datensätzen (x_0, t_0), wobei x_0 der unveränderte *Feature Vektor* der Eingabe und t_0 die Klasse mit dem höchsten Wert darstellt (vorhergesagte Klasse). Eine Veränderung der Features wird beschrieben durch den Vektor δ mit der Veränderungsgleichung $x = x_0 + \delta$. Der Algorithmus durchläuft drei Schritte, um die Werte δ_{pos}, δ_{neg} zu bestimmen, welche einen beliebigen *Feature Vektor* so verändern, dass er eine Klassifizierung in die positive oder negative Klasse bei einem binären Klassifizierungsproblem verursacht. Nachfolgend sind diese Schritte beschrieben und entsprechende Gleichungen zur Problemdarstellung mit ihren Eingaben [7] aufgezeigt [8]:

1. Lösung des Optimierungsproblems:
   $\delta_{pos} \leftarrow \text{argmin}_{\delta \in \Delta_{PP}}\ c \cdot f_k^{pos}(x_0, \delta) + \beta \|x_0 + \delta - b\|_1 + \|x_0 + \delta - b\|_2^2 + \gamma \|x_0 + \delta - AE(x_0 + \delta)\|_2^2$

2. Lösung des Optimierungsproblems:
   $\delta_{neg} \leftarrow \text{argmin}_{\delta \in \Delta_{PN}}\ c \cdot f_k^{neg}(x_0, \delta) + \beta \|\delta\|_1 + \|\delta\|_2^2 + \gamma \|x_0 + \delta - AE(x_0 + \delta)\|_2^2$

3. Rückgabe von δ_{pos}, δ_{neg}

Der *Auto Encoder AE* dient zur zielgerichteten Generierung von den wahrscheinlichsten *Feature Vektoren.* Im Wesentlichen wird eine Funktion optimiert, die die Bedeutung einzelner *Features* für die finale Klassifikation abschätzt. Das bedeutet, der Einfluss, den ein leeres *Feature* auf die Vorhersage und deren Berechnung hat, kann anhand eines Basiswertes bestimmt werden.

Dieser wird ebenfalls zunächst abgeschätzt als der Median über die Menge an Featureausprägungen. Für kategorische *Features* wird der am häufigsten vorkommende Wert als Basiswert gesetzt, da diese Ausprägung für den Klassifikator die am wenigsten Interessante sein sollte [9]. Der Quellcode sowie die detaillierte Beschreibung der Optimierung und Berechnung sind frei auf Github verfügbar [10].

6.4 Use Cases

Versicherungen

Das Versicherungsgeschäft wird stetig wettbewerbsintensiver und ist von zahlreichen Herausforderungen geprägt. Fachpersonal kann die Bewertung von Risiken und Zusammenhängen aufgrund mangelnder Einheitlichkeit und Übersichtlichkeit in den unstrukturierten Datenmengen nur bedingt vornehmen. Hier können Modelle des maschinellen Lernens, die auf historischen Kunden- und Schadenfalldaten basieren, unterstützen, konsistentere und genauere Risikobewertungen vorzunehmen. Diese Modelle können Preisvorschläge für einzelne Kunden auf der Grundlage unterschiedlicher Merkmale in ihrem Profil liefern. Um den regulatorischen Compliance-Standard einzuhalten, müssen die Versicherer allerdings in der Lage sein, zu erklären, wie diese Modelle funktionieren. Mithilfe von MACEM können die genauen Überlegungen nachvollzogen werden, die vom Modell bei der Abgabe priorisiert werden. Darüber hinaus kann durch die Implementierung von Bias-Reduzierungs-Algorithmen sichergestellt werden, dass ebendieses Modell faire Empfehlungen abgibt.

Telekommunikation

Die effektive Aufrechterhaltung physischer Anlagen und Infrastrukturen ist in der heutigen Zeit für Telekommunikationsunternehmen unerlässlich. Anlagenausfälle können zu Serviceausfällen führen, einer der Hauptursachen für Kundenabwanderungen sowie negative Kundenbewertungen. Die Priorisierung der Wartungen ist ein kostenintensiver und komplexer Prozess. Es ist oft schwierig, Anlagenausfälle im Feld zu erkennen, bevor sie zu einem Problem im Netzwerk führen. Modelle für das maschinelle Lernen, die auf der Grundlage historischer Anlagenausfalldaten trainiert wurden – einschließlich Sensordaten, Bildern, die von Drohnen aufgenommen wurden, und alten Wartungsberichten – können den Telekommunikationsunternehmen helfen, Anlagenausfälle vorherzusagen, bevor diese tatsächlich geschehen. Netzwerkingenieure und IT-Betriebe müssen sicherstellen, dass ihre KI-Modelle bei komplexen Daten den exakten Ausfall vorhersagen. Durch die Rückverfolgung mit MACEM wird den Teams ermöglicht, den Zustand ihrer Modelle über die gesamte Laufzeit nachzuvollziehen sowie bestimmte Geschehnisse mit Vorhersagen in Verbindung zu bringen und eine gewisse Zuverlässigkeit und Genauigkeit des Modells zu garantieren. Die Reduzierung von Bias ist in diesem Fall hilfreich, wenn in der Vergangenheit bestimmte Wartungen unfair durch Menschen priorisiert worden sind. Somit kann eine gleichmäßige und faire Wartung garantiert werden.

Lieferketten

Eine effektive Nachfrageprognose ist unerlässlich, um die Betriebskosten niedrig zu halten und gleichzeitig die Verbrauchernachfrage zu decken. Oftmals sind Unternehmen jedoch nicht in der Lage, mit dem Umfang und der Vielfalt der Daten umzugehen, die benötigt werden, um Echtzeit-Änderungen der Nachfrage zu berücksichtigen. Prognosen, die sich nicht an ständig wechselnde Variablen im heutigen Markt anpassen können, können zu Fehleinschätzungen im Wert von mehreren Millionen Euro führen und das Ergebnis eines Unternehmens erheblich beeinträchtigen. Machine-Learning-Modelle können trainiert werden, um einem Unternehmen zu helfen, seine Forecast-Value-Added-Kennzahlen zu verbessern, indem sie aus historischen, erfolgreichen und erfolglosen Prognosedaten lernen. Diese Modelle helfen, die Bedarfsprognose besser anzupassen. Dem Supply-Chain-Team wird durch MACEM ermöglicht, die Genauigkeit des Modells jederzeit zu überwachen, sodass geprüft werden kann, ob die KI-basierten Anwendungen konsistente Ergebnisse liefern.

Finanzen

Banken geben bis zu einem Fünftel ihrer Betriebskosten für die Bekämpfung der Geldwäsche aus. Geschieht dies nicht effektiv, können massive Geldbußen von den staatlichen Aufsichtsbehörden verhängt werden. Die meisten Institute setzen regelbasierte Systeme ein, um potenziell verdächtige Aktivitäten zu erkennen. Allerdings sind die Daten so komplex, dass die Banken eine große Anzahl von Analysten beschäftigen müssen, um manuell regelbasierte Warnmeldungen durchzugehen. Machine-Learning-Modelle, die auf historischen Transaktionsdaten geschult wurden, können verdächtige Muster identifizieren, die ein regelbasiertes System übersehen würde. Wenn in den historischen Trainingsdaten Transaktionen aufgrund von Bias-Attributen fälschlicherweise als kritisch eingestuft wurden und so andere kritische Transaktionen vernachlässigt wurden, kann dem mit Anti-Bias-Technologie vorgebeugt werden. Die Erklärungs- und Rückverfolgbarkeitsfunktionen, die durch Algorithmen wie MACEM möglich sind, helfen Banken zudem, mit sich ändernden Vorschriften Schritt zu halten und ermöglichen es Finanzkriminalitätsanalysten, die Gründe für die Alarmanalyse ihrer Modelle zu verstehen. So können letztendlich bessere Entscheidungen darüber getroffen werden, welche Warnungen zu verwerfen und welche zu eskalieren sind.

6.5 Literaturverzeichnis

[1] IBM (2017), IBM Watson Open Scale White Paper, S. 1 f.
[2] Lohia et al. (2018),»Bias Mitigation Post-processing for Individual and Group Fairness«, S. 2
[3] Ebenda
[4] Lohia et al. (2018), Bias Mitigation Post-processing for Individual and Group Fairness, S. 2
[5] Lohia et al. (2018), Bias Mitigation Post-processing for Individual and Group Fairness, S. 3
[6] Dhurandhar et al. (2019), Model Agnostic Contrastive Explanations for Structured Data, S. 1
[7] Unbekanntes, intransparentes Model M und optional die Basiswerte b, ein Koeffizient c, der zugelassene Datenbereich X und ein Autoencoder AE
[8] Dhurandhar et al. (2019), Model Agnostic Contrastive Explanations for Structured Data, S. 2
[9] Dhurandhar et al. (2019), Model Agnostic Contrastive Explanations for Structured Data, S. 3 f.
[10] ↗ https://github.ibm.com/tejaswinip/CEM

7 Extraktion von Erklärungen zu Produktionsprozessen aus künstlichen Neuronalen Netzen

7 Extraktion von Erklärungen zu Produktionsprozessen aus künstlichen Neuronalen Netzen

Nina Schaaf, Marco Huber

7.1 Einleitung

In den letzten Jahren hat das maschinelle Lernen (ML) als Teildisziplin der künstlichen Intelligenz in einer Vielzahl von Bereichen, wie etwa der Fertigung, der Medizin oder im Finanzbereich, zunehmend an Bedeutung gewonnen. Hohe Verbreitung hat dabei das sogenannte Deep Learning gefunden, also die Verwendung von tiefen künstlichen Neuronalen Netzen (KNN), welche mittels großer Datensätze für eine bestimmte Aufgabe trainiert werden. Allerdings stellen viele ML-Verfahren, und hierzu zählen auch tiefe KNN, eine Art »Black Box« dar, d. h. getroffene Entscheidungen dieser Verfahren sind aufgrund komplexer interner Prozesse für den Menschen oft nicht nachvollziehbar. Der Mangel an Transparenz und Nachvollziehbarkeit ist für einige Anwendungen unkritisch. Beispiele sind Filmempfehlungen auf Online-Plattformen oder maschinelle Textübersetzungen. Für viele Anwendungsfälle jedoch besteht ein berechtigtes Interesse an der Herstellung von Transparenz und Erklärbarkeit.

Im Bereich der Produktionstechnik kann hier als Beispiel der Tiefdruck aufgeführt werden. Dies ist ein Druckverfahren, bei dem die Farbe über gravierte Druckzylinder auf das Papier übertragen wird. Druckereibetreiber finden großen Nutzen darin, mittels einer Simulation zukünftige Qualitätswerte, basierend auf den aktuellen Maschinenparametern, vorherzusagen. Für die Erstellung eines dafür notwendigen Tiefdruckmaschinenmodells, das Zusammenhänge zwischen verschiedenen Eingangskonfigurationen und der Qualität des Drucks lernt, können KNNs eingesetzt werden. Ein KNN liefert allerdings Vorhersagen, ohne innere Entscheidungsprozesse darzulegen. Ein Instrument, dieser mangelnden Transparenz zu begegnen, ist der Einsatz von interpretierbaren Stellvertretermodellen, wie etwa Entscheidungsbäumen. Durch Interpretation eines an das KNN angepassten Entscheidungsbaumes können allgemeine Zusammenhänge abgeleitet werden, wie z. B. welche Kombination von Eingabeparametern zu welchem Prozessergebnis führt. Konkret tritt im Tiefdruck zeitweise sogenanntes »Banding« auf, was sich in Streifen im Druckbild äußert [1]. Wird ein solches Streifenmuster entdeckt, so muss der Druck angehalten und eventuell der Druckzylinder gewechselt werden, was eine erhebliche Prozessverzögerung zur Folge hat. Rückschlüsse, die aus der Interpretation eines aus dem KNN extrahierten Entscheidungsbaumes gewonnen werden, können somit helfen, Prozesswissen abzuleiten und anzureichern.

In diesem Beitrag wird ein praktikabler Ansatz zur Erlangung der Erklärbarkeit von KNN unter Verwendung eines interpretierbaren Ersatzmodells, basierend auf Entscheidungsbäumen, dargestellt [2]. Einfach einen Entscheidungsbaum aus einem trainierten KNN zu extrahieren, führt in der Regel zu unbefriedigenden Ergebnissen in Bezug auf die Genauigkeit und Wiederga-

betreue. Wird allerdings während des Trainings eine L1-orthogonale Regularisierung angewandt, so wird die Genauigkeit des KNN beibehalten, zugleich kann es jedoch sehr gut von kleinen Entscheidungsbäumen angenähert werden. Tests mit verschiedenen Datensätzen bestätigen, dass L1-orthogonale Regularisierung Modelle mit geringerer Komplexität und gleichzeitig höherer Wiedergabetreue liefert als andere Regularisierungstechniken.

7.2 Problemverständnis

Eine Möglichkeit, *Erklärbarkeit* durchzusetzen ist es, ML-Modelle[1] zu verwenden, bei denen Erklärbarkeit ein wesentlicher Bestandteil des Designs ist. *Ante-hoc-Modelle* – gerne auch als »White Box«-Modelle bezeichnet – sind so konzipiert, dass sie von Natur aus erklärbar sind. Beispiele hierfür sind die logistische Regression, regelbasierte Systeme oder Entscheidungsbäume. Im Gegensatz dazu bezieht sich die *Post-Hoc-Erklärbarkeit* auf das nachträgliche Herstellen von Transparenz. Methoden zur Herstellung von Erklärbarkeit können *modellspezifisch* sein, also nur für eine Art von ML-Modellen funktionieren, oder *modellagnostisch* und somit für verschiedene Modellarten anwendbar. Bei der Suche nach einem geeigneten Erkläransatz ist zudem der Geltungsbereich der erzeugten Erklärung zu beachten. *Globale* Erklärbarkeit – manchmal auch *Modellerklärung* genannt – setzt voraus, dass die erzeugten Erklärungen für das Modell als Ganzes gelten. Globale Erklärbarkeit ermöglicht etwa, Einblicke in Nichtlinearitäten oder Wechselwirkungen in den Eingabedaten zu erlangen. Im Gegensatz dazu zielt die *lokale* oder *Ausgabeerklärbarkeit* darauf ab, die Gründe für die Entstehung einer einzelnen Prognose (oder einer Gruppe von Prognosen) des untersuchten ML-Modelles zu verstehen. Für beides, also globale wie lokale Erklärbarkeit, wird typischerweise ein interpretierbares Stellvertreter- oder *Surrogatmodell* erzeugt. Die sogenannte *Modellprüfung* kommt ohne dieses Surrogatmodell aus und erzeugt eine Erklärung direkt aus dem Black-Box-Modell. In Abbildung 25 sind die unterschiedlichen Abläufe zur Erzeugung von nachvollziehbaren Erklärungen schematisch dargestellt.

7.2.1 Grundidee

Im speziellen Fall tiefer KNN ist in den letzten Jahren eine Vielzahl an Verfahren zur Erzeugung von Transparenz und Erklärbarkeit entstanden. Üblich sind hierbei lokale Ansätze zur Erzeugung von Visualisierungen wie etwa Heatmaps, welche die für eine Ausgabe des Netzes maßgeblichen Elemente der Eingabedaten visuell hervorheben. Unter den globalen Ansätzen ist die Regelextraktion sehr beliebt. Regelextraktionsverfahren für KNN folgen der Idee der Ableitung einfacher, menschennachvollziehbarer Regeln, um sich den internen Entscheidungsprozessen des Netzes anzunähern und so Erklärungsansätze zu bieten. Derart abgeleitete Regeln können in unterschiedlicher Form präsentiert werden, z. B. als Entscheidungsbäume, Entscheidungstabellen oder einfache Regeln der Art: **IF** Druckgeschwindigkeit = hoch **and** Viskosität ≤ 47 **THEN** kein Banding

1 Es wird zwischen ML-Modell und ML-Verfahren unterschieden. ML-Verfahren verarbeiten Daten und trainieren damit ein ML-Modell, welches dann auf neuen, unbekannten Daten angewendet werden kann. Lineare Modelle etwa können durch eine Vielzahl von Verfahren erzeugt werden, z. B. logistische Regression, Perzeptron oder lineare Support-Vektor-Maschine.

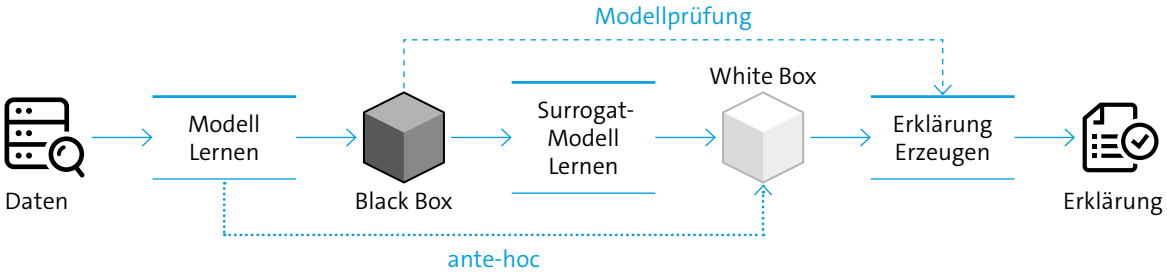


Abbildung 25: Von einem Black-Box-Modell zu einer Erklärung. Entweder durch direkte Erzeugung einer Erklärung aus dem Modell (gepunktete Linie, Modellprüfung) oder durch die post-hoc Extraktion eines interpretierbaren Stellvertretermodells (Surrogat), gefolgt von der Erzeugung von Erklärungen. Bei der ante-hoc Vorgehensweise (gestrichelte Linie) wird direkt ein interpretierbares ML-Modell erzeugt.

In diesem Beitrag wird die Regelextraktion in Form von Entscheidungsbäumen für eine weit verbreitete Art tiefer KNN namens *multi-layer perceptrons* (MLP) betrachtet. Diese Netze stellen gewissermaßen den Archetyp eines KNN dar. Neuronen sind hier in Schichten angeordnet, wobei die Neuronen einer Schicht lediglich mit Neuronen der nachfolgenden Schicht verbunden sind.

Die grundlegende Idee dabei ist, das Training von tiefen MLP dahingehend zu beeinflussen, dass die post-hoc Erzeugung eines Entscheidungsbaums deutlich verbessert wird. Hierzu wird das Optimierungsproblem, welches dem Trainieren eines MLP zugrunde liegt, gezielt verändert. Es sollen MLPs begünstigt werden, deren Entscheidungsgrenzen durch kleine Entscheidungsbäume viel einfacher angenähert werden können. Hierdurch wird dem Umstand Rechnung getragen, dass Entscheidungsbäume zwar allgemeinhin als White-Box-Modell anerkannt sind, deren Nachvollziehbarkeit aber mit zunehmender Größe, d. h. mit zunehmender Tiefe, abnimmt. Auf diese Weise wird globale Erklärbarkeit von tiefen MLP hergestellt.

7.2.2 Klassifikation

Es wird im Folgenden ein Klassifikationsproblem betrachtet. Zum Training des MLP steht ein (Trainings-)Datensatz $D = \{x_n, y_n\}_{n=1}^{N}$, bestehend aus Instanzen (x_n, y_n) mit Merkmalsvektor $x_n \in R^d$ und entsprechendem Klassenlabel $y_n \in \{1,2, ..., L\}$ zur Verfügung. Die einzelnen Elemente des Vektors x werden als Merkmale oder Attribute bezeichnet. Für den Anwendungsfall des Tiefdrucks könnte x beispielsweise Maschinen- und Auftragsparameter enthalten, während y beschreibt, ob Banding aufgetreten ist oder nicht. Im Allgemeinen lassen sich Klassifikationsprobleme als Optimierungsproblem gemäß

$$\min_{\theta} \sum_{n=1} E(y_n - f(x_n; \theta)) \qquad (1)$$

darstellen, wobei die nichtlineare Funktion $f(x;\theta) = y$ den Klassifikator darstellt. Bei diesem Problem geht es darum, die Parameter θ des Klassifikators derart zu optimieren, dass der Klassifikationsfehler E zwischen dem bekannten Label y und der Prognose des Klassifikators kleinstmöglich ist. Das Lösen des Optimierungsproblems wird gemeinhin als *Training* bezeichnet.

Im vorliegenden Fall dient als Klassifikator ein MLP, dessen Parameter θ die Gewichte der Verbindungen zwischen allen Neuronen umfasst. Genauer gilt $\theta = \{W_l\}_{l=1}^{L}$, wobei die Gewichtsmatrix W_l alle Gewichte der Neuronen von Schicht l – 1 zu Neuronen der l-ten Schicht enthält. Eine Spalte w_i der Gewichtsmatrix W_l enthält die Gewichte der Verbindungen aller Neuronen der Schicht l – 1 zum i-ten Neuron der Schicht l. Eine derartige Spalte wird im Folgenden als Gewichtsvektor bezeichnet.

7.2.3 Entscheidungsbaum

In diesem Beitrag werden Entscheidungsbäume verwendet, um Erklärungen aus einem MLP zu extrahieren. Ein Entscheidungsbaum besteht aus internen Knoten und Blattknoten (siehe Abbildung 28 für einen beispielhaften Entscheidungsbaum mit sieben internen Knoten und acht Blattknoten). An internen Knoten werden die Werte eines Merkmals der Eingabe x überprüft, während an den Blattknoten die Zuordnung zu einer der L Klassen stattfindet. Um einen Merkmalsvektor x zu klassifizieren, wird der Baum von oben nach unten durchlaufen. Wenn ein interner Knoten erreicht wird und die Überprüfung bestanden ist, wird der linke Teilbaum verfolgt, ansonsten der rechte. Dieses Verfahren wiederholt sich an jedem der folgenden internen Knoten, bis ein Blattknoten erreicht ist. Jeder Weg von der Wurzel des Baumes zu einem Blattknoten kann in eine If-Then-Regel wie die oben Gezeigte übersetzt werden. Die Bedingungen der If-Klausel entsprechen den Überprüfungen an den einzelnen internen Knoten entlang des Weges.

7.3 Extraktion von Entscheidungsbäumen

Um einen Entscheidungsbaum zu extrahieren, wird ein bereits trainiertes MLP verwendet, um dessen Prognosen des Klassenlabels $\hat{y}_n$ für jeden Merkmalsvektor x_n zu bestimmen. Dadurch ergibt sich ein neuer Datensatz $D' = \{x_n, \hat{y}_n\}_{n=1}^{N}$, welcher sich vom ursprünglichen Datensatz D lediglich in den Klassenlabels unterscheidet. Das MLP fungiert hier gewissermaßen als »Orakel«. Mittels des neuen Datensatzes wird nun ein Entscheidungsbaum trainiert, der möglichst mit den Prognosen des MLP übereinstimmen soll. Ist dies der Fall, kann der für den Menschen nachvollziehbare Entscheidungsbaum zur Erklärung des MLP herangezogen werden.

7.3.1 Regularisierung

Das zuvor beschriebene naive »Fitting« des Entscheidungsbaums an ein MLP führt in der Praxis leider zu wenig zufriedenstellenden Ergebnissen:

- Die Übereinstimmung zwischen MLP und Entscheidungsbaum ist nicht ausreichend.
- Die Übereinstimmung der Prognosen des Entscheidungsbaums mit den Trainingsdaten D ist ebenfalls nicht ausreichend.
- Die Entscheidungsbäume werden zu groß und damit für den Menschen schwer nachvollziehbar.

Um die Extraktion zu verbessern, wird das ursprüngliche Optimierungsproblem (1) um einen sogenannten Regularisierungsterm Ω ergänzt, so dass nun

$$\min_{\theta} \sum_{n=1} E(y_n - f(x_n; \theta)) + \lambda \cdot \Omega \qquad (2)$$

gilt. Die Stärke bzw. der Einfluss der Regularisierung wird durch den Parameter $\lambda \in R^+$ gesteuert. Die Verwendung einer Regularisierung beim Training von KNNs ist durchaus üblich und dient normalerweise der Vermeidung der sogenannten Überanpassung (engl. Overfitting) an die Trainingsdaten. Hier soll nun die Regularisierung vielmehr zur besseren Anpassung eines Entscheidungsbaums an ein MLP dienen.

Wird $\lambda = 0$ gesetzt, erhält man das oben beschriebene naive Verfahren mit den bekannten mäßigen Ergebnissen. Wu et al. [3] schlagen daher einen neuartigen Regularisierungsterm namens *Baumregularisierung* vor, bei dem Ω die durchschnittliche Weglänge (engl. average path length *APL*) eines Entscheidungsbaums misst. Hiermit soll das Training des MLP derart beeinflusst werden, dass ein extrahierter Entscheidungsbaum möglichst klein und somit gut interpretierbar ist. Allerdings kann die APL nicht in geschlossener Form berechnet werden: es muss zunächst ein Entscheidungsbaum erzeugt werden, um die APL zu bestimmen. Hierdurch ist die APL auch nicht differenzierbar, was das Training des MLP deutlich erschwert. Um diese Probleme zu umgehen, nutzen Wu et al. ein weiteres MLP, welches zum Schätzen der APL dient. Es ist offensichtlich, dass das gleichzeitige Trainieren zweier voneinander abhängiger MLPs mit erheblichem Rechenaufwand und einer sorgfältigen Parametereinstellung verbunden ist.

7.3.2 Spärlichkeit und Orthogonalität

Um die Einschränkungen der Baumregularisierung zu vermeiden, aber gleichzeitig dessen ansprechende Idee beizubehalten, wird ein neuer Regularisierungsterm vorgeschlagen, welcher die *Spärlichkeit* (engl. sparseness) und *Orthogonalität* der Gewichtsvektoren in den Gewichtsmatrizen fördert. Dieser Term ist geschlossen berechenbar, einfach zu implementieren und differenzierbar. Zudem drängt er das MLP dazu, Entscheidungsgrenzen zu bilden, welche durch einen Entscheidungsbaum leicht approximiert werden können. Darüber hinaus soll der Entscheidungsbaum kleiner sein als ein solcher, welcher sich entsprechend des naiven Vorgehens, also ohne Regularisierung, ergibt. Jedoch soll durch die Verwendung des neuartigen Regularisierungsterms die Prognosegenauigkeit des MLPs nicht signifikant abnehmen.

Zur Veranschaulichung der Wirkungsweise des neuen Regularisierungsterms wird im Folgenden der Zusammenhang zwischen Gewichtsvektor und Entscheidungsgrenze betrachtet. Es ist bekannt, dass ein Gewichtsvektor den Normalenvektor einer linearen Entscheidungsgrenze repräsentiert. Durch die Forcierung von Spärlichkeit, bei der viele oder sogar alle Elemente eines Gewichtsvektors bis auf eines nahe Null sind, wird die lineare Entscheidungsgrenze achsparallel. Diese Darstellung harmoniert gut mit Entscheidungsbäumen, da deren Entscheidungsgrenzen sich ebenfalls aus achsparallelen Segmenten zusammensetzen. Jedes Segment entspricht dabei einem internen Knoten, welcher den Merkmalsraum in achsparallele Hyperebenen unterteilt, wobei jede Hyperebene einer Klasse zugeordnet ist.

Außerdem gilt es zu vermeiden, dass zu viele Entscheidungsgrenzen entstehen, die (nahezu) parallel zueinander ausgerichtet sind. Dies entspräche vielen Gewichtsvektoren mit einer ähnlichen Wertebelegung, was die Prognosefähigkeit des MLP stark einschränken würde. Deshalb wird die Spärlichkeit mit Orthogonalität kombiniert, d. h. die Gewichtsvektoren werden »ermutigt«, paarweise orthogonal zueinander zu sein. Durch die Kombination einer spärlichen mit einer orthogonalen Regularisierung während des MLP-Trainings wird beabsichtigt, dass eine Gewichtsmatrix eine kleine Zahl von Einträgen ungleich Null enthält (Spärlichkeit), aber dennoch eine breite Abdeckung an Merkmalen gewährleistet (Orthogonalität). Diese Art der Regularisierung drängt MLPs dazu, Entscheidungsgrenzen anzunehmen, die denen von Entscheidungsbäumen ähnlich sind und somit besser approximiert werden können.

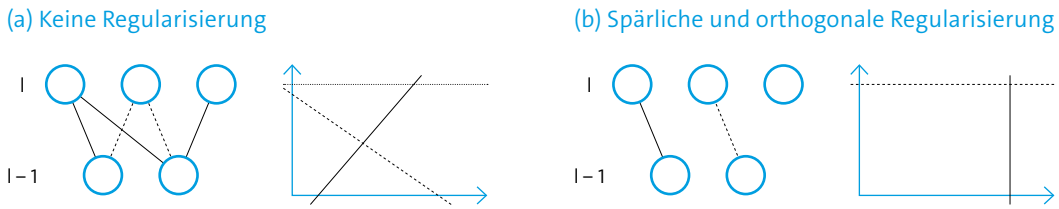


Abbildung 26: Auswirkung der Kombination aus spärlicher und orthogonaler Regularisierung auf die Gewichtsvektoren eines MLP. In jeder Teilabbildung ist links ein MLP und rechts die Entscheidungsgrenzen der Neuronen der oberen Schicht dargestellt.

Abbildung 26 zeigt die beabsichtigten Auswirkungen auf die Gewichtsvektoren. Es werden die Kanten zwischen zwei aufeinanderfolgenden Schichten eines MLP schematisch dargestellt. Die Gewichte auf den Verbindungen aller Neuronen der Schicht $l-1$ zu einem Neuron der Schicht l bilden die Elemente des Gewichtsvektors. Keine Verbindung entspricht einem Gewicht gleich Null. Wenn ein Neuron der Schicht l mehr als eine Verbindung zur vorhergehenden Schicht hat, ist der Gewichtsvektor nicht spärlich bzw. die Entscheidungsgrenze nicht achsparallel, wie in Abbildung 26a zu sehen ist. Das Netzwerk in Abbildung 26b hingegen ist sowohl spärlich (wenige Verbindungen) und besteht auch aus orthogonalen Gewichtsvektoren (Verbindungen zu verschiedenen Neuronen in Schicht $l-1$).

7.3.3 Umsetzung

Um diese Eigenschaften in (2) einzubringen, wird der kombinierte Regularisierungsterm

$$\Omega(\theta) = \lambda_1 \cdot \Omega_1(\theta) + \lambda_{orth} \cdot \Omega_{orth}(\theta) \qquad (3)$$

genutzt. Hierbei forciert der Teilterm Ω_1 die Spärlichkeit, was durch die L1-Norm der Gewichtsvektoren erreicht wird. Der Teilterm Ω_{orth} dient der Orthogonalität und bewertet die Abweichung der Grammatrix $G_l = W_l^T \cdot W_l$ von der Einheitsmatrix. Anstelle eines einzelnen Parameters λ wird der Einfluss von Ω_1 und Ω_{orth} durch zwei unabhängige Parameter gesteuert. Hierdurch kann eine ausgewogene Wahl zwischen beiden Größen getroffen werden.

7.4 Ergebnisse

Die Leistungsfähigkeit der vorgestellten Regularisierung, im Folgenden durch SO (für Spärlich und Orthogonal) abgekürzt, soll anhand von vier Open-Source-Datensätzen aufgezeigt werden:

- **Iris [4]:** Hierbei handelt es sich um einen Datensatz von Schwertlilien mit 150 Instanzen und jeweils vier Attributen. Gemessen wurden dabei jeweils die Breite und die Länge des Kelchblatts (Sepalum) sowie des Kronblatts (Petalum) in Zentimeter. Es wird zwischen drei Arten von Schwertlilien unterschieden. Dieser Datensatz ist ein sehr gängiger Benchmark im Bereich der ML-Forschung.

- **Cylinder Bands [4]:** Dieser Datensatz enthält Prozessparameter von Tiefdruckaufträgen, bestehend aus 540 Instanzen mit je 40 Attributen. Es soll eine Auftragsklassifikation durchgeführt werden, die Verzögerungen, verursacht durch Banding, identifiziert.

- **Mushroom [4]:** Dieser Datensatz enthält Beschreibungen von hypothetischen Proben von 23 Pilzarten. Jede Art wird als definitiv essbar, definitiv giftig oder von unbekannter Essbarkeit identifiziert. Es ist bekannt, dass es keine einfache Regel für die Bestimmung der Essbarkeit eines Pilzes gibt. Der Datensatz enthält 8.124 Instanzen mit jeweils 22 Merkmalen.

- **Diabetes [5]:** Dieser Datensatz repräsentiert die klinische Versorgung an 130 US-Krankenhäusern und zugehöriger Liefernetzwerke in den Jahren 1999–2008 und wird verwendet, um vorherzusagen, ob ein Patient, der wegen Diabetes in Behandlung war, binnen 30 Tagen wieder in ein Krankenhaus eingeliefert wird. Der Datensatz enthält 100.000 Instanzen mit 50 Merkmalen.

Für den Cylinder-Bands-, den Iris- und den Mushroom-Datensatz wird ein MLP mit einer verdeckten Schicht mit acht (Iris, Cylinder Bands) bzw. 16 (Mushroom) Neuronen verwendet. Ein solches KNN wird als »flach« bezeichnet. Der deutlich komplexere Diabetes-Datensatz bedarf eines tiefen MLPs mit drei verdeckten Schichten mit 32, 16 bzw. acht Neuronen. Als Aktivierungsfunktion in den verdeckten Schichten aller MLPs wird ReLU verwendet.

Zum Vergleich werden neben der SO-Regularisierung zudem noch folgende weitere Ansätze verwendet:

- **Nativ:** Entscheidungsbaum, der direkt auf den Daten gelernt wird, also nicht durch Extraktion aus einem MLP. Dieser dient als Referenz.
- **MLP:** Es wird ein MLP ohne Regularisierung trainiert. Dieses dient als weitere Referenz.
- **Baumreg.:** Die von Wu et al. vorgeschlagene Regularisierung.
- **S:** Extraktion nur durch Verwendung der Spärlichkeits-Regularisierung.
- **O:** Extraktion nur durch Verwendung der Orthogonalitäts-Regularisierung.

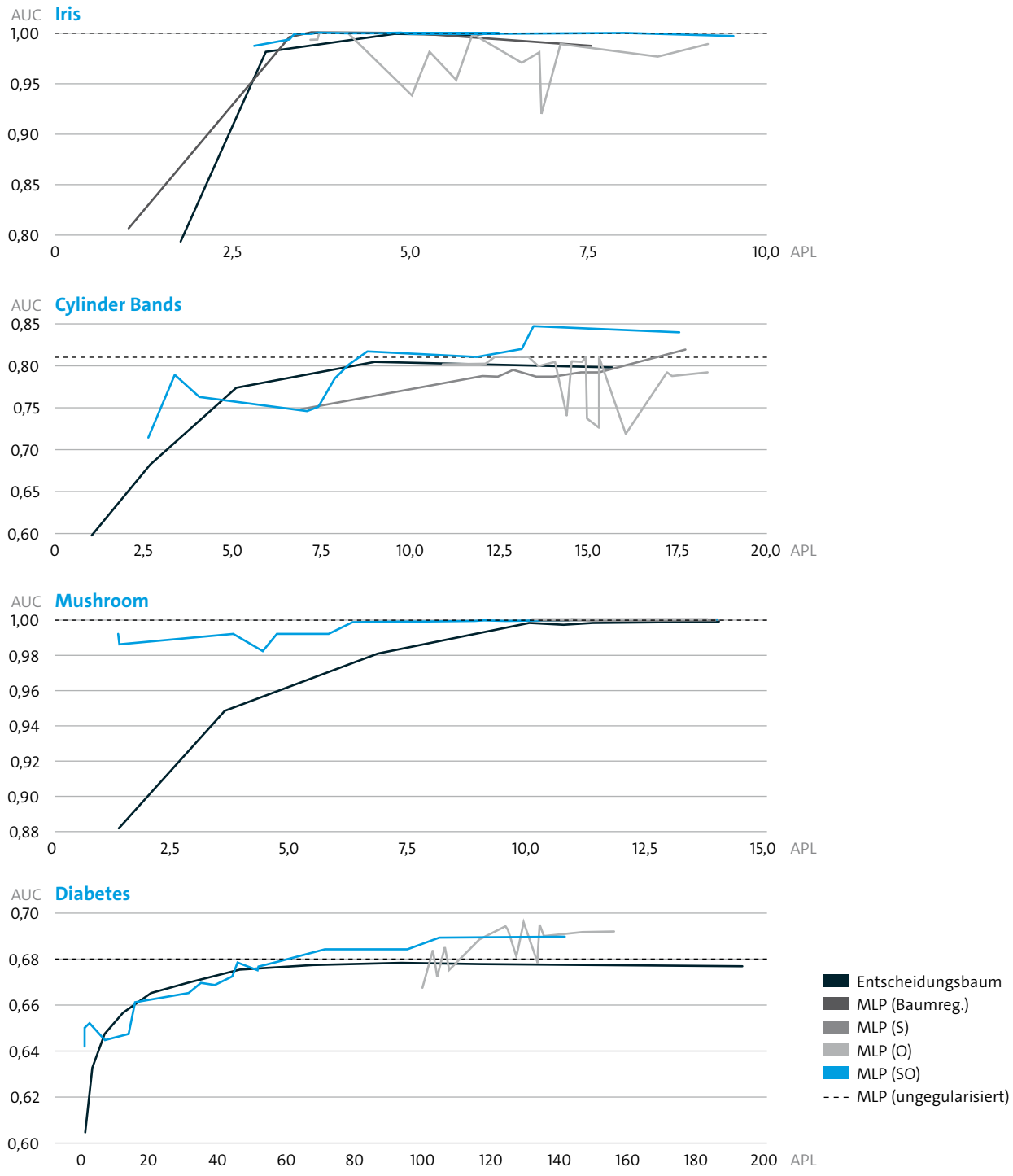


Abbildung 27: Entwicklung der Prognosegenauigkeit (AUC)

In Abbildung 27 ist die Prognosegenauigkeit der trainierten MLPs über die Komplexität der extrahierten Entscheidungsbäume aufgetragen. Die Prognosegenauigkeit wird hierbei durch den AUC-Wert[2] angegeben. Dieser Wert liegt zwischen Null und Eins, wobei ein höherer Wert besser ist. Die Komplexität der Bäume wird durch den APL-Wert angegeben. Je kleiner dieser Wert, umso kleiner ist auch der entsprechende Entscheidungsbaum. Zur Berechnung der einzelnen AUC-APL-Wertepaare wurden die Regularisierungsparameter der einzelnen Verfahren variiert.

Beim Iris-Datensatz ist gut zu erkennen, dass alle Methoden, abgesehen von der O-Regularisierung, eine nahezu optimale Prognosegenauigkeit bei gleichzeitig kleinen Entscheidungsbäumen erreichen. Jedoch erreicht die SO-Regularisierung am ehesten den besten Kompromiss aus Baumgröße und Prognosegenauigkeit. Da das Klassifikationsproblem für diesen Datensatz eher einfach ist, kann auch mit einem nativen Entscheidungsbaum ein sehr gutes Ergebnis erzielt werden. Die Anwendung der Baumregularisierung war für die restlichen Datensätze aufgrund der komplexen Parametereinstellung nicht möglich.

Der Cylinder-Bands- sowie der Mushroom-Datensatz erreichen eine hohe Prognosegenauigkeit, allerdings erst bei recht großen Entscheidungsbäumen. Hier zeigt sich deutlich der Vorteil der SO-Regularisierung, bei welcher eine hohe, wenn auch nicht perfekte Prognosegenauigkeit bereits mit einem kleinen APL-Wert einhergeht. Hierbei sind die Prognosen besser als bei einem nativen Entscheidungsbaum. Zusätzlich lässt sich beim Cylinder-Bands-Datensatz erkennen, dass die Vorhersagekraft eines SO-regularisierten MLPs bereits ab einer APL von ca. 8.5 die Vorhersagekraft aller anderen Verfahren übertrifft. In Abbildung 28 ist ein per SO-Regularisierung extrahierter Entscheidungsbaum für den Cylinder-Bands-Datensatz dargestellt. Dieser Baum ist von geringer Größe und daher für einen Menschen gut nachvollziehbar.

Die Ergebnisse zum Diabetes-Datensatz zeigen deutlich dessen Komplexität. Es bedarf verhältnismäßig großer Bäume, um eine annehmbare Prognosegenauigkeit zu erhalten. Während der native Entscheidungsbaum zwar nahe an das MLP ohne Regularisierung heranreicht, kann die Leistung dieser Referenz erst durch die Verwendung der vorgestellten SO-Regularisierung übertroffen werden. Dies gelingt dabei mit deutlich kleineren APL-Werten als dies mit der O-Regularisierung alleine der Fall ist.

In Tabelle 1 ist die sogenannte Wiedergabetreue (engl. Fidelity) der einzelnen Verfahren aufgeführt. Die Wiedergabetreue gibt an, wie sehr das MLP und der extrahierte Entscheidungsbaum in ihren Prognosen übereinstimmen. Ein hoher Wert und damit eine hohe Übereinstimmung sind wichtig, um den Entscheidungsbaum auch tatsächlich zur Erklärung des zugehörigen MLP heranziehen zu können. Zur Berechnung der Werte in Tabelle 1 wurden die Regularisierungsparameter so gewählt, dass die Prognosegenauigkeit der extrahierten Bäume vergleichbar mit der Prognosegenauigkeit eines unregularisierten MLPs ist. Es zeigt sich, dass mit der SO-Regularisierung eine hohe Wiedergabetreue für jeden Datensatz erzielt wird. Dabei sind die Werte stets gleich oder höher als die der anderen Verfahren (vgl. Cylinder Bands und Diabetes).

2 AUC = Area under the ROC-Curve

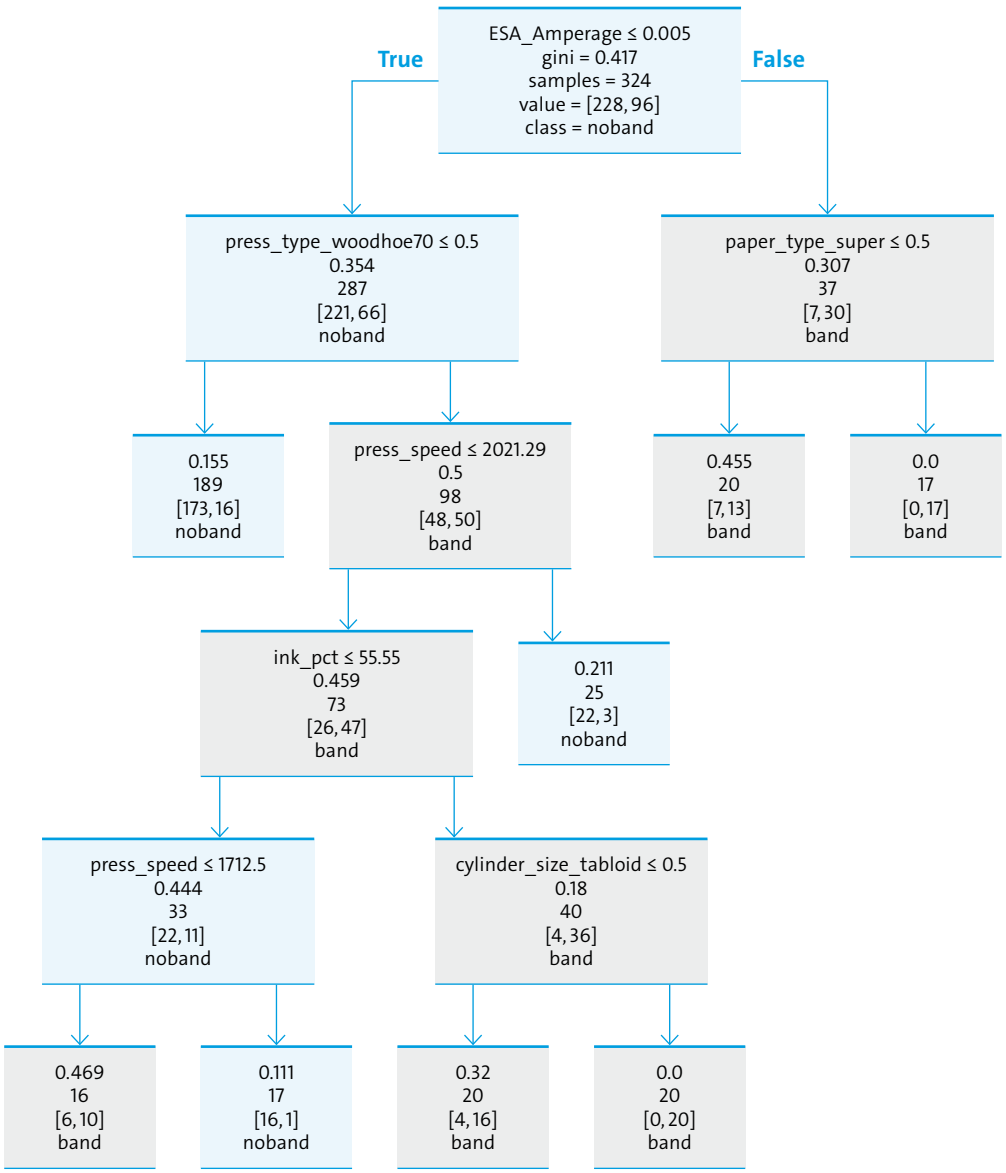


Abbildung 28: Entscheidungsbaum, extrahiert aus einem mittels SO-Regularisierung optimierten MLP. Der Baum hat eine APL von neun und einen Wiedergabetreuewert von 0,83.

Datensatz	SO	S	Baumreg.	unregularisiert
Iris	0,99 ± 0,02	0,99 ± 0,02	0,97 ± 0,02	0,96 ± 0,01
Cylinder Bands	0,80 ± 0,04	0,77 ± 0,04	–	0,73 ± 0,06
Mushroom	0,98 ± 0,00	0,98 ± 0,00	–	0,98 ± 0,00
Diabetes	0,92 ± 0,02	–	–	0,81 ± 0,01

Tabelle 1: Wiedergabetreue der extrahierten Entscheidungsbäume. Die Werte liegen zwischen Null und Eins. Es wurde eine 5-fache Kreuzvalidierung zur Berechnung der Werte angewendet.

7.5 Fazit

Eine geeignete Regularisierung des Trainings von KNN begünstigt die Extraktion von nachvollziehbaren Entscheidungsbäumen, welche zugleich mit den Prognosen des Netzes gut übereinstimmen. Diese Bäume liefern somit Erklärungen, welchen Nutzern verschiedener Interessengruppen Einblicke in die Entscheidungsfindung von Neuronalen Netzen in leicht verständlicher Form geben. Beispielsweise könnten solch einfache Entscheidungsbäume in der Fertigung dem Maschinenführer Hinweise darauf liefern, welche Prozessparameter maßgeblichen Einfluss auf das Druckergebnis haben. Dieses Wissen kann dann beim Druckprozess berücksichtigt werden und dadurch im besten Fall Prozessverzögerungen durch Banding verhindern.

Noch ist diese Form der Erklärungsfindung auf einfache KNN beschränkt. Zukünftige Forschungsarbeiten fokussieren sich auf komplexere Netztypen wie etwa Convolutional Neural Networks, welche insbesondere in der Bildverarbeitung sehr weit verbreitet sind.

7.6 Literaturverzeichnis

[1] ONYX Graphics Inc.: »Banding Issues With Wide Format Printing«, Salt Lake City, 2011.

[2] Schaaf, Nina et al.: »Enhancing Decision Tree based Interpretation of Deep Neural Networks through L1-Orthogonal Regularization«, in Proceedings of the 18th IEEE International Conference on Machine Learning and Applications (ICMLA), 2019.

[3] Wu, Mike et al.: »Beyond sparsity: Tree regularization of deep models for interpretability«, in AAAI, 2018.

[4] Dua, Dheeru und Graff, Casey: UCI Machine Learning Repository. ↗ **http://archive.ics.uci.edu/ml**. Irvine, CA: University of California, School of Information and Computer Science, 2019.

[5] Strack, Beata et al: »Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records«, in BioMed Research International, 2014.

8 Wissensextraktion aus Texten mittels semantischer KI

8 Wissensextraktion aus Texten mittels semantischer KI

Bernd Geiger

8.1 Problemstellung

Im Zuge der digitalen Transformation von Geschäftsprozessen entsteht die Notwendigkeit, in natürlicher Sprache verfasste Handlungsanweisungen, Verträge und vertragsähnliche Dokumente in computerausführbaren Code zu wandeln, um z. B. Workflow Automation oder Smart Contracts umzusetzen. Mit herkömmlichen (statistischen) KI-Methoden ist die Transformation oft unsicher und nicht nachvollziehbar, da es bei den Ursprungsdokumenten meist auf jedes Wort ankommt und Sätze linguistisch (z. B. aufgrund von Auslassungen und Grammatik) mehrdeutig sein können. Mit dem hier vorgestellten alternativen Ansatz der »semantischen KI« werden mittels Higher-Order Logic (HOL) Algorithmen mit nur geringem Einrichtungsaufwand natürlich sprachliche Texte in semantisch eindeutige und ausführbare Computer-Modelle transformiert. Dabei bleibt die Kausalitätsbeziehung zwischen Transformationsergebnis und Textursprung immer exakt nachvollziehbar – das Verfahren ist deshalb inhärent in seiner Arbeitsweise nachvollziehbar und somit auch auditierbar. Es wurde bislang auf englische und deutsche Texte angewandt.

Das hier vorgestellte Verfahren SemanticMatcher wurde für die Übersetzung von PDF-basierten Wartungshandbüchern für Flugzeuge in ausführbare BPMN-Abläufe getestet und optimiert. Die Adaption für natürlichsprachliche Dokumente aus anderen Kontexten geschieht durch einfache Modifikation der semantischen HOL-Algorithmen und wurde auch schon bei Vertragstexten erfolgreich umgesetzt. Wie im Bereich der Wartungshandbücher ist der Einrichtungsaufwand auch für Vertragstexte gering.

8.2 Einleitung

Allgemeine Formalien (Instruktionen, Vorschriften, etc.), die in textlicher Beschreibung vorliegen, sollen computerverständlich gemacht werden. Das Anwendungspotenzial hierfür ist enorm: alle menschenlesbaren Formalien sind typischerweise in natürlicher Sprache niedergeschrieben (Montageanleitungen, Verträge, gesetzliche Regulierungen, etc.). Das Bestreben, Prozesse zu automatisieren (z. B. robotergestützte Wartung, Regelkonformität bei Banktransaktionen, etc.) macht es erforderlich, textlich gefasste Formalien computerisiert ausführbar zu machen. Der klassische Weg hierfür ist die manuelle »Übersetzung« der textlichen Formalien in Skripte bzw. prozedurale Programmiersprachen. Diese Übertragung ist nicht nur sehr aufwendig, sondern auch fehleranfällig und eine formale Verifikation mit dem Inhaltsursprung ist nicht möglich.

Grundsätzlich handelt es sich bei der Aufgabenstellung um eine semantische Erweiterung des sogenannten Natural Language Processing (NLP). Der SemanticMatcher extrahiert nicht nur Einzelfakten, sondern verteilte Faktenzusammenhänge und somit exaktes Wissen.

Exaktes Wissen ist typischerweise dort notwendig, wo es sich um operativ kritische oder juristische Anwendungsdomänen handelt. Um die benötigte Exaktheit zu gewährleisten, ist es essenziell, dass das NLP-Verfahren sowohl algorithmisch als auch im Erzielen des Extraktionsergebnisses nachvollziehbar ist und dokumentiert, aus welchen Worten im Textteil und unter Zuhilfenahme welcher zusätzlichen Wissenskomponenten die computerisierte Repräsentation generiert wurde.

Der hier vorgestellte SemanticMatcher ist fokussiert auf Texte, die auf Prägnanz und Vermeidung linguistischer Redundanzen optimiert sind, sogenannte Controlled Natural Languages (CNLs). Dies trifft auf Texte in operativ kritischen bzw. in juristischen Anwendungsdomänen üblicherweise per Design zu, um menschliche Fehlinterpretationen zu vermeiden. Herkömmliche, statistische NLP-Verfahren wurden meist zu einem anderen Zweck entwickelt – sie sollen möglichst universell unterschiedliche Ausdrucksformen »verstehen« können und haben daher eine hohe (linguistische) Ausdruckstoleranz. Die Toleranz bedingt aber in Folge eine Detailunschärfe und eignet sich daher nicht zur präzisen Wissensextraktion[1,2].

8.3 Semantische KI

Was ist semantische KI und worin besteht der Unterschied zu KI basierend auf neuronalen Netzen?

Grundsätzlich kann man zwischen KI basierend auf statistischen Methoden (meist Neuronale Netze) und KI basierend auf logischen, mathematischen Modellen (logische Semantik) unterscheiden, wobei jede Methode auch mit der jeweils anderen erweitert werden kann. Allgemein betrachtet ist die Vorgehensweise die gleiche: ein unbekanntes Prüfmuster soll mit einem Standardmuster verifiziert bzw. falsifiziert werden. Bei statistischer KI ist das Standardmuster in der Regel im neuronalen Netz kodiert, bei semantischer KI in dem HOL-Modell.

1 Johnson (2009), How the statistical revolution changes (computational) linguistics, in Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?, S. 3–11.

2 Gao/Fodor/Kifer (2018), Knowledge authoring for rule-based reasoning, in OTM Confederated International Conferences »On the Move to Meaningful Internet Systems«, S. 461–480.

Wo wendet man die unterschiedlichen KI-Verfahren an?

Statistische KI-Verfahren werden bevorzugt dann angewendet, wenn Signale mit hohem statistischem Anteil vorhanden sind, d. h. wenn im Verhältnis zur Aussage der Erkennung (z. B. »das ist ein Stoppschild«) sehr viele Daten vorliegen. Die Komplexität in der Signalerkennung (z. B. Bild des geometrischen Verkehrsschildes) liegt im Wesentlichen darin begründet, dass das Signal aufgrund der Zerlegung in einzelne Pixel hochgradig redundant und verrauscht ist.[3]

Im Gegensatz dazu werden *semantische KI-Verfahren* primär dort angewendet, wo die Komplexität in den inhaltlichen Zusammenhängen begründet liegt. So müssen beispielsweise die über viele Textpassagen aufgeteilten Gesamtzusammenhänge eines Vertrages mit hoher Präzision verifizierbar erfasst werden. Dabei sind die Redundanz und das Rauschen des Signals (des zugrunde liegenden Textes) um Größenordnungen geringer als bei dem vorher genannten Bildbeispiel des Stoppschildes. Vergleichbares gilt für die knapp kommunizierten Instruktionen eines Wartungshandbuchs, die nur bei Kenntnis des detaillierten Kontextes Sinn ergeben, weil das Handbuch eben für den »Experten« geschrieben ist.

Semantische KI für hohe Präzision und Nachverfolgbarkeit der Ergebnisse

Je deterministischer das Datensignal ist und je geringer die Redundanzen, desto eher lassen sich handhabbare HOL-Modelle finden, mit denen sich die ursprüngliche Bedeutung des Signals rekonstruieren lässt. Ein Beispiel für eine sehr knappe und dadurch vermeintlich mehrdeutige (verrauschte) Nachrichtenübermittlung ist die folgende Kommunikation in deutscher Sprache: »Schließe noch die Waschmaschine an und dann machen wir Schluss für heute.«

Um diesen Zuruf des Meisters an den Gesellen roboterausführbar zu machen, muss man die ursprüngliche Intention/Semantik verstehen:

- Waschmaschine funktional mit der notwendigen Infrastruktur verbinden
- Es wird keine weitere Tätigkeit folgen

Das in dem Beispielsatz vorhandene »Restrauschen« – die linguistische Mehrdeutigkeit von »Anschließen« der Waschmaschine und »Schluss machen« – lässt sich durch a priori Wissen (das ebenso mittels HOL modelliert wird) eliminieren:

- Beim technischen Gerät Waschmaschine bedeutet die Tätigkeit »Anschließen« ein technisches Verbinden (im Gegensatz z. B. zum Anschließen eines Fahrrads),
- Die idiomische Phrase »Schluss machen«, bezogen auf ein zeitliches Intervall, bedeutet die Beendigung einer Tätigkeitsreihe.

3 Das Zeichen »Stoppschild« ist überlagert mit nicht auf das Stoppschild bezogenen Signalen im Sinne eines stochastischen Prozesses. Diese Signale sind z. B. Vegetation, Schmutz und sonstige Objekte, die von der Kamera mit erfasst werden.

Natürlich lassen sich Einzelinstruktionen und idiomatische Phrasen auch mit statistischen KI-Verfahren erkennen. Die notwendigen Trainingsphasen für die Zuordnung der jeweiligen Semantik zu den Einzelinstruktionen sind aufgrund der Vielfalt (alle möglichen Instruktionen müssen berücksichtigt werden) extrem aufwendig und bedürfen großer (oft nicht verfügbarer) Referenzdaten zum Trainieren des Systems. Darüber hinaus lassen sich die Ergebnisse statistischer KI-Verfahren aufgrund der inhärenten Blackbox-Natur der Systeme nicht verifizieren.

8.4 Semantische KI und Natürliche Sprache

Klassifizierung von natürlicher Sprache und deren kontrollierte Varianten (CNLs)

Unterschiedliche Ausdrucksformen einer natürlichen Sprache (z. B. Englisch) können nach dem PENS Klassifikationsschema eingeteilt werden[4]. Hierbei bedeutet: P = Precision, E = Expressiveness, N = Naturalness und S = Simplicity. Den einzelnen Parametern werden üblicherweise Werte von 1 = schwach bis 5 = stark zugeordnet. Die natürliche Sprache Englisch in ihrer allgemeinen Form wird mit $P^1E^5N^5S^1$ klassifiziert. Die CNL-Varianten Simplified (Technical) English[5] sowie Legislative Drafting Language[6] mit $P^2E^5N^5S^1$, die »Semantics Of Business Vocabulary and Rules« (SBVR) Sprache[7] mit $P^3E^4N^4S^2$. Alle Ausdrucksformen, die im Vergleich zur natürlichen englischen Sprache eine höhere Präzision haben, erreichen dies durch geregelte Einschränkungen (»Controlled«) z. B. bei Wortumfang, Grammatik, Satzbau und Idiomatik. Das zuvor Beschriebene ist unabhängig von der Sprache und kann genauso für die deutsche Sprache angewendet werden.

Modelle zur Wissensrepräsentation

Da das Ziel der semantischen Wissensextraktion in der Transformation des relevanten Textes (z. B. Wartungsanweisungen) in eine computerausführbare Form besteht, muss für die Wissensrepräsentation auf der Computerseite eine adäquate Ausdrucksform gewählt werden. Die Repräsentation muss eindeutig (Präzision = 5) sein (was in der Regel bei formalen Sprachen der Fall ist) und gleichzeitig eine hohe Expressivität haben. Prozedurale Sprachen wie Java können unter anderem aufgrund der Anforderungen zur Expressivität nicht berücksichtigt werden.

4 Kuhn (2014), A survey and classification of controlled natural languages, Computational Linguistics, 40(1), S. 121–170.

5 Aerospace and Defence Industry Association of Europe (2017), SIMPLIFIED TECHNICAL ENGLISH, Specification ASD-STE100, Issue 7.

6 Massachusetts Senate (2003), Legislative, Drafting and Legal Manual, third edition.

7 Bollen (2008), SBVR: A fact-oriented OMG standard, in OTM Confederated International Conferences »On the Move to Meaningful Internet Systems«, S. 718–727.

Higher-Order Logic (HOL) zur Wissensrepräsentation zu verwenden ist naheliegend, da dafür effiziente computerbasierte Umsetzungen existieren. In unserem Fall wird die industriell erprobte HOL-Sprache ObjectLogic[8,9,10] verwendet. ObjectLogic wird mit $P^5E^5N^2S^3$ klassifiziert. Prinzipiell zeichnen sich HOL-Sprachen durch eine hohe Expressivität aus. Gleichzeitig erlaubt die vergleichsweise »gute« Einfachheit mit S = 3 die Anwendung von ObjectLogic auch für Nichtmathematiker

Um ein Beispiel zu geben, wie einfach ObjectLogic eingesetzt werden kann, müssen zunächst die notwendigen Grundelemente der Sprache benannt werden:

Elemente	Notation
Funktionen	Head :- body
Frames, Maps, Lists	I[P → R], [P → R], [a, b, c]
Instanzen von Klassen	I : C
Vererbung von Klassen und Properties	SC :: C, SP << P
Prädikate	$P(n_1, ..., n_m)$
Quantoren	FORALL, EXIST
Junktoren	AND, OR, NOT, ->, <-, <->

Tabelle 2: Grundelemente ObjectLogic

ObjectLogic beinhaltet beliebig verschachtelte Objekte, Frame-Atome (F-Atome), F-Moleküle, parametrisierbare Prädikate, Attribute und Mengen- bzw. Logik-Funktionen, mit diesen Elementen kann man beliebige Realitäten beschreiben. Die Auswertung erfolgt in unserem Fall über den Interpreter (der »Reasoner«) OntoBroker (von semafora systems GmbH).

8 Kifer/Lausen/Wu (1995), Logical foundations of object-oriented and frame-based languages, Journal of the ACM (JACM), 42(4), S. 741–843.

9 Angele/Kifer/Lausen (2009), Ontologies in F-logic, in Handbook on Ontologies, S. 45–70.

10 Maier et al. (2018), Datalog: Concepts, History, and Outlook, Declarative Logic Programming: Theory, Systems, and Applications, S. 3–100.

8.5 Das Wissensmodell

Ein Wissensmodell für technische Instruktionen hat angelehnt an das BPMN-Modell[11] folgende allgemeine Charakteristika (die man je nach Anwendung verfeinern oder weiter reduzieren kann):

- Aktivität
- Voraussetzung, unter der die Aktivität stattfinden darf
- Agent, der die Aktivität ausführt
- Beschreibung der Dynamik der Aktivität:
  - Objekt 1 wird in Relation zu Objekt 2 verändert
  - Verwendung von Hilfsmitteln bei der Umsetzung der Aktivität
  - Zustandsbeschreibung vor und nach der Aktivität von Objekt 1 und Objekt 2 (pre/post state)
- Reihenfolge der Aktivitäten

Das Wissensmodell hat eine grammatikalische Repräsentation (für die jeweils verwendete natürliche Sprache), die sogenannten semantischen N-Gramme[12], die im Folgenden für den Matching-Prozess eingesetzt werden.

Die konkreten Informationen, um das Modell zu füllen, befinden sich in den Texten der Wartungsanweisungen. Dieses zu extrahierende Wissen liegt in einer Aneinanderreihung von Grammatiksegmenten vor, die Text N-Gramme. Nun werden im Zuge des »semantic matching« die abstrakten Konzepte der semantischen N-Gramme des Modells mit den konkreten Text N-Grammen aus den Wartungsanweisungen gepaart. Bei erfolgreicher Paarung wird das Modell mit den konkreten Werten (in diesem Fall Worten) gefüllt. Auf diese Weise entsteht aus dem abstrakten Modell ein konkretes detailliertes Abbild der Anweisung.

11 Chinosi/Trombetta (2012), BPMN: An introduction to the standard, Computer Standards & Interfaces, 34(1), S. 124–134.

12 Wikipedia, (Retrieved July 21, 2019), ↗ **https://en.wikipedia.org/wiki/N-gram**.

In der HOL/ObjectLogic-Darstellung sieht das in seiner für den Anwendungsfall allgemeinsten Form wie nachfolgend aus (mit Schattierungen illustriert für die ObjectLogic-Elemente Konzepte, Instanzen, Properties und Einzelwerten). In dem Modell ist berücksichtigt, dass eine Instruktion aus mehreren Einzelschritten (»steps«) bestehen kann:

```
?InstructionID:Instruction[
        Act,
        next→?InstructionID:Instruction,
        prior→?InstructionID:Instruction,
        serial→?Serial,
        condition→?StateID:State[
                id→?ID,
        state→?State,
                stateSpecifier→?StateSpecifier,
                target→?Target,
                targetSpecifier→?TargetSpecifier
        ],
        step→?StepID:Step[ {AlternativeStep}
                id→?ID,
                serial→?Serial,
                next→?StepID:Step,
                prior→?StepID:Step,
                act→?Act,
        actor→?Actor,
                actZone→?ActZone,
                actSpecifier→?ActSpecifier,
                actDirection→?ActDirection,
                actObject→?ActObject,
        actObjectSpecifier→?ActObjectSpecifier,
                actTool→?ActTool,
                actToolSpecifier→?ActToolSpecifier,
                actConsumable→?ActConsumable,
         actConsumableSpecifier→?ActConsumableSpecifier,
        ]
        step→?StepID:Step[
                target→?Target,
                targetSpecifier→?TargetSpecifier,
                targetPostState→?StateID:State[...],
                purpose→?StepID:Step[...]
        ]
]
```

8.6 Der Matching-Prozess zur Wissensextraktion

Schritt 0: Initialisierung

Folgende Komponenten werden vorbereitend für die sechs im Wesentlichen automatischen Schritte des Matchings, also der Wissensextraktion benötigt:

- Die Textdatei mit der Satzsammlung. Optional: ein Layout-Parsing des Ursprungstextes (siehe Schritt 6).
- Eine WordNet[13]-Variante für die jeweilige Sprache zur Umsetzung der Named Entity Recognition[14] (NER – siehe Schritt 1).
- Setup des a priori Wissens bestehend aus:
  - Technische Taxonomie bzw. Wissensmodell (public domain oder käuflich zu erwerben).
  - Domänenspezifische Wörterbücher (public domain oder käuflich zu erwerben).
  - Partonomien, Designbäume (im Unternehmen vorhanden, können automatisch konvertiert werden).
  - Plausibilisierungsfunktionen bezogen auf das technische Wissensmodell, Partonomien, etc. (werden automatisch u. a. mithilfe der technischen Taxonomien erzeugt).
  - Optional: Stücklisten und Designreferenz (im Unternehmen vorhanden, können automatisch konvertiert werden).
- Semantische N-Gramme, die die Abfolge der Teil-Grammatik für die betrachtete Domäne in der CNL-Ausführung beschreiben (werden aus dem Wissensmodell abgeleitet).

Schritt 1: Grammatikbasierte NER

Jedes einzelne Wort (bzw. Funktionseinheit, wie z. B. Referenz auf eine Darstellung in Klammer oder Teilenummer) bekommt automatisch eine grammatikalische Funktion zugeordnet (grammar-tag) – siehe Spalte 2 der Tabelle 2.

Schritt 2: Auflösen von Referenzidentitäten

Hier erfolgt das einfache Auflösen bzw. »Bereinigen« von Sätzen:

- Stilistische Co-Referenzen (Pronomen) werden automatisch zur konkreten Referenz umbenannt.
- Aus dem a priori Wissen bekannte Wortgruppen mit fester Bedeutung, wie z. B. *»Emergency Escape Path Marking Systems«* werden als feste Begrifflichkeit erfasst, damit diese nicht weiter linguistisch ausdifferenziert werden müssen.

Spätestens auf dieser Stufe werden auch Sätze mit unbekannten Worten ausgesondert, um ggfs. die unbekannten Worte in ein Benutzerwörterbuch aufzunehmen.

13 Wikipedia, (Retrieved August 20, 2019), ↗ https://en.wikipedia.org/wiki/WordNet

14 Wikipedia, (Retrieved August 20, 2019), ↗ https://en.wikipedia.org/wiki/Named-entity_recognition

Schritt 3: Auflösen von Mehrdeutigkeiten

Die Instruktion der für einen bestimmten Arbeitsschritt notwendigen Vorbereitung:

»... in the cockpit, on the overhead panel, make sure that the dial is set to the release position ...«

ist für den Menschen einfach zu verstehen, aus der Linguistik ergibt sich aber nicht, wo sich das Einstellrad (»dial«) genau befindet. Ein Roboter könnte hier auch verstehen: *gehe auf das overhead panel und suche das Einstellrad im Cockpit* – eine Anweisung, die im Zweifel gefährlich ist. Die Auflösung ergibt sich erst aus dem Teile-Modell des Flugzeugs. In ObjectLogic ist die einfachste Form einer Partonomie für diesen Fall:

- Fuselage[hasPart → Cockpit[hasPart → OverheadPanel]]

Schritt 4: Matchen von semantischen N-Grammen des Modells mit den konkreten N-Grammen der Texte

In diesem Satz

- *»Detach the optical cable (3) and adhere it to the chair structure with BONDING AND ADHESIVE COMPOUNDS (Material No: xx-yyy).«*

sind die folgenden N-Gramme zutreffend (in Prädikatsform dargestellt):

- nGram(verb, adjectiv, noun)
- nGram(verb, noun, preposition, noun, noun, preposition, noun).

Nun muss in einem zweiten Schritt die konkrete Bedeutung (die Semantik) der Bezüge ausgewertet werden, insbesondere die Präpositionen »to« und »with«. Hier im Gesamtüberblick:

Satzteil (PoS)	Grammatik	Strukturelle Bedeutung	Fluss	HOL Teil-Modell
Detach	verb	act, implies agent who does it	step 1	i:Instruction[Act, s1:Step[agent → ?Agent, act → detach]]
the	article	specifier of the object	step 1	–
optical	adjective	specifier to act on	step 1	i:Instruction[s1:Step[objectSpecifier → optical]]
cable	noun	target to act on	step 1	i:Instruction[s1:Step[object → cable]]
(3)	parenthesized reference	reference to drawing	step 1	i:Instruction[s1:Step[refDrawing → 3]]
and	conjunction	divides step 1 and step 2	–	–
adhere	verb	act, implies agent who does it	step 2	i:Instruction[s2:Step[agent → ?Agent, act → adhere]]
it	pronoun	co-reference to previous object optical cable	step 2	i:Instruction[s2:Step[objectSpecifier → optical, object → cable]]
to	preposition	direction of the act	step 2	i:Instruction[s2:Step[actDirection → torwards]]
the	article	–	step 2	–
chair	noun	specifier of the 2nd target	step 2	i:Instruction[s2:Step[targetSpecifier → chair]]
structure	Noun	2nd »argument« i. e. target of the verb adhere	step 2	i:Instruction[s2:Step[target → structure]]
with	preposition	reference to the »helper« of verb	step 2	–
BONDING AND ADHESIVE COMPOUNDS	compound noun (capitalized as defined term)	the »helper« which is a consumable as known from the BoM	step 2	i:Instruction[s2:Step[actConsumable → »BONDING AND ADHESIVE COMPOUNDS«]]
(Material No: xx-yyy).	parenthesized reference	Reference to BoM	step 2	i:Instruction[s2:Step[refBoM → xx-yyy]]

Tabelle 3: Transformation eines Satzes in eine HOL-Repräsentation

Eine besondere Teil-Aufgabe kann die Differenzierung von Aktivitäten und Status sein, wenn der Status idiomatisch beschrieben wird. Im Satz: *»... loosen the screws to take the chair off the rails ...«* ist der zweite Teilsatz *»... to take the chair off the rails ...«* eine weitere Aktivität. Im Satz: *»... remove the circuit breaker to cut off the power ...«* ist der zweite Teilsatz *»... cut off the power ...«* der Status der elektrischen Versorgung nach der Aktivität, die im ersten Teilsatz beschrieben wird. Solche idiomatischen Ausnahmen sind Teil des a priori Wissens und werden automatisch vom SemanticMatcher als »post state« berücksichtigt.

Schritt 5: Diskriminieren von Falsch-Positiven und Erkennen von Falsch-Negativen Ergebnissen

Falsch-Positive treten in zwei unterschiedlichen Formen auf:

- Es gibt zu einem Teilsatz der Wartungsanweisung mehr als eine Modellrepräsentation (weil mehr als ein N-Gramm auf den Teilsatz passt). Somit gibt es in dem Teilsatz eine (nicht intendierte, aber linguistisch vorhandene) Mehrdeutigkeit, die vom Modell nicht automatisch aufgelöst werden kann. Für eine korrekte Abbildung der Wartungsanweisung ist ein eindeutiger Bezug erforderlich, was in der Trainingsphase zu berücksichtigen ist.

- Es gibt zwar nur eine Modellrepräsentation, diese macht aber in der Anwendung keinen Sinn. Um dies festzustellen, gibt es zwei Stufen der Plausibilitätsprüfung. Die Erste erfolgt bei der N-Gramm-Auswertung: das Verb *»adhere«* kann zum Beispiel nicht mit der Präposition *»away«* verbunden sein. Die zweite Plausibilitätsprüfung findet auf der Anwendungsebene statt: *»Loosen anti-rattle nuts with a ladder«* würde inhaltlich kein Sinn machen. Dies wird mit dem technischen Wissensmodell des a priori Wissens abgefangen. Daraus ergibt sich, dass man Muttern nur mit bestimmten Werkzeugen bestimmungsgemäß bearbeiten kann. Diese Plausibilitätsprüfungen erfolgen auf Basis des a priori Wissens automatisch.

Textteile, die von den N-Grammen nicht zugeordnet wurden, werden zur Verifikation möglicher Falsch-Negativer genutzt.

Die Berücksichtigung von Falsch-Positiven und Falsch-Negativen in einer initialen Trainingsphase ist essenziell, um die Korrektheit der Standardmuster und die Vollständigkeit des a priori Wissens zu gewährleisten. Aber auch in der Produktionsphase können und müssen Falsch-Positive und Falsch-Negative zur Qualitätssicherung gesammelt werden.

Schritt 6: Serialisierung

In einem an das BPMN-Modell angelehnten Verfahrensfluss müssen die Einzelaktivitäten im abschließenden Schritt serialisiert werden. Ein Abfolge-Ordnungskriterium ist durch die Satzabfolge der Instruktionen und Teilinstruktionen innerhalb eines Satzes gegeben. Gegebenenfalls muss zusätzlich die Semantik der Formatierung analysiert werden, da nicht jeder Satz eine Instruktion beinhaltet, z. B. weil manche Sätze Überschriften sind. Im nachfolgend abgebildeten Layout der Wartungsinstruktionen liegen die Instruktionen immer auf der zu den nächsten Nachbarn untersten Einrückungsebene. In den Überschriftsätzen sind dabei gegebenenfalls Konditionale formuliert, da die nachfolgenden Instruktionen nur dann ausgeführt werden dürfen, wenn die Konditionale erfüllt sind. Wenn Layout-Semantik eine Rolle spielt, werden die Einrückungen bei der Extraktion der Texte den Sätzen als Attribute mitgegeben.

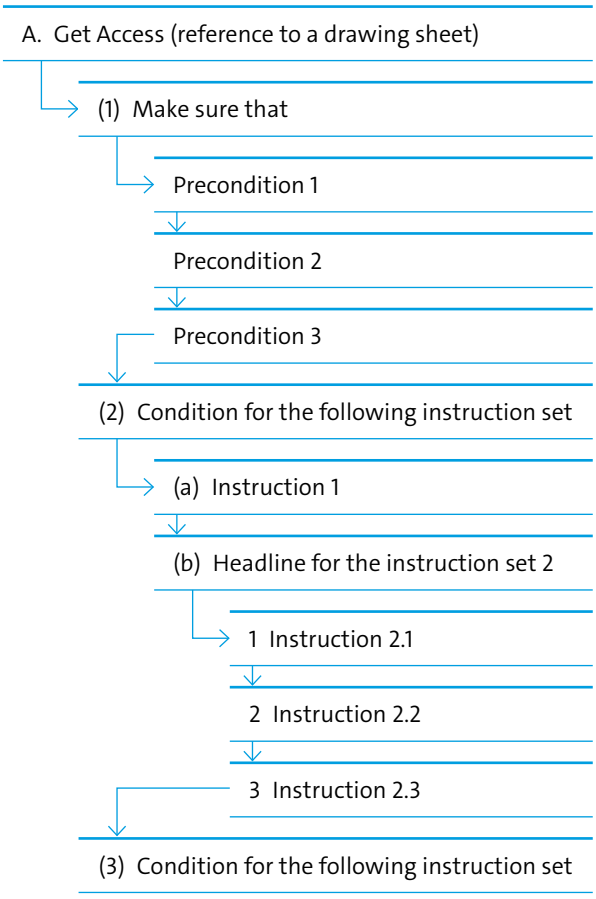


Abbildung 29: Beispiel-Schema einer Instruktions-Layout-Semantik

8.7 Erklär- und Nachvollziehbarkeit

Erklärbarkeit ist existenziell und Teil der Qualitätskontrolle bei der Bestimmung von Falsch-Positiven und Falsch-Negativen. Jeder der einzelnen Wissensextraktionsschritte baut aufeinander auf. D. h., wenn man sich einen bestimmten Schritt in der HOL-Wissensrepräsentation anschauen will, müssen zuvor alle anderen Schritte in einer deterministischen Abfolge abgearbeitet werden. Das bedeutet im Umkehrschluss, dass sich jedes Detail einer Instruktion auf seinen Ursprung im Text aufgrund der logischen Verkettung der Bezüge zurückverfolgen lässt.
Dieser deterministische Bezug zwischen Ursache und Wirkung ist eine der großen Stärken des Verfahrens und der Grund, weswegen sich die semantische KI insbesondere für präzise Erkennung in sensitiven Arbeitskontexten eignet (operativ kritische oder juristische Anwendungsdomänen). Siehe auch Annex 1 für eine beispielhafte Umsetzung.

8.8 Die IT-technische Umsetzung der Wissensextraktion

Die IT-technische Umsetzung der Wissensextraktion kommt aufgrund der Effizienz und Tragfähigkeit der verwendeten KI-Verfahren ohne umfangreiche manuelle Modellierung aus.
Sie greift stattdessen in automatisierter Form auf vorhandene Vorwissensstrukturen (z. B. public domain Wissensnetze) zu und wird auf etablierten Systemen prozessiert. OntoBroker ist ein kommerzieller HOL-Reasoner, der seit 1999 kontinuierlich erweitert und verbessert wurde.
Seit der Version 6.0 wird F-Logic 2 (ObjectLogic) unterstützt und derzeit liegt OntoBroker in der Version 6.3 vor. OntoBroker ist vollständig in Java implementiert und kann logische Auflösungen unter Ausnutzung von Multicore Prozessoren massiv parallel durchführen.

In der verwendeten Ausführung ist der Reasoner OntoBroker eng in Microsoft Excel integriert (MSO365, Excel 2019), wobei der Reasoner sowohl lokal (Windows 10) als auch in beliebiger Cloud-Installation laufen kann (Azure, AWS, nativer Linux-/Windows-Server oder als Docker):

Die Deklarationen der HOL-Modelle sind in Excel Zellen erfasst und dadurch leicht handhabbar (N-Gramm Matcher, a priori Wissen, etc.).

- Textdaten werden von externen Dateien in Excel eingelesen oder schlicht hineinkopiert und dort in für den OntoBroker verarbeitbare ObjectLogic-Repräsentation konvertiert.
- Externe Massendaten wie Wörterbücher (z. B. WordNet) oder technische Taxonomien werden von Excel gesteuert in OntoBroker eingelesen.
- Das transformierte Wissen wird in Excel zur weiteren computerbasierten automatisierten Verwendung ausgegeben.
- Die Verwaltung von fehlerhaften (bzw. nicht vorhandenen) Zuordnungen wird in Excel durchgeführt (Fortsetzung des Trainingsprozesses aus Schritt 5 zur Erweiterung der N-Gramm Sammlung, der benutzerdefinierten Wörterbücher, etc.).

8.9 Zusammenfassung und Ausblick

HOL-Matcher erscheinen als das einfachste und effizienteste Verfahren um präzise Wissensextraktion aus Texten vorzunehmen. Präzision ist bei operativ kritischen bzw. in juristischen Anwendungsdomänen existenziell. Jedes inhaltlich relevante Detail eines Textes muss berücksichtigt werden und man benötigt die Rückverfolgungsmöglichkeit bzw. die Erklärbarkeit der Erkennung zur Verifikation der Ergebnisse.

Der HOL-Matcher zur Erkennung von Textmustern kann auch bei anspruchsvolleren HOL-Modellen eingesetzt werden. Das von uns auf dieser Basis entwickelte Vertragsmodell OntoLegal benötigt eine Anzahl von N-Grammen im oberen zweistelligen Bereich, um die Vielfalt der Ausdrucksformen abzudecken. Zukünftig wird hier ein semantischer Multi-Grid-Ansatz die Anzahl der N-Gramme weiter reduzieren.

8.10 Annex 1: Beispiel (simplifiziert) zur Rückverfolgung von Ergebnissen

Es gibt drei Aussagen A1–A3, in einem Satzprädikat P: P(A1[a, b, c]), P(A2[d, e, f]), P(A3[i, j, k]). Diese Aussagen habe jeweils unterschiedliche Attribute anhand derer die Aussagen gematched werden sollen. Ein Matcher dazu ist m(S1):M[d, e, f] (lies: der Matcher m für die Semantik S1 aus der Klasse der Matcher mit den Matching-Attributen d, e, f), der also immer Aussagen matched, die die Attribute d, e, f haben. Die HOL-Funktion @{f1} T1[?S] :- P(?A[?p1, ?p2, ?p3]), m(?S):M[?p1, ?p2, ?p3] »erzeugt« ein neues F-Atom, nämlich T1[S1], da der Matcher m(S1) einen Match gefunden hat. Falls eine weitere Funktion hinzu kommt, die bestimmte Semantiken modifiziert, z. B. Begriffe mithilfe einer Taxonomie normiert (aus S1 mach S11): @{f2} T2[?SMod] :- T1[?S], SException(?S, ?SMod), hat man zwei hintereinander durchgeführte Transformationen und der Ausgangspunkt ist nur durch die Rückverfolgung der Transformationsschritte möglich. Mit der Funktion TraceFunc() geht diese wie folgt: ?- TraceFunc(T2(S11), ?ID, ?RootBody). Das Ergebnis ist dann: ?ID = [f2, f1], also eine Liste von hintereinander durchlaufende Funktionen (diese kann auch verschachtelt sein bei komplexen Abhängigkeiten) und ?RootBody = [[P(A1[d, e, f]), m(S1):M[d, e, f]]], also der Ausgangspunkt des Matching-Prozesses.

9 Die gesellschaftliche Relevanz von Transparenz bei intelligenten Systemen

9 Die gesellschaftliche Relevanz von Transparenz bei intelligenten Systemen

Frank Wisselink, Nikolai Nölle, Dominik Schneider

9.1 Einleitung

Intelligente Systeme erhalten verstärkt Einzug in die Gesellschaft. Bereits heute setzen viele Alltagsanwendungen auf intelligente Systeme auf – sei es bei der Online-Suche, in Online-Shops oder bei der Nutzung von Chatbots und Sprachassistenten. Intelligente Systeme sind computerbasierte Strukturen, die in der Lage sind »menschenähnliche‹, intelligente Verhaltensweisen zu zeigen« [1]. Wie in der Publikation »Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung« beschrieben, werden »Intelligente Systeme [...] zukünftig in vielen Lebensbereichen Entscheidungen (selbstständig) treffen und damit die Handlungsfähigkeit und Handlungsmächtigkeit jedes Einzelnen beeinflussen« [2]. Damit sind sie potentiell in der Lage, menschliches Handeln zu beeinflussen und gesellschaftliche Strukturen zu verschieben [3]. Aus Sicht des Bitkom wird die »zentrale ethische Herausforderung [...] sein, intelligente Systeme humangerecht und werteorientiert zu gestalten« [4].

9.2 Die gesellschaftliche Relevanz von intelligenten Systemen macht Digitale Ethik erforderlich

Die ethische Herausforderung ist unter dem Begriff »Digitale Ethik« zusammengefasst. Sie beschäftigt sich mit der Frage, wie intelligente Systeme die »menschlichen Fähigkeiten stützt, erweitert und dem Gemeinwohl dient« [5]. Digitale Ethik hat nicht nur einen ideellen Wert. Auch aus wirtschaftlicher Sicht hat Digitale Ethik eine hohe Relevanz [6]. Die Bedeutung von Digitaler Ethik wird zudem von der EU bestätigt. Diese hat sich zum Ziel gesetzt »ethische, sichere und hochmoderne KI-Anwendungen die am Standort Europa entwickelt werden« [7] zu unterstützen. Um dieses Ziel zu erreichen und um den beteiligten Parteien einen Rahmen für ethisches Handeln zu geben, sind mit den Ethik-Leitlinien für eine vertrauenswürdige KI [8] entsprechende Richtlinien aufgestellt worden.

Auch in der Industrie ist die Bedeutung Digitaler Ethik erkannt. Führende Unternehmen haben bereits ethische Richtlinien vorgestellt und intern umgesetzt. Die Deutsche Telekom Gruppe gilt als ein Beispiel. Als eines der ersten Unternehmen hat sie selbstbindende Leitlinien für den Einsatz von KI formuliert [9]. Einige andere Unternehmen wie Google [10] oder SAP [11] haben ebenfalls Leitlinien für den Einsatz von Künstlicher Intelligenz ausgegeben.

Doch ob USA, Europa oder Deutschland: Allgemeingültige und für KI spezifische Rahmenbedingungen bestehen bisher nicht. Wie kann eine Gesellschaft sich also darauf verlassen, dass Künstliche Intelligenz nachhaltig Mehrwert schafft?

9.3 Transparenz ist essentiell um vertrauensvoll Mehrwert für die Gesellschaft zu schaffen

Intelligente Systeme schaffen Mehrwert, indem sie Informationen so aufbereiten, dass diese gezielt eingesetzt werden können [12]. Im ersten Schritt wird durch die Informationsaufbereitung eine verbesserte Kenntnis in Bezug auf eine konkrete Situation geschaffen. Diese verbesserte Kenntnis ermöglicht eine höhere Entscheidungsagilität. Erst wenn diese verbesserte Kenntnis zur Erhöhung der Entscheidungsagilität angewandt wird, entsteht Mehrwert [13]. Dieses Prinzip ist für die Gesellschaft genauso gültig [14] wie für den wirtschaftlichen Einsatz von intelligenten Systemen. Dieser Mehrwert ist jedoch nur dann nachhaltig, wenn das Kundenvertrauen dauerhaft aufrechterhalten wird [15]. Vertrauen ist die positive Erwartung eines Einzelnen in das Handeln und die Intentionen eines anderen [16]. Der Einzelne begibt sich dabei stets in eine Abhängigkeit und einen Zustand potentieller Verletzlichkeit. Handlungen mit positiven Auswirkungen auf den Einzelnen fördern das Vertrauen und die Sicherheit gegenüber dem anderen. Negative, schädliche Handlungen z. B. auf Basis selbstsüchtiger, boshafter Motive mindern hingegen das Vertrauen [17].

In Bezug auf intelligente Systeme bedeutet dies, dass der Anwender positive Handlungsweisen von dem System erwartet. Diese betreffen auf der einen Seite die direkten Ergebnisse wie z. B. konkrete Handlungsempfehlungen. Diese dürfen für den Anwender nicht nachteilig sein. Aber auch der Umgang mit den von dem System bereitgestellten Informationen ist von Bedeutung. Auch hier ist eine Anwendung schädlich, die dem Anwender Nachteile bringen könnte. Ist kein Vertrauen in ein intelligentes System oder dessen Anbieter vorhanden, wird die Anwendung entsprechend abgelehnt. Der Mehrwert bleibt somit aus. Wiederholt sich diese Erfahrung des Anwenders auch bei anderen Systemen, wird diese auf ein gesamtes Unternehmen oder sogar die gesamte Technologie übertragen. Die Folgen wären eine verminderte Bereitschaft zum Teilen von Informationen und zur Nutzung intelligenter Systeme insgesamt.

Abbildung 30: Beziehungen zwischen den sieben Anforderungen der EU [18]

Damit durch intelligente Systeme entsprechend positives Handeln entsteht und somit das Vertrauen geschützt wird, müssen die Systeme ethischen Richtlinien folgen. Aus diesem Grund hat die EU High-Level Expert Group folgende Anforderungen an intelligente Systeme formuliert:

- Vorrang menschlichen Handelns und menschliche Aufsicht
- Technische Robustheit und Sicherheit
- Schutz der Privatsphäre und Datenqualitätsmanagement
- Transparenz
- Vielfalt, Nichtdiskriminierung und Fairness
- Gesellschaftliches und ökologisches Wohlergehen
- Rechenschaftspflicht

In dem vorliegenden Papier liegt der Schwerpunkt auf der Anforderung »Transparenz«. Transparenz lässt einen Einzelnen die Handlungsweisen eines anderen nachvollziehen. Sie kann somit die Erwartung des Einzelnen in Bezug auf die Handlung des anderen leiten. Transparenz ermöglicht dem Einzelnen dadurch, den Grad seiner Verletzlichkeit einzuschätzen und zu entscheiden, in welchem Maße er sich dem anderen gegenüber öffnet [19].

In Bezug auf intelligente Systeme ist der Einzelne der Anwender und der andere das intelligente System bzw. stellvertretend dessen Anbieter. Um Transparenz bzgl. der Handlungen intelligenter Systeme zu schaffen, müssen deren Komponenten sowie Algorithmen nachvollziehbar sein.

9.4 Rückverfolgung, Erklärbarkeit und Kommunikation machen intelligente Systeme transparent

Transparenz von intelligenten Systemen wird laut EU in drei Elemente unterteilt: Rückverfolgbarkeit, Erklärbarkeit und Kommunikation [20]. Transparenz bezieht sich demnach auf die für ein intelligentes System relevanten Komponenten: Die Daten, das System selbst und die damit verfolgten Geschäftsmodelle.

Rückverfolgbarkeit: Zur Rückverfolgbarkeit sollen die von der KI für die Entscheidungsfindung verwendeten Datensätze sowie Prozesse so gut wie möglich [21] dokumentiert werden. Hierzu gehört auch die Erfassung und Kennzeichnung der durch das KI-System verwendeten Daten sowie die eingesetzten Algorithmen. Auf diese Weise sollen die durch die KI hergeleiteten Entscheidungen für den Menschen nachvollziehbar und erklärbar sein.

Erklärbarkeit: Durch KI hergeleitete Entscheidungen sollen für den Menschen nachvollziehbar und erklärbar sein. Die Erklärung soll dabei stets gegenüber der jeweiligen Zielgruppe (z. B. Laien, Regulierungsbehörden oder Forscher) verständlich sein. Sie ist somit an das entsprechende Vorwissen der Interessengruppe anzupassen, was an sich wiederum ein eigenes Verständnis in Bezug auf die Entscheidungsfindung voraussetzt. Durch die Erklärbarkeit soll entsprechend wahrgenommene oder tatsächliche Willkür gegenüber der Interessengruppe ausgeschlossen werden.

Kommunikation: Eine KI soll in jedem Fall gegenüber dem Anwender als solche erkennbar sein. Ein Auftreten oder Erscheinen der KI als Mensch darf nicht erfolgen. KI-Systeme können beispielsweise innerhalb einer Konversation auf ihre Natur hinweisen oder durch einen entsprechenden Namen als solche gekennzeichnet sein. Zudem sollen die Fähigkeiten und Einschränkungen des KI-Systems gegenüber dem Anwender kommuniziert werden. Sollte der Anwender nicht mit einem KI-System interagieren wollen, soll ein entsprechender menschlicher Ansprechpartner vermittelt werden [22].

Ähnliche Anforderungen formuliert die Deutsche Telekom Gruppe in ihren KI-Leitlinien. Auch hier muss sich nach Leitsatz »4. Wir stehen für Transparenz« ein KI-System als solches zu erkennen geben [23]. Zudem soll Transparenz darüber hergestellt werden, wie Kundendaten

genutzt werden. Auch SAP widmet der Transparenz einen Paragraphen in deren KI-Grundsätzen. Demnach werden »zugehörigen Eingaben, Funktionen, der Nutzungszweck und die Grenzen« eines intelligenten Systems an den Anwender kommuniziert. Zudem sollen dem Anwender entsprechende »Überprüfungs- und Kontrollfunktionen« bereitgestellt werden.

Auch die Bundesregierung fordert in ihrer »Strategie Künstliche Intelligenz« die Nachvollziehbarkeit von KI-basierten Entscheidungssystemen ein [24]. Hierzu möchte sie die »Forschung zu Transparenz und Nachvollziehbarkeit von KI-Systemen vorantreiben« [25]. Zudem soll für solche Systeme ein Ordnungsrahmen geschaffen werden, der die Grundrechte der Bürger schützt. In Bezug auf KI-basierte Entscheidungen sind dies insbesondere das Recht auf allgemeine Handlungsfreiheit, auf Schutz der Privatsphäre und das Recht auf informationelle Selbstbestimmung. Diese Linie wird durch die Datenethikkommission weiter unterstrichen [26]. Die Datenethikkommission empfiehlt, dass ethische und rechtliche Grundsätze im gesamten Prozess verankert werden. Dies umfasst sowohl die Entwicklung als auch die Anwendung von KI-Systemen. Somit ergeben sich Handlungsfelder in Forschung und Wissenschaft als auch in der Gesetzgebung.

Weitere Anforderungen in Bezug auf die Transparenz der verwendeten Daten werden durch die Datenschutzgrundverordnung (EU-DSGVO) gestellt. Dort befassen sich Art. 12 (Transparente Information, Kommunikation und Modalitäten für die Ausübung der Rechte der betroffenen Person) und Art. 13 (Informationspflicht bei Erhebung von personenbezogenen Daten bei der betroffenen Person) mit der Thematik.

Damit intelligente Systeme solche ethischen Richtlinien einhalten können, sind Prinzipien zur Gestaltung erforderlich und müssen weiterentwickelt werden [27].

In Bezug auf maschinelles Lernen (ML) ist Erklärbarkeit kein einheitlich definierter Begriff und umfasst in der Praxis eine Vielzahl unterschiedlicher Ansätze. Man unterscheidet diese beispielsweise danach, ob sie als Whitebox-Tests auf spezielle Verfahren zugeschnitten sind oder als Blackbox-Tests auf beliebige Verfahren angewandt werden können [28]. Zudem unterscheidet man zwischen Ansätzen, die versuchen, intelligente Systeme global zu erklären, und solchen, die lediglich versuchen, für einzelne Entscheidungen oder Vorhersagen Erklärungen zu liefern. Lokale Erklärungsverfahren gewinnen dabei immer stärker an Bedeutung, da gerade bei sehr komplexen intelligenten Systemen oft keine globale Erklärbarkeit erzielt werden kann. In den vergangenen Jahren hat sich die Forschung in diesem Bereich intensiviert, was zu einer Vielzahl an neuen Ansätzen geführt hat. Wir sind jedoch noch weit davon entfernt, eine generelle Lösung für das Problem der Erklärbarkeit von intelligenten Systemen zu haben. Die Herausforderung der Erklärbarkeit besteht auch bei komplexen nicht KI-basierten Entscheidungsabläufen.

Dies liegt auch daran, dass Erklärbarkeit und Transparenz keine Eigenschaften sind, die durch rein technische Maßnahmen gewährleistet werden können. Sie setzen vielmehr auch ein Verständnis davon voraus, wie intelligente Systeme in Prozesse integriert werden und welche Ziele mit diesen verfolgt werden. Denn bereits bei der initialen Gestaltung eines intelligenten Systems wird eine Vielzahl an Entscheidungen getroffen, die sich nur implizit im finalen Modell

wiederfinden: Ziele werden definiert, Datensätze ausgewählt und vorbereitet, verschiedene Verfahren mit Daten trainiert und verglichen, Parameter und Modell-Architekturen optimiert und Ergebnisse validiert. In jedem dieser Schritte werden implizite oder explizite Annahmen getroffen, beispielsweise über die Verteilung von Trainingsdaten oder das wünschenswerte Verhalten des intelligenten Systems. Hierbei müssen oft Kompromisse eingegangen werden: Schriftliche Zieldefinitionen können z. B. nicht immer perfekt in mathematische Optimierungsmodelle übersetzt werden, sodass bereits bei der Definition der Optimierungsziele keine vollständige Nachvollziehbarkeit mehr gewährleistet ist. Um Verfahren nachvollziehbar und transparent zu machen, müssen daher auch solche Annahmen dokumentiert und analysiert werden können. In der Praxis ist hierfür immer eine Kombination aus organisatorischen und technischen Maßnahmen nötig.

In dieser Publikation wurden dazu Beispiele für transparente Algorithmen und intelligente Systeme aus der Praxis vorgestellt. In diesem Zusammenhang wurde darauf eingegangen, wie bei den einzelnen Beispielen die Transparenz entlang des gesamten Lebenszyklus berücksichtigt wurde.

9.5 Literaturverzeichnis

[1] Bitkom/DFKI (2017), Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, S. 28.
[2] Bitkom/DFKI (2017), Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, S. 112.
[3] Vgl. Bitkom/DFKI (2017), Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, S. 19.
[4] Bitkom/DFKI (2017), Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, S. 112.
[5] Bitkom/DFKI (2017), Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, S. 19.
[6] Vgl. Wisselink/Schneider/Nölle (2019), The Bottom Line: The Economic Benefits of Digital Ethics.
[7] Europäische Kommission (2018), COM(2018) 795 final – Mitteilung der Kommission an das Europäische Parlament, den Europäischen Rat, den Rat, den Europäischen Wirtschafts- und Sozialausschuss und den Ausschuss der Regionen: Koordinierter Plan für künstliche Intelligenz, S. 2.
[8] High-Level Expert Group on Artificial Intelligence (2019), Ethics Guidelines for trustworthy AI.
[9] Vgl. Deutsche Telekom AG (2018), Leitlinien für Künstliche Intelligenz.
[10] Vgl. Pichai (2018), AI at Google: our principles.
[11] Vgl. SAP (2018), Die Grundsätze für Künstliche Intelligenz von SAP.
[12] Vgl. Wisselink et al. (2015), The Value of Big Data for a Telco, S. 155 ff.
[13] Vgl. Bitkom/DFKI (2017), Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, S. 68.
[14] Vgl. Wisselink/Schneider/Nölle (2019), The Bottom Line: The Economic Benefits of Digital Ethics.

[15] Vgl. Wisselink/Schneider (2019, The Artificial Intelligence Challenge: How Telcos Can Obtain a Grand Prix for Insights Monetization, S. 336.
[16] Vgl. Möllering (2001), The Nature of Trust: From Georg Simmel to a Theory of Expectation, Interpretation and Suspension, S. 404.
[17] Vgl. Manstead/Hewstone (2004), The Blackwell encyclopedia of social psychology. Blackwell Publishing, S. 656 f.
[18] High-Level Expert Group on Artificial Intelligence (2019), Ethics Guidelines for trustworthy AI, S. 15.
[19] Vgl. High-Level Expert Group on Artificial Intelligence (2019), Ethics Guidelines for trustworthy AI, S. 18 f.
[20] Vgl. High-Level Expert Group on Artificial Intelligence (2019), Ethics Guidelines for trustworthy AI, S. 18 f.
[21] Vgl. High-Level Expert Group on Artificial Intelligence (2019), Ethics Guidelines for trustworthy AI, S. 18.
[22] Vgl. SAP (2018), Die Grundsätze für Künstliche Intelligenz von SAP.
[23] Vgl. Deutsche Telekom AG (2018), Leitlinien für Künstliche Intelligenz.
[24] Vgl. Die Bundesregierung (2018), Strategie Künstliche Intelligenz der Bundesregierung, S. 10.
[25] Die Bundesregierung (2018), Strategie Künstliche Intelligenz der Bundesregierung, S. 16.
[26] Vgl. Datenethikkommission (2018), Empfehlungen der Datenethikkommission für die Strategie Künstliche Intelligenz der Bundesregierung.
[27] Vgl. Bitkom/DFKI (2017), Künstliche Intelligenz – Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, S. 19.
[28] Vgl. Dewes/Jarmul (2018), Algoneer: Toolkit for Analyzing and Breaking AI-Systems.

10 Zertifizierung und Attestierung von KI Systemen: Schwerpunkt Nachvollziehbarkeit und Transparenz

10 Zertifizierung und Attestierung von KI Systemen: Schwerpunkt Nachvollziehbarkeit und Transparenz

Kentaro Ellert , Hendrik Reese

10.1 Was versteht man unter Nachvollziehbarkeit von KI und warum wird diese benötigt?

Eine häufig im Zusammenhang mit Künstlicher Intelligenz (KI) diskutierte Thematik ist die unzureichende Transparenz aufgrund fehlender Nachvollziehbarkeit einiger KI Methoden. Vor allem im Zusammenhang mit Neuronalen Netzwerken und Support Vektor Maschinen wird oftmals die Blackbox Problematik und das dadurch fehlende Verständnis des Entscheidungsprozesses von KI Systemen diskutiert. Durch den zunehmenden Einsatz von KI Systemen in kritischen Anwendungsbereichen fordern Stakeholder vermehrt Sicherheit und Transparenz. Wenn Fehler und ungewolltes Verhalten aufgedeckt und vermieden werden sollen, müssen Anbieter und Nutzer von KI Systemen Maßnahmen ergreifen, die den Entscheidungsweg von KI Systemen nachvollziehbar darlegen. Ein Beispiel dafür sind Expertensysteme, die Ärzten bei Diagnoseverfahren von schwerwiegenden Krankheiten und damit verbundenen Entscheidungen in Bezug auf Patienten unterstützen sollen. Der Forderung nach höher Nachvollziehbarkeit von KI Systemen stellt sich die Frage gegenüber, ob Nachvollziehbarkeit immer notwendig und unter Berücksichtigung technischer sowie wirtschaftlicher Restriktionen möglich ist.

10.2 Müssen alle KI Systeme nachvollziehbar sein?

Da manche KI Systeme (die der Blackbox Problematik unterliegen) bessere Ergebnisse erzielen als nachvollziehbarere Methoden, muss eine Balance zwischen der Genauigkeit der Ergebnisse und ihrer Nachvollziehbarkeit gefunden werden. [1] Für weniger kritische Anwendungsfälle, wie beispielsweise die Erkennung von Körperbewegungen bei Computerspielen, kann der Nutzen des Anwenders durch bessere Ergebnisse des KI Systems deutlich höher ausfallen, als durch erhöhte Nachvollziehbarkeit. Daher ist es sinnvoll eine anwendungsfallspezifische Bewertung zur Notwendigkeit der Nachvollziehbarkeit durchzuführen.

10.3 Welche Rolle spielt Ethik im Zusammenhang mit Nachvollziehbarkeit?

Aus ethischer Sicht ist die Nachvollziehbarkeit eines Entscheidungsprozesses von großer Bedeutung, da sie ein wichtiger Aspekt fur die Verantwortungszuschreibung darstellt. Des Weiteren wird durch Transparenz der Handlungen intelligenter Systeme das Vertrauen in diese Systeme erhöht (siehe Kapitel 1.1). Wenn die Prozesse in einem KI System nachvollziehbar sind, d. h. Entscheidungen erklärbar sind, können diese leichter gerechtfertigt und auf diese Weise Verantwortlichkeiten bestimmt werden. Somit besteht ein ethischer Wert darin, ein KI System möglichst nachvollziehbar zu gestalten. Unter Berücksichtigung der genannten Balance zwischen Genauigkeit und Nachvollziehbarkeit ist es geboten, einen bestimmten Grad an Nachvollziehbarkeit einzuführen, falls das Resultat einen sehr hohen Nutzen darstellt. Verschiedene Grade der Nachvollziehbarkeit können beispielsweise durch Erklärungen des Ergebnisfindungsprozesses oder einer Input-Output-Validierung umgesetzt werden. (siehe unten: »Welche Arten von Nachvollziehbarkeit sind zu berücksichtigen?«).

10.4 Wie kann ein ethisches Rahmenwerk bei Nachvollziehbarkeit helfen?

Es besteht die Möglichkeit, mit Hilfe eines Ethik-Rahmenwerk eine risikoethische Analyse durchzuführen und so die Nachvollziehbarkeit eines KI Systems in Bezug zu der oben beschriebenen Abwägungen (zwischen Genauigkeit der Ergebnisse und der Nachvollziehbarkeit des Systems) zu setzen. Ein Ethik-Rahmenwerk muss mit gewissen Prinzipien wie Autonomie, Nachvollziehbarkeit, Fairness, Verantwortung und Sicherheit ausgestattet sein, die sich wiederum in konkretere Unterprinzipien (z. B. autonom, teilautomatisiert etc.) aufgliedern. Auf diese Weise können abstrakte ethische Prinzipien konkret auf bestimmte Anwendungsfälle angewendet werden. Ein speziell auf KI Systeme zugeschnittenes Ethik-Rahmenwerk sollte Orientierung in der Wahl der Unterprinzipien geben. Die Abwägungen zwischen der Genauigkeit der Ergebnisse und der Nachvollziehbarkeit des Systems sowie der damit einhergehender Verzicht auf Nachvollziehbarkeit sollte ethisch angemessen sein.

10.5 Warum brauchen wir Zertifikate für KI Systeme und in welchem Umfang sollte eine Zertifizierung durchgeführt werden?

Für 91% aller Entscheider in deutschen Unternehmen ist (laut einer Studie von PwC) die Implementierung von Sicherheit und Transparenz in KI Lösungen von hoher Bedeutung. Dies ist notwendig, um das Vertrauen in die Technologie zu erhöhen. [3]

Durch eine Zertifizierung von KI Systemen kann der sichere und ordnungsgemäße Einsatz von KI im Unternehmensumfeld geprüft und nachgewiesen werden. Außerdem wird Transparenz über die Einhaltung relevanter Prinzipien (wie beispielsweise Hilfsmittel zur Bewertung des Outputs durch ein Konfidenzmaß), den Grad der Nachvollziehbarkeit, Reproduzierbarkeit und der Autonomie des Systems geschaffen. So können Kunden über die Eignung eines KI Systems hinsichtlich ihrer individuellen Bedürfnisse entscheiden. Um eine umfängliche Prüfung von KI Systemen durchführen zu können, sind neben herkömmlichen IT- und Cloud- spezifischen Aspekten (wie beispielsweise organisatorische und personelle Anforderungen), Anforderungen an die logische und physische KI Infrastruktur zu stellen. Darüber hinaus sind vor allem KI spezifische Aspekte zu prüfen, um für den Einsatz von KI Systemen relevante Risiken zu mitigieren. KI spezifische Anforderungen betreffen insbesondere die Generierung, Auswahl und Aufbereitung von (Trainings-) Daten, sowie einen angemessenen Validierungsmechanismen. Weitere Anforderungen betreffen die Robustheit von KI Systemen zur Vermeidung von Angriffen. Beispiele stellen hier Adversarial Attacks oder Angriffe auf die verwendeten Daten oder das zugrundeliegende Modell (Model Theft) dar. Dies impliziert, dass Nachvollziehbarkeit sehr stark auf der Ebene des Betriebs- und Kontrollmodells umgesetzt wird. Das interne Kontrollsystem einer Organisation in Kombination mit technischen Überwachungsmaßnahmen bildet den Rahmen zur Operationalisierung von nachvollziehbarer KI. Zur Prüfung von KI Systemen hat PwC deshalb den Trust in AI Anforderungskatalog entwickelt, der diese Betrachtungsweise und Validierungsmaßnahmen abdeckt.

10.6 Welche Arten von Nachvollziehbarkeit sind zu berücksichtigen?

Der Begriff der Nachvollziehbarkeit lässt ein breites Spektrum an Möglichkeiten über die Art der Umsetzung bei der Verwendung von KI Systemen zu. Um den Grad an Nachvollziehbarkeit zu erhöhen, gibt es verschiedene Ansätze zur Bewertung der Ergebnisse von KI-Systemen durch den Nutzer. Hierbei lassen sich folgende Ausprägungen unterscheiden:

a) Erklärung des Ergebnisfindungsprozesses:

Bei dieser Form der Nachvollziehbarkeit liefert das KI System eine Erklärung für den Prozess der Ergebnisermittlung. Dies ist jedoch insbesondere bei komplexen Verfahren, wie beispielsweise Neuronalen Netzen oder Support Vektor Machines, ein schwieriges Unterfangen.

Mit zusätzlichen sogenannten »Erklärmodellen« ist eine Nachvollziehbarkeit dennoch teilweise möglich. Erklärmodelle können zum Beispiel mit Hilfe von Local Interpretable Model-agnostic Explanations (LIME) umgesetzt werden. Dabei wird für einen Input eine Erklärung zur Ergebnisfindung geliefert. Im Falle einer Bilderkennung kann das die Kennzeichnung der Bereiche sein, die für die Ergebnisermittlung ausschlaggebend sind (siehe Abbildung 31). Dabei wird das Originalbild – ein Gitarre spielender Hund (Bild links) – in jenen Bereichen dargestellt, die auf eine E-Gitarre, eine akustische Gitarre und einen Hund hinweisen.

Abbildung 31: Ergebnisfindungsprozess mit Hilfe von LIME [4]

b) Input-Output-Validierung:

Eine weitere Möglichkeit um KI Systeme nachvollziehbarer zu gestalten, ist die Bewertung des Inputs. Es wird untersucht, welche Teile des Inputs für die Generierung eines bestimmten Outputs von besonderer Bedeutung sind. Dabei können Methoden wie SHapley Additive exPlanations (SHAP) zur Bewertung der einzelnen Daten verwendet werden (siehe Abbildung 32). [5] Dabei werden Daten hinsichtlich ihrer positiven und negativen Auswirkung bei unterschiedlichen Ausprägungen gegenübergestellt.

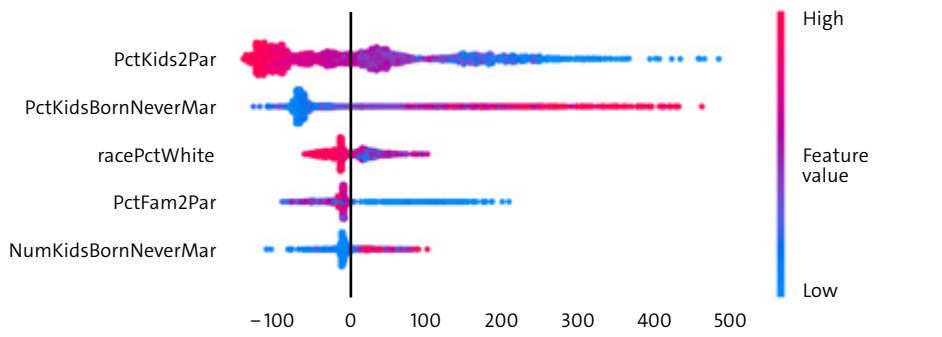


Abbildung 32: SHAP Value am Beispiel eines KI Systems zur Bewertung von Diabetes (Quelle: Eigene Darstellung)

10.7 Wie können KI-Systeme zertifiziert werden?

Bei einer Zertifizierung der Nachvollziehbarkeit von KI-Systemen sollte geprüft werden, ob der Betreiber des KI-Systems (das Unternehmen, bei dem eine Prüfung hauptsächlich erfolgt) eine angemessene Bewertung des notwendigen Grads an Nachvollziehbarkeit für den jeweiligen Anwendungsfall (das zu zertifizierende KI-System) durchgeführt hat und ob daraus geeignete Maßnahmen abgeleitet wurden. Diese Festlegung muss durch qualifiziertes Personal des zu prüfenden Unternehmens durchgeführt werden und sollte neben der Kritikalität (z. B. Lernfrequenz, Grad der Autonomie) den gesetzlichen und ethischen Anforderungen (z. B. Auswirkungen auf Individuen und Gruppen) des zu prüfenden Unternehmens gerecht werden. Aus den Bewertungsergebnissen sind Maßnahmen für die ordnungsgemäße Implementierung und Nutzung von KI-Systemen hinsichtlich ihrer Nachvollziehbarkeit abzuleiten. Die gewünschte Form der Nachvollziehbarkeit muss so vorliegen, dass diese von Verantwortlichen, sowie prüfenden Dritten, mit ausreichenden Kenntnissen über die Nutzung des KI-Systems (jedoch ohne tiefergehende technische Fachkompetenz) verstanden werden kann. Des Weiteren ist zu prüfen, ob eine regelmäßige Überwachung der Einhaltung des notwendigen Grads an Nachvollziehbarkeit durchgeführt wird. Außerdem muss der Grad der Nachvollziehbarkeit offengelegt werden, so dass potenzielle Kunden über die Angemessenheit hinsichtlich ihrer Interessen entscheiden können.

Bei der Bewertung eines internen Kontrollsystems zur Überwachung von KI müssen die Verantwortlichkeiten klar definiert werden, um keine blinden Flecken zuzulassen. Das bedeutet, dass neben Kontrollen auf der Seite des Unternehmens (sog. Entity Level Controls – ELC), welches das KI-System als Produkt anbietet, auch auf der Seite des Anwenders geeignete Kontrollen implementiert werden müssen (sog. Complementary User Entity Controls – CUEC). Eine angemessene Abstimmung zwischen ELCs und CUECs stellt eine volle Abdeckung aller Aspekte sicher, die für den Betrieb und die Überwachung von KI-Systemen notwendig sind.

Dieses Prüfvorgehen zur Nachvollziehbarkeit sowie weitere Anforderungen für einen verantwortungsvollen und transparenten Einsatz von KI-Systemen (wie bspw. die oben angesprochenen Anforderungen an Robustheit) sind von PwC in dem »Trust in AI«-Prüfkatalog ausgearbeitet. In diesem werden sowohl KI-spezifische wie auch herkömmliche Anforderungen für IT-Systeme vereint, um eine umfängliche Zertifizierung von KI-Systemen zu ermöglichen.

10.8 Welche technischen Hilfsmittel können für eine umfängliche Zertifizierung relevant sein?

Um KI-Systeme möglichst umfänglich zu prüfen, sollten je nach Schutzbedarf des KI-Systems entsprechende Tools und technische Hilfsmittel zur Bewertung und Verbesserung des Systems verwendet werden. Dadurch kann die Sicherheit einer Zertifizierung gegenüber einer Prüfungshandlung ohne technische Hilfsmittel erhöht werden. Technische Hilfsmittel können unter anderem zur Bewertung der Robustheit durch die Generierung von Adversarial Attacks

erfolgen. Dabei kann durch gezielte Attacken die Robustheit zum einen geprüft und zum anderen durch zusätzliches Training erhöht werden. Des Weiteren können Trainingsdaten von KI-Systemen auf Bias (z.B. die Benachteiligung einer bestimmten Gruppe von Menschen) untersucht werden. Ein dritter Bereich betrifft Data Poisoning, bei dem (insbesondere für den Fall, wenn Kundenfeedback zum Training des Systems verwendet wird) sichergestellt werden muss, dass die Trainingsdaten nicht maliziös beeinflusst werden können. Für die beschriebenen Fälle hinsichtlich der Robustheit, Bias und Data Poisoning entwickelt PwC Tools zur Bewertung und Verbesserung von KI-Systemen.

10.9 Wie können wir Nachvollziehbarkeit und Transparenz erreichen?

Ein wichtiger Faktor, für den produktiven Einsatz von Künstlicher Intelligenz im Unternehmensumfeld, ist fehlendes Vertrauen in die Technologie. Um das volle Potenzial und die Vorteile von KI zu entfalten, brauchen wir daher sowohl nationale als auch internationale Standards, welche eine Zertifizierung von KI-Systemen ermöglichen. Durch unabhängige Bewertungsverfahren kann sowohl in B2B- als auch B2C-Interaktionen ein gemeinsames Verständnis und ein geeignetes Maß an Sicherheit und Nachvollziehbarkeit in KI etabliert werden.

10.10 Literaturverzeichnis

[1] Gleicher, M. (2016). A Framework for Considering Comprehensibility in Modeling. Big Data, 75–88.

[2] Gandhi, P. (2019). Von Explainable Artificial Intelligence: ↗ https://www.kdnuggets.com/2019/01/explainable-ai.html

[3] High-Level Expert Group on AI. (2019). Ethics guidelines for trustworthy AI. Brüssel: Europäische Kommission.

[4] M. T. Ribeiro, S. Singh, C. Guestrin, »Why Should I Trust You?« Explaining the Predictions of Any Classifier.

[5] Molnar, C. (2019). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable.

Bitkom vertritt mehr als 2.700 Unternehmen der digitalen Wirtschaft, davon gut 1.900 Direktmitglieder. Sie erzielen allein mit IT- und Telekommunikationsleistungen jährlich Umsätze von 190 Milliarden Euro, darunter Exporte in Höhe von 50 Milliarden Euro. Die Bitkom-Mitglieder beschäftigen in Deutschland mehr als 2 Millionen Mitarbeiterinnen und Mitarbeiter. Zu den Mitgliedern zählen mehr als 1.000 Mittelständler, über 500 Startups und nahezu alle Global Player. Sie bieten Software, IT-Services, Telekommunikations- oder Internetdienste an, stellen Geräte und Bauteile her, sind im Bereich der digitalen Medien tätig oder in anderer Weise Teil der digitalen Wirtschaft. 80 Prozent der Unternehmen haben ihren Hauptsitz in Deutschland, jeweils 8 Prozent kommen aus Europa und den USA, 4 Prozent aus anderen Regionen. Bitkom fördert und treibt die digitale Transformation der deutschen Wirtschaft und setzt sich für eine breite gesellschaftliche Teilhabe an den digitalen Entwicklungen ein. Ziel ist es, Deutschland zu einem weltweit führenden Digitalstandort zu machen.

Bundesverband Informationswirtschaft,
Telekommunikation und neue Medien e.V.

Albrechtstraße 10
10117 Berlin
T 030 27576-0
F 030 27576-400
bitkom@bitkom.org
www.bitkom.org

bitkom