Wo Maschinen irren können

Verantwortlichkeiten und Fehlerquellen in Prozessen algorithmischer Entscheidungsfindung

BertelsmannStiftung

Wo Maschinen irren können

Fehlerquellen und Verantwortlichkeiten in Prozessen algorithmischer Entscheidungsfindung

Arbeitspapier

Prof. Dr. Katharina A. Zweig, TU Kaiserslautern
im Auftrag der Bertelsmann Stiftung

unter Mitwirkung von Dr. Sarah Fischer und Konrad Lischka,
Bertelsmann Stiftung

Impressum
© Februar 2018 Bertelsmann Stiftung
Bertelsmann Stiftung
Carl-Bertelsmann-Straße 256
33311 Gütersloh
www.bertelsmann-stiftung.de

Verantwortlich
Konrad Lischka
Ralph Müller-Eiselt

Autorin
Prof. Dr. Katharina A. Zweig, TU Kaiserslautern unter Mitwirkung von Dr. Sarah Fischer und Konrad Lischka, Bertelsmann Stiftung

DOI 10.11586/2018006 https://doi.org/10.11586/2018006

Inhalt

1 Vorwort

Im Sommer 2016 startet die australische Regierung ein Algorithmenexperiment in großem Stil: Eine neu eingeführte Software soll abschätzen, ob Bürger[1] zu viel Sozialleistungen bezogen haben. Das Programm gleicht automatisiert Daten zu Leistungsbezügen mit Steuererklärungen ab.

Zum Start hebt der verantwortliche Minister Alan Tudge Leistungsgewinne durch automatisierte Entscheidungen hervor. Man könne nun viel mehr Mahnungen an Menschen verschicken, die zu hohe Leistungen bezogen hätten: „Zuvor haben wir für 20.000 Interventionen ein Jahr gebraucht, nun schaffen wir 20.000 in der Woche“ (Cosier 2017).

Das ist ein typischer Effekt algorithmischer Entscheidungsfindung: Die Logik eines Systems ist mit vergleichsweise geringem Mehraufwand auf nahezu beliebig viele Fälle anwendbar. Das führt dazu, dass einzelne Prognosen günstiger werden und Menschen häufiger beurteilt werden. Diese Skalierbarkeit hat auch potenzielle Nachteile:

- Fehler werden häufiger. Fallen 52-mal so viele Entscheidungen, wird sich auch die absolute Menge von Fehlurteilen vervielfachen – wenn die Fehlerquote gleichbleiben sollte.
- Fehler haben größere Folgen. Denn dieselbe Entscheidungslogik wird nun in jedem Einzelfall konsistent angewendet. Sie skaliert also auch, wenn sie fehlerhaft sein sollte.

Diese Effekte schadeten in Australien einer Vielzahl von maschinell bewerteten Menschen. Was genau bei der algorithmischen Überprüfung in Australien schieflief, ist bis heute nicht abschließend aufgeklärt. Bekannt ist: Zum Jahreswechsel 2016/2017 berichteten mehrere australische Medien über angeblich ungerechtfertigte Mahnschreiben. Auf einmal hatten viele Menschen Schulden bei der Regierung, weil eine Software das so ausgerechnet hatte. Australische Medien nutzten dafür schon bald einen neuen Begriff: „Robo-Debt“ („Roboterschulden“). Laut Recherchen des TV-Senders ABC hat die Regierung in den ersten Monaten des Einsatzes 200.000 Schreiben wegen Widersprüchen zwischen Steuererklärungen und bezogener Sozialleistungen verschickt. Bei etwa 80 Prozent dieser Fälle lautete das Ergebnis der algorithmischen Entscheidungsfindung: Menschen schulden dem Staat Geld (Rohde 2017).

Australische Medien berichten von Einzelfällen wie diesen: Eine 76-jährige Ethnographin sollte 7600 australische Dollar Rente zurückzahlen. Sie arbeitet ehrenamtlich weiter an ihrer alten Universität und vermutlich hat das System Forschungsgelder als Einkommen angerechnet (Knaus 2017). Ein pensionierter Grundschullehrer sollte 4500 Dollar zurückzahlen. Nach einer mehrmonatigen Überprüfung korrigieren menschliche Sachbearbeiter den Betrag auf 63,17 Dollar. Warum und wie es zu der Fehlberechnung gekommen ist, weiß niemand so genau. Fest steht nur, dass der Lehrer kurz vor der Rente ein untypisches Erwerbsleben hatte. Er leidet an Depression, hörte vor der Pensionierung auf zu unterrichten, arbeitete geringfügig beschäftigt als Platzwart, bezog vor seiner Pensionierung wegen des niedrigen Gehalts anteilig Sozialleistungen.

Es ist nicht überraschend, dass ein algorithmisches System bei ungewöhnlichen Fällen problematische Ergebnisse liefert. Bei ungewöhnlichen Fällen fehlt der Software oft die Flexibilität, auf relevante, aber unerwartete Details adäquat zu reagieren. Das ist ein Nachteil algorithmischer Systeme, die eine vorgegebene Entscheidungslogik konsistent in jedem Einzelfall abarbeiten. Demgegenüber steht der Vorteil, dass sie genau das viel zuverlässiger tun als Menschen. Im Gegensatz zu menschlichen Entscheidern ist Software nicht tagesformabhängig und wendet nicht willkürlich in Einzelfällen neue, unter Umständen ungeeignete Kriterien an. Aber wenn der Einzelfall von typischen Mustern abweicht, kann die algorithmische Konsistenz zum Nachteil werden. Das ist gerade im Sozialsystem problematisch, wo bei ungewöhnlichen Einzelfällen oft Unterstützung am nötigsten gebraucht wird.

[1] Aus Gründen der Einfachheit und besseren Lesbarkeit verwendet diese Publikation vorwiegend die männliche Sprachform. Es sind jedoch jeweils beide Geschlechter gemeint.

Wie viele Fehleinschätzungen das australische Robo-Debt-System traf, ist unbekannt. Es gibt keine öffentlich zugänglichen, systematischen Tests der Fehlerquoten. Es gibt keine Informationen über die genutzte Software. Es gibt keinen wissenschaftlich unabhängigen Vergleich der Entscheidungsqualität des neuen maschinellen und des alten auf menschlicher Einschätzung basierenden Verfahrens. Und die Diskussion darüber, ob ein solches System automatisiert Zahlungsaufforderungen verschicken sollte, wurde erst geführt, nachdem schon Zehntausende solcher Mahnbriefe versendet waren.

Robo-Debt ist ein Paradebeispiel für den missglückten Einsatz algorithmischer Systeme in gesellschaftlich relevanten Zusammenhängen. Das gilt unabhängig davon, wie fehleranfällig das neue Verfahren im Vergleich zum alten tatsächlich ist: Wenn maschinelles Entscheiden mit derart weitreichenden Folgen ohne gesellschaftliche Debatte, ohne unabhängige Qualitätsprüfung ex ante und ex post eingesetzt wird, verlieren die Bewerteten das Vertrauen in das Entscheidungssystem.

Das vorliegende Arbeitspapier ist ein erster Diskussionsvorschlag, wie man es besser machen kann. Katharina A. Zweig skizziert anschaulich die verschiedenen Phasen der Entwicklung und des Einsatzes solcher algorithmischer Systeme. Sie zeigt nicht nur auf, was dabei alles schiefgehen kann, sondern beschreibt auch mögliche Maßnahmen und Instrumente, mit denen solche Fehler aufgedeckt und behoben werden können. Dieser lösungsorientierte Ansatz soll auch dabei helfen, die teilweise aufgeregt geführte Diskussion über den Einsatz von Algorithmen und künstlicher Intelligenz zu versachlichen. Denn eine Schwarz-Weiß-Debatte über Heil und Unheil dieser Technologien wird uns nicht weiterbringen.

Wir veröffentlichen die vorgeschlagene Systematisierung als Arbeitspapier, um einen Beitrag zu einem sich schnell entwickelnden Feld zu geben, auf dem auch andere aufbauen können, und freuen uns über Erweiterungen, Verbesserungen, weiterführende Analysen von Fallbeispielen und natürlich auch konstruktive Kritik. Um einen solchen Diskurs zu erleichtern, veröffentlichen wir das Arbeitspapier unter einer freien Lizenz (CC BY-SA 3.0 DE).

Die Analyse von Katharina Zweig ist Teil des Projekts „Ethik der Algorithmen“, in dem sich die Bertelsmann Stiftung näher mit den gesellschaftlichen Auswirkungen algorithmischer Entscheidungssysteme beschäftigt. Bislang sind eine Sammlung internationaler Fallbeispiele (Lischka und Klingel 2017), eine Untersuchung des Wirkungspotenzials algorithmischer Entscheidungsfindung auf Teilhabe erschienen (Vieth und Wagner 2017) und eine Analyse des Einflusses algorithmischer Prozesse auf den gesellschaftlichen Diskurs (Lischka und Stöcker 2017) erschienen. Das vorliegende Arbeitspapier fokussiert auf die Fehlerquellen in Prozessen algorithmischer Entscheidungsfindung und zeigt beispielhaft erste Lösungsansätze auf. Darauf aufbauend erscheint im Frühjahr 2018 ein weiteres Papier, das eine Vielzahl an Lösungen systematisiert und umfassender betrachtet.

Ralph Müller-Eiselt
Senior Expert
Taskforce Digitalisierung
Bertelsmann Stiftung

Konrad Lischka
Project Manager
Projekt Ethik der Algorithmen
Bertelsmann Stiftung

2 Zusammenfassung

Dieses Arbeitspapier erklärt, wie algorithmische Entscheidungssysteme entwickelt und in einen gesellschaftlichen Kontext eingebettet werden, und zeigt dabei potenziell auftauchende Fehlerquellen auf.
Nach einem einleitenden Teil, in dem die Begriffe Algorithmus und Entscheidungssystem näher definiert werden, beschreibt das Papier den Entwicklungs- und Einbettungsprozess von algorithmischen Entscheidungssystemen. Er besteht aus fünf Phasen: In der ersten Phase werden Algorithmen designt und in Software implementiert. In der zweiten Phase folgt als optionaler Schritt die Operationalisierung, in der Konstrukte (z. B. Relevanz einer Nachricht) in Indikatoren überführt (z. B. Häufigkeit des Anklickens) und dadurch messbar gemacht werden. Außerdem erfolgt die Auswahl der Daten, mit denen der Algorithmus trainiert werden soll, sowie einer Bewertungs- oder Vorhersagemethode. In der dritten Phase werden eine Methode des maschinellen Lernens mit den Trainingsdaten zusammengebracht und das Entscheidungssystem konstruiert. In der vierten Phase erfolgt die Einbettung in die gesellschaftliche Praxis: Das System wird auf neue Daten angewendet, die Ergebnisse werden interpretiert und anschließend in einer Handlung umgesetzt. In der fünften Phase findet eine Evaluation des Entscheidungssystems statt.
Der Prozess, den ein Entscheidungssystem von seiner Entwicklung bis zur Evaluation durchläuft, ist lang und an vielen Stellen von zahlreichen Entscheidungen abhängig. Dafür sind in den einzelnen Phasen unterschiedliche Personen verantwortlich: Wissenschaftler und Programmierer genauso wie Data Scientists und unterschiedliche Akteure (z. B. staatliche, wirtschaftliche, wissenschaftliche Institutionen oder Nichtregierungsorganisationen). Die beteiligten Akteure verfügen jedoch häufig nicht über die eigentlich für ihre verantwortungsvolle Aufgabe notwendigen Kompetenzen: sei es, weil die Programmiererausbildung die sozialen Konsequenzen ihrer Arbeit nicht hinreichend reflektiert oder weil ihnen als Anwender das Wissen fehlt, um Ergebnisse richtig zu interpretieren.
Die hohe Anzahl an Entscheidungen sowie unterschiedlicher Beteiligter macht den Prozess anfällig für Fehler, die in allen Phasen auftauchen können. Sie unterscheiden sich in ihrer Tragweite sowie in ihrer Auffindbarkeit und Vermeidbarkeit. So können etwa Fehler in der ersten Phase des Algorithmendesigns dazu führen, dass ein Algorithmus nicht immer das korrekte Ergebnis berechnet. Solche Fehler sind jedoch für Informatiker relativ leicht zu entdecken und zu beheben, wenn klar ist, was der Algorithmus leisten soll und der Quellcode zugänglich ist.
Operationalisierungsfehler in der zweiten Phase führen zu Ergebnissen, die nicht sinnvoll interpretierbar sind. Es ist jedoch häufig schwierig, sie aufzudecken und zu vermeiden. Mängel in der Datenauswahl können hingegen entdeckt werden, wenn Qualitätsmaße für die Daten bekannt sind.
In der dritten Phase ist es möglich, dass bei der Konstruktion des Entscheidungssystems Unstimmigkeiten auftauchen, wenn Daten und Algorithmus nicht zu einander passen. Dies führt ebenfalls zu unbrauchbaren Ergebnissen.
In der vierten Phase, der Einbettung in den gesellschaftlichen Kontext, entstehen Fehler, wenn Anwender mangelhafte Daten nicht erkennen und die Resultate falsch interpretieren. In dieser Phase ist der Prozess besonders fehleranfällig, weil dort Anwender mit dem System interagieren. Dies kann nicht intendierte Wirkungen hervorrufen.
Fehler der fünften Phase entstehen durch fehlendes oder falsches Feedback. Generell machen die meisten algorithmischen Entscheidungssysteme Fehler, weil es keine hundertprozentigen Entscheidungsregeln gibt, die zu einer perfekten Einordnung aller Daten führen würden. Daneben können etwa veraltete oder falsche Daten zu Fehlprognosen führen. Diese können auch entstehen, wenn sich die soziale Situation geändert hat, sodass die einmal gefundenen Entscheidungsregeln nicht mehr optimal sind. Alle Arten von Fehlentscheidungen bleiben bestehen, wenn das System darüber keine Rückmeldung erhält. Manche dieser Fehlprognosen können mit einigem Aufwand durch Anpassung der Entscheidungsregeln vermieden werden. Diese evolutionäre Weiterentwicklung ist aber dann nicht möglich, wenn das System nur einseitiges Feedback erhält (z. B. kein Feedback darüber, dass derjenige, der keinen Kredit gewährt bekommen hat, ihn eigentlich hätte zurückzahlen können).
Für die meisten der genannten Fehler gibt es Lösungsansätze, mit denen sie sich vermeiden oder beheben lassen. In diesem Arbeitspapier werden beispielhaft für alle fünf Phasen des Prozesses Lösungsvorschläge skizziert, die naturgemäß auf verschiedenen Ebenen ansetzen. Während manche dieser Ansätze Fehler in allen Phasen adressieren können, beziehen sich andere auf bestimmte Phasen. Eine unabhängige Prüfstelle („Algorithmen-TÜV“) würde es ermöglichen, den angemessenen und korrekten Einsatz von Entscheidungssystemen zu prüfen und auf Mängel in verschiedenen Phasen einzugehen. Ein Inputmonitoring würde prüfen, ob Trainingsdaten angemessen

und qualitativ gut sind (Phase 2). Ein falsch konstruiertes Entscheidungssystem kann durch Black-Box-Experimente entdeckt werden (Phase 3), mit denen sich die Funktionalität solcher Systeme testen lässt. Eine Professionsethik für Data Scientists, die in den meisten Phasen beteiligt sind, sowie ein Beipackzettel, der Anwendern die Interpretation der Ergebnisse erleichtert (Phase 4), können zu einem kompetenteren Umgang mit Entscheidungssystemen beitragen. Nicht zuletzt würde eine bessere externe Beforschbarkeit der algorithmischen Entscheidungssysteme generell die unabhängige Evaluation des Gesamtprozesses sicherstellen (vor allem in Phase 5).

Das Arbeitspapier verdeutlicht, dass es sich bei der Entwicklung von algorithmischen Systemen um einen komplexen Prozess mit vielen Entscheidungen und Verantwortlichkeiten handelt, der dadurch an vielen Stellen fehleranfällig ist. Es zeigt zudem, dass diese Fehler unterschiedlich komplex, folgenreich und beeinflussbar sind. Die ersten skizzierten Lösungsansätze geben jedoch Hinweise darauf, dass die meisten Fehlerquellen durch eine aktive Gestaltung vermieden oder behoben werden können.

3 Executive Summary

This working paper explains how automated decision-making systems (ADM-systems) are developed before being embedded in a social context, and highlights potential sources of error that can thereby occur.
Following an introductory section in which the terms “algorithm” and “decision-making system” are defined in more detail, the paper describes the development and embedding process of ADM-systems. This process comprises five phases. In the first phase, algorithms are designed and then implemented in software. An optional step in the second phase is operationalization, in which the constructs (e.g., the relevance of a message) are converted into indicators (e.g. frequency of clicks) and thereby rendered measurable. This second phase also involves the selection of data that will be used to train the algorithm, as well as the selection of a method for evaluation or prediction. In the third phase, a method of machine learning is combined with the training data, and the decision-making system is constructed. The process of embedding into social practice takes place in the fourth phase: The system is applied to new data, the results are interpreted and subsequently translated into action. The fifth phase encompasses an evaluation of the ADM-system.
The process that a decision-making system passes through – from development to evaluation – is long, and at many stages is dependent on numerous decisions. Along the way, different people are responsible in each individual phase: scientists and programmers, as well as data scientists and a variety of stakeholders (e.g., governmental, scientific or economic institutions, NGOs). However, the participating stakeholders frequently lack the competencies that are actually required for the task for which they are responsible, either because the programmers’ qualifications do not adequately reflect the social consequences of their work, or because end users lack the knowledge for a proper interpretation of the results.
The high number of decisions and different participants make the process vulnerable to errors that can occur in all phases. These differ in terms of scope as well as *detectability* and avoidability. For example, errors in the first phase of algorithm design can result in an algorithm that does not always calculate the correct result. However, when it is clear what the algorithm is intended to perform, and if the source code is available, such errors are relatively easy for computer scientists to detect and correct.
Operationalization errors in the second phase lead to results that cannot be interpreted meaningfully. Moreover, these kinds of errors are frequently difficult to detect and avoid. Deficiencies in data selection, however, can be detected, if there are known quality indicators for the data.
In the third phase, inconsistencies can occur in the construction of the ADM-system if the data and algorithm do not correspond. This likewise leads to unusable results.
In the fourth phase, namely the process of embedding in the social context, errors occur when users fail to recognize poor data and misinterpret the results. The process is particularly error-prone in this phase because users are interacting with the system, which can give rise to unintended effects.
Errors in the fifth phase occur as the result of missing or incorrect feedback. In general, the majority of ADM-systems make mistakes because there are no absolute decision rules that would result in a perfect classification of all data. At the same time, outdated or incorrect data can lead to mispredictions. These can also occur when the social situation has changed in such a way that once-found decision rules are no longer optimal. All types of incorrect decisions will remain in place if the system fails to receive the corresponding feedback. By adjusting the decision rules, some of these mispredictions can be avoided, albeit with considerable effort. On the other hand, such evolutionary progression is not possible if the system receives exclusively one-sided feedback (e.g., no feedback regarding the fact that an individual denied credit was actually able to repay).
There are solutions for avoiding or eliminating the majority of the aforementioned errors. In this working paper, the proposed solutions are outlined by way of example for all five phases of the process. By definition, these are targeted at different levels. While some of these approaches are able to address errors in all phases, others relate to specific phases. An independent inspection body (“roadworthy testing” for algorithms) would facilitate inspections for the appropriate and correct application of algorithmic decision-making systems, and could address deficiencies at various stages. Input monitoring would determine whether training data is adequate and of good quality (Phase 2). An improperly constructed decision-making system can be identified through black-box experiments (Phase 3), which are used to test the functionality of such systems. A set of professional ethics for data scientists, who are involved in most phases, as well as an instructional leaflet aimed at helping users to interpret results (Phase 4)

could contribute to the more competent utilization of decision-making systems. Not least, improved external explorability of ADM-systems would generally safeguard the independent evaluation of the overall process (particularly in Phase 5).

The working paper makes clear that developing algorithmic systems is a complex process involving numerous decisions and responsibilities that render it prone to error at many stages. It also shows that these errors vary in complexity, their consequences and the extent to which they can be influenced. However, the solutions outlined here provide guidance on how active design can serve to avoid or remedy most sources of error.

4 Worum es geht: Definition und Potenzial von Algorithmen

Bevor der Entwicklungs- und Einbettungsprozess von Entscheidungssystemen erläutert wird, werden in diesem Kapitel zunächst die Begriffe Algorithmus und Entscheidungssystem definiert. Dieses Arbeitspapier fokussiert auf solche Algorithmen, die die Teilhabe[2] einzelner Menschen beeinflussen können, denn bei ihnen haben Fehler die größten Konsequenzen. Deshalb gehört zu einer Definition von Entscheidungssystemen auch, die Teilmenge von Algorithmen zu bestimmen, die das größte Potenzial aufweisen, die Teilhabe positiv oder negativ zu beeinflussen. Abschließend werden die möglichen Vorteile solcher Algorithmen im Vergleich zu menschlichen Entscheidungen dargestellt.

Algorithmen

Algorithmen sind informatische Werkzeuge, um mathematische Probleme automatisiert zu lösen. Sie berechnen zuverlässig eine Lösung für ein Problem, wenn sie die dafür nötigen Informationen bekommen, den sogenannten „Input". Das mathematische Problem definiert, welche Eigenschaften der dazugehörige Output, also das Resultat der Berechnung, haben soll – es gibt aber selbst nicht an, wie man zu dieser Lösung kommt oder ob es überhaupt einen zuverlässigen Weg gibt, der für jeden zulässigen Input auch den korrekten Output berechnet. Ein typisches Beispiel für ein mathematisches Problem ist die Berechnung einer Fahrstrecke von A nach B. Als Input dient eine Straßenkarte, in der alle Straßen und ihre Länge gespeichert sind. Dazu kommen noch der jetzige Standort und das Ziel der Fahrerin. Basierend auf diesem Input soll eine Strecke berechnet werden, die die kürzeste Länge oder wahlweise die kürzeste erwartete Fahrzeit hat (Output). Das Problem an sich beschreibt also nur das Verhältnis von Input zu gewünschtem Output. Der Lösungsweg von Input zu Output wird im Algorithmus beschrieben.[3]

Viele dieser Algorithmen lösen sehr einfache Probleme: Sie finden in einer Datenbank den Eintrag, der zu einer bestimmten Kunden-ID gehört, oder berechnen, auf welchem Weg eine E-Mail durch das Internet geschickt werden soll. Unter der Voraussetzung, dass keine handwerklichen Fehler gemacht wurden, sind diese Berechnungen fehlerfrei und objektiv. Trotzdem können sie Auswirkungen auf die Gesellschaft haben: Die Verantwortlichen in einem Logistikunternehmen können zum Beispiel entscheiden, dass ihnen der Kundenservice wichtiger ist als die Umwelt, und deshalb die Produkte so schnell aus ihrem Lager verschicken wie möglich anstatt Pakete zu bündeln. Dadurch wird sowohl die gemeinschaftlich bezahlte Infrastruktur stärker abgenutzt als auch der Benzinverbrauch erhöht – beides gesellschaftlich relevante Ressourcen. Diese Algorithmen stehen bisher aber nicht im Verdacht, die gesellschaftliche Teilhabe von Menschen zu erhöhen oder zu senken, und sollen daher hier nicht betrachtet werden.

Lernende Algorithmen

Es gibt eine spezielle Klasse von Algorithmen, die das aktuelle Verhalten von Menschen bewerten und/oder Vorhersagen über das zukünftige Verhalten von Personen machen und dafür in vielen Fällen aus vorher erhobenen Daten lernen. Unter „Lernen" versteht man im Bereich der künstlichen Intelligenz und des maschinellen Lernens

[2] „Im Rahmen dieser Studie umfasst der Begriff Teilhabe die gleichberechtigte Einbeziehung von Individuen und Organisationen in politische Entscheidungs- und Willensbildung sowie die faire Partizipation aller an sozialer, kultureller und wirtschaftlicher Entwicklung. Es geht also erstens um Teilhabe an demokratischen Prozessen – und damit um politische Gleichberechtigung – und zweitens um Teilhabe an Errungenschaften eines sozialen Gemeinwesens, ‚angefangen von guten Lebens- und Wohnverhältnissen, Sozial- und Gesundheitsschutz, ausreichenden und allgemein zugänglichen Bildungschancen und der Integration in den Arbeitsmarkt bis hin zu vielfältigen Freizeit- und Selbstverwirklichungsmöglichkeiten' (Beirat Integration 2013: 1)" (Vieth und Wagner 2017: 9).

[3] Es gibt mathematische Probleme, für die es keinen Algorithmus gibt, um sie zu lösen. Am bekanntesten ist das sogenannte „Halteproblem", die Frage danach, ob ein Computerprogramm mit einem bestimmten Input jemals zum Ende seiner Berechnungen kommen wird. Es gibt keinen Algorithmus, der dies für alle Computerprogramme zuverlässig berechnen kann.

das Finden von Mustern in großen Datenmengen, die mit dem zu bewertenden oder zu prognostizierenden Verhalten korrelieren (Flach 2012). Diese Muster werden in verschiedenen Arten von Strukturen („Modelle“ genannt) abgelegt, die es dann erlauben, weitere Daten derselben Art ebenfalls in diese Muster einzuordnen.

Das mathematische Problem, das diese Algorithmen lösen, ist also: Gegeben ist eine Menge von Daten (z. B. bisherige Leistungen von Schülern plus weitere Beobachtungen und persönliche Daten), finde heraus, welche dieser Informationen am meisten mit dem Lernerfolg zusammenhängen. Diese Korrelationen werden in unterschiedlicher Art und Weise erhoben und abgespeichert. Daher gibt es eine große Menge von Algorithmen des maschinellen Lernens, die jeweils für unterschiedliche Fragestellungen unterschiedlich gut geeignet sind, um Regeln in Daten zu finden und abzuspeichern.

Die meisten lernenden Algorithmen bewerten Daten auf einer Skala („Scoring“) oder teilen sie in Klassen ein (Klassifikation). Sie haben das Potenzial, das Wissen von beliebig vielen menschlichen Experten in sich zu vereinigen und darüber hinaus von einer nahezu unbegrenzten Datenmenge zu lernen. Lernende Algorithmen können zumindest im Prinzip so konstruiert werden, dass sie weder nach Geschlecht oder Herkunft diskriminieren noch von Gefühlen in ihrer Urteilsfindung beeinträchtigt werden. Damit wird auch schon ein Teil ihres Potenzials beschrieben, die gesellschaftliche Teilhabe von Menschen zu vergrößern, die bisher unter Diskriminierung zu leiden haben.

(Lernende) Algorithmen finden sich – insbesondere in Deutschland – heute hauptsächlich in sogenannten „Entscheidungsunterstützungssystemen“ („Decision Support Systems“), die es ausgebildeten Experten erlauben sollen, sich eine zweite Meinung einzuholen.[4] Hier wird deren Entscheidung also nur unterstützt. Neben Systemen, die Entscheidungen unterstützen, existieren solche, die automatisch eigenständige Entscheidungen treffen (vgl. Abbildung 1). Es ist unklar, zu welchem Anteil sie heute schon in Algorithmen in Deutschland eingesetzt werden, um tatsächlich eigenständig zu entscheiden. Im Folgenden wird eine Hardware oder Software als „Automated Decision Making System“ (AuDM System) bezeichnet, wenn sie

- erstens durch einen Algorithmus eine Bewertung einer Situation oder eines Menschen vornimmt oder eine Vorhersage über die Wahrscheinlichkeit des Eintretens einer Situation trifft,
- zweitens daraufhin eine Software oder Hardware aktiviert, die auf Grundlage der Bewertung oder Prognose eine Entscheidung trifft, deren Aktion unmittelbar das Leben eines Menschen betrifft.

Ein Beispiel dafür wäre ein Algorithmus, der die finanzielle Situation einer Person bewertet. Daraufhin aktiviert dieser Algorithmus einen weiteren, der die Entscheidung trifft, dieser Person keine Arbeitslosenhilfe mehr auszuzahlen, und einen Prozess in Gang setzt, der die dafür notwendigen Eintragungen in den Datenbanken macht und den Auszahlungsauftrag stoppt. Ein extremes Beispiel ist eine Drohne, die automatisch Gesichter mit einer Terroristendatenbank abgleicht und bei genügend hoher Passgenauigkeit die vermeintlich identifizierte Person über Aktivierung einer Waffe tötet.

[4] Entscheidungsunterstützungssysteme müssen keine lernenden Anteile besitzen – sie können auch auf explizit aufgestellten Regeln beruhen. Solche sogenannten „Expertensysteme“ sind ebenfalls oft intransparent, aber zumindest gemeinsam von Domänenexperten und Informatikern aufgestellt. Zudem sind mögliche Fehlentscheidungen leichter zu überprüfen, weil diese Systeme jederzeit die Regeln angeben können, nach denen sie zu einer Entscheidung gekommen sind. Daher werden in diesem Report hauptsächlich die Chancen und Risiken von lernenden Algorithmen behandelt.

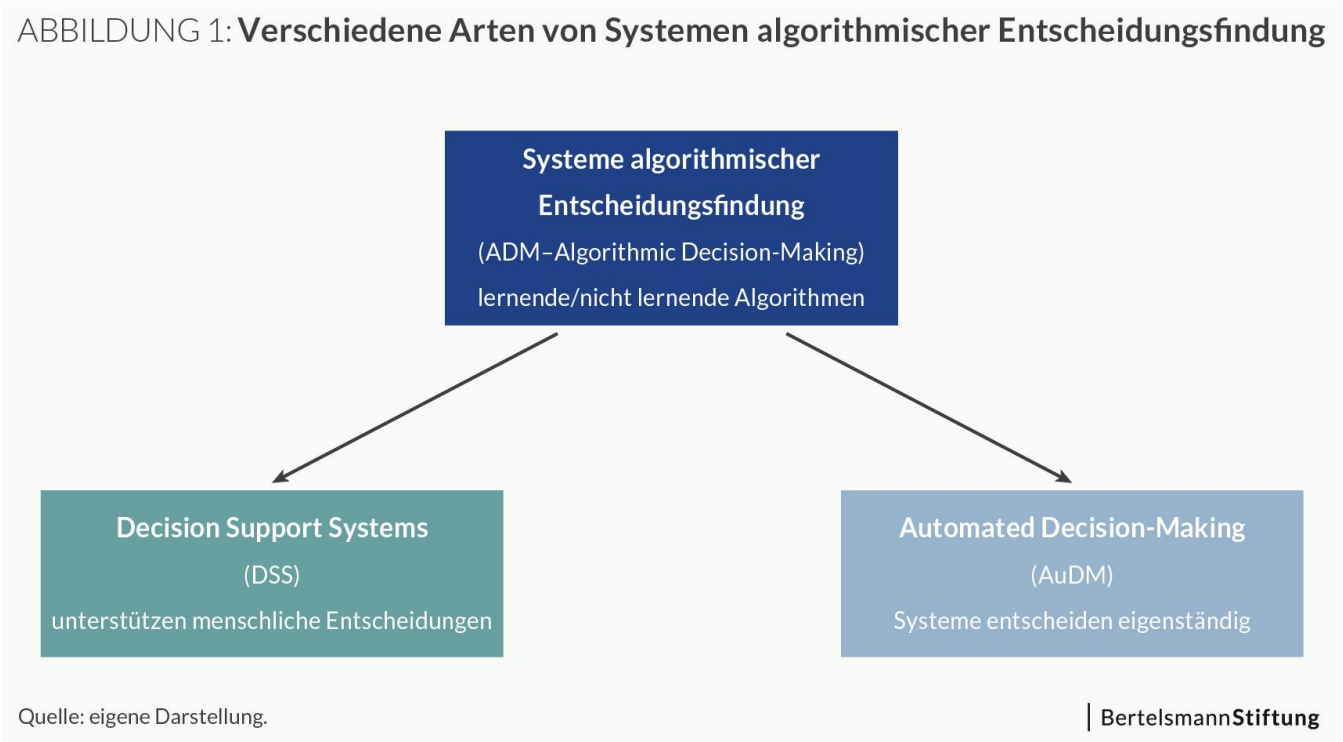


Abbildung 1: Verschiedene Arten von Systemen algorithmischer Entscheidungsfindung (Quelle: eigene Darstellung)

Verhältnis von algorithmischen Entscheidungssystemen und Algorithmen

Es ist wichtig zu betonen, dass algorithmische Entscheidungssysteme, wenn sie Algorithmen aus dem maschinellen Lernen beinhalten, aus (mindestens) **zwei** Algorithmen bestehen, deren Wirkungsweise nicht unabhängig voneinander zu betrachten ist. Es ist in diesem Falle irreführend, das ADM-System selbst als **einen** Algorithmus zu bezeichnen. Der erste Algorithmus lernt aus Daten, wie Personen in der Vergangenheit kategorisiert wurden oder welches Verhalten sie zeigten. Das daraus entstehende Regelwerk wird gespeichert. Um eine Entscheidung über eine Person zu fällen, werden dann deren Daten in das Regelwerk eingebracht und der zweite Algorithmus berechnet daraus die Kategorie, in die diese Person fällt, bzw. gibt eine Einschätzung über die Wahrscheinlichkeit des zukünftigen Verhaltens dieser Person ab. Dieser zweite Algorithmus ist meist extrem simpel und ist der Teil, der von den Anwendern als Algorithmus wahrgenommen wird: Er bekommt Daten und liefert eine Ausgabe. Der eigentliche Algorithmus von Interesse ist aber derjenige, der das Regelwerk liefert, nachdem Personen dann letztendlich klassifiziert werden oder ihr Verhalten vorhergesagt wird. Insofern sind algorithmische Entscheidungssysteme **keine Teilmenge** der Algorithmen, sondern enthalten Algorithmen. Da zudem die Daten, aus denen der erste Algorithmus die Entscheidungsregeln extrahiert hat, grundlegend für die Wirkungsweise des zweiten Algorithmus sind, sprechen wir von einem **System** der algorithmischen Entscheidung oder Entscheidungsunterstützung.

Teilhaberelevante Algorithmen und algorithmische Entscheidungssysteme

Um im Sinne dieser Studie teilhaberelevant zu sein, müssen Algorithmen oder algorithmische Entscheidungssysteme Entscheidungen unterstützen oder treffen, die eine Auswirkung auf das Leben von Personen haben.

Das ist nicht bei allen der Fall: Algorithmische Entscheidungssysteme können zum Beispiel auch in der Produktion genutzt werden, um per Kamera vermutlich beschädigte Produkte auszusortieren, die dann je nach Qualität des Algorithmus und Kosten des Produktes noch einmal von einer Person durchgesehen oder direkt entsorgt werden.

Ein solches Gesamtsystem kann zwar beispielsweise die Anzahl an Arbeitsplätzen von Niedrigqualifizierten verringern und wirkt sich somit insgesamt auf Teilhabemöglichkeiten aus, aber nicht auf dem individuellen Level (vgl. zur Einschätzung des Teilhabewirkungspotenzials von Algorithmen Vieth und Wagner 2017).

Ein weiteres Beispiel sind Algorithmen, die die Relevanz von Nachrichten berechnen und die Verbreitung von Informationen steuern. Sie beeinflussen den gesellschaftlichen Diskurs und werfen Fragen zu Relevanzbewertung und Meinungsvielfalt auf. Damit wirken sie sich auf gesamtgesellschaftlicher Ebene auf Teilhabechancen aus, aber nicht auf individueller Ebene. Derartige Fragen werden an anderer Stelle näher diskutiert (vgl. Lischka und Stöcker 2017) und hier ausgeklammert.

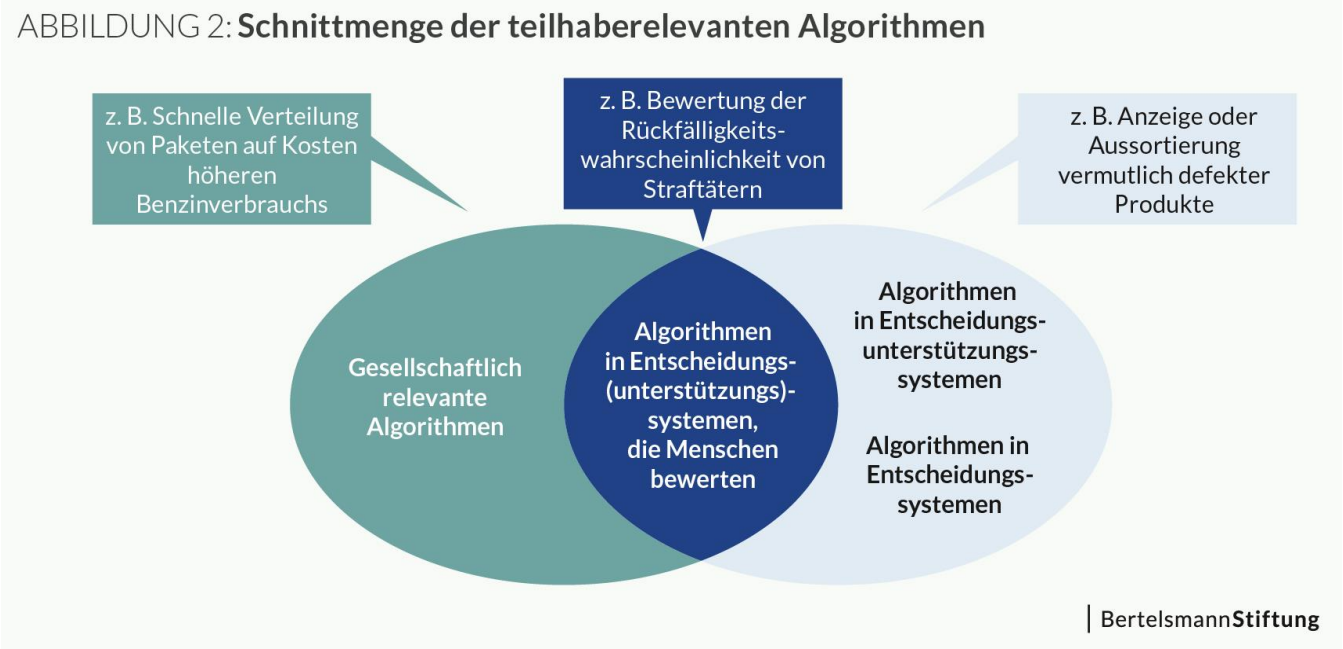


Abbildung 2: Schnittmenge der teilhaberelevanten Algorithmen (Quelle: eigene Darstellung)

Abbildung 2 zeigt, dass es zum einen viele Algorithmen gibt, die gesellschaftlich relevante Konsequenzen nach sich ziehen können. Ein Beispiel ist der oben genannte Algorithmus, mit dem Pakete möglichst schnell verteilt werden, aber auf Kosten höheren Benzinverbrauchs, wodurch die Infrastruktur abnutzt und die Umwelt belastet wird. Zum anderen sind Algorithmen häufig Teil von komplexeren Entscheidungssystemen („Decision Support“ oder „Automated Decision-Making Systems“), die aber in vielen Fällen keine gesellschaftlich relevanten Auswirkungen haben. Nur eine kleine Schnittmenge von Algorithmen wird in Entscheidungssystemen eingesetzt, die Auswirkungen auf einzelne Personen haben. In dieser Expertise geht es um diejenige Schnittmenge an Algorithmen, die aufgrund von historischen Daten das aktuelle Verhalten von Menschen bewerten oder ihr zukünftiges Verhalten prognostizieren und damit Entscheidungen vorbereiten oder selbstständig treffen, die direkt die Teilhabe von Individuen befördern oder behindern können.

Vergleich automatischer und menschlicher Entscheidungen

Um das positive Potenzial von Entscheidungsunterstützungssystemen und automatischen Entscheidungssystemen einordnen zu können, ist es zweckmäßig, sie menschlichen Entscheidern gegenüberzustellen, die sie ersetzen oder ergänzen sollen. Menschliche Entscheider haben unterschiedliche Qualifikationen – vom Laien bis zum ausgebildeten Experten. Wie genau Menschen entscheiden, ist immer noch Gegenstand der Forschung. Die Tendenz der aktuellen Forschung geht dahin, dem Menschen das rationale Denken in entscheidenden Situationen

abzusprechen. Nachdem im späten 19. Jahrhundert der Begriff des „Homo oeconomicus“ geprägt wurde und die Spieltheorie Vorhersagen über das Verhalten in diesem Modell machte, war der zweite Teil des 20. Jahrhunderts von Forschungen bestimmt, die das irrationale Verhalten des Menschen hervorhoben. Forscher wie Kahnemann und Tversky (Kahnemann 2012) oder Ariely (2010) wiesen nach, dass kognitive Verzerrungen menschliches Entscheiden beeinflussen, dass sich Personen manipulieren lassen und nicht immer die optimale Entscheidung treffen – wobei die „Optimalität“ durch das „Homo oeconomicus“-Modell vorgegeben war. Neuere Forschungen zeigen, dass Menschen durchaus optimieren, dabei aber nicht immer rein ökonomisch vorgehen, sondern auch andere Aspekte miteinbeziehen, wie beispielsweise ihre begrenzte Energie (Aufmerksamkeitsökonomie; vgl. Ariely 2010). Dazu kommen nachweisbare Tagesformeffekte (Danziger 2011) und explizite oder implizite Vorurteile, die zu falschen Bewertungen des Verhaltens oder zukünftigen Verhaltens eines Menschen führen können. Diese subjektiven Entscheidungen haben jedoch auch eine positive Seite. Denn Menschen bewerten den Einzelfall und können auch Kriterien berücksichtigen, die eigentlich aus dem Bewertungsraster fallen. So kann etwa ein Bewerber mit durchschnittlichen Noten trotzdem eingestellt werden, weil er im Vorstellungsgespräch durch seine Präsentationsfähigkeiten und seine Überzeugungskraft beeindruckt. Menschen können daher bei unerwarteten Abweichungen flexibel reagieren und Entscheidungen treffen, die von den vorgegebenen Kriterien abweichen. Algorithmen können dagegen ganz prinzipiell keine Ermessenspielräume abbilden.

Gerade im Vergleich zu algorithmischen Entscheidungen sind Menschen jedoch langsam und limitiert in der Informationsverarbeitung: Das macht das menschliche Bewerten teuer und erlaubt es dem Einzelnen nicht, von beliebig vielen Situationen zu lernen. Ein Beispiel dafür ist die Behandlung sogenannter „seltener Krankheiten“. Das sind Krankheiten, von denen weniger als eine von 10.000 Personen betroffen sind. Es ist offensichtlich, dass die absolute Anzahl der Betroffenen global gesehen in vielen Fällen grundsätzlich groß genug ist, um systematisch Erfahrungen in ihrer Behandlung zu sammeln, dass aber der einzelne Arzt in seinem Leben nicht genügend von ihnen kennenlernen wird, um zum Experten zu werden. Die Definition der „Seltenheit“ ist also direkt abhängig von der begrenzten Lebenszeit und Verarbeitungsgeschwindigkeit menschlicher Experten.

Damit lassen sich die möglichen Vorteile von algorithmischen Entscheidungs- oder Entscheidungsvorbereitungssystemen direkt ableiten, die grundsätzlich das Potenzial haben, mehr gesellschaftliche Teilhabe zu ermöglichen:

1. Algorithmen können aus nahezu beliebig vielen Datenpunkten lernen. Die Menge der als selten anzusehenden Ereignisse wird dadurch erheblich eingeschränkt und die Menge der Situationen, über die etwas gelernt werden kann, stark erhöht.
2. Solange sie keinen Input als Trainingsdatensatz bekommen, der eine Diskriminierung beinhaltet, und solange eine Diskriminierung nicht explizit in den Programmcode implementiert wird, können Algorithmen diskriminierungsfrei entscheiden. Dazu ist es allerdings nötig, dass sich Gesellschaft auf einen in Zahlen messbaren Diskriminierungsbegriff einigt.
3. Algorithmen kommen für denselben Input immer auf denselben Output – sie sind nicht tagesformabhängig und nicht bestechlich.
4. Algorithmen entscheiden optimal nach den vorgegebenen Kriterien – wenn es überhaupt technisch möglich ist, das Optimum zu berechnen. Ist dies nicht möglich, müssen sie – so wie der menschliche Experte auch – Heuristiken verwenden, um eine Lösung zu finden, die möglichst nah am Optimum ist. Anders als Menschen unterliegen Algorithmen bei einer heuristischen Vorgehensweise keinen kognitiven Verzerrungen.
5. Sind Algorithmen erst einmal trainiert, können sie leicht kopiert werden und in kürzester Zeit hochwertige Entscheidungen in großer Zahl vorbereiten oder treffen. Dies erlaubt aber auch eine schnelle Monopolisierung der Entscheidungsfindung durch einen einzigen Algorithmus.
6. Lernende Algorithmen sind in der Lage, Informationen, Produkte und Dienstleistungen zu personalisieren. Sie können Menschen damit an ihren Bedürfnissen ausgerichtet bei verschiedenen Aufgaben wie zum Beispiel der Informationssuche oder Produktauswahl unterstützen.

Bei den ersten vier Punkten geht es vor allen Dingen um die Erhöhung gesellschaftlicher Teilhabe durch objektivere, diskriminierungsfreie Entscheidungssysteme. Die beiden letzten Eigenschaften sind mindestens ebenso

wichtig. Denn Skalierung und Personalisierung führen dazu, dass Wissen und Dienstleistungen, die zuvor einer relativ kleinen Bevölkerungsgruppe vorbehalten waren, nun für eine breite Masse zugänglich werden. So konnte sich oft nur derjenige spezifisches Expertenwissen aneignen, der sich die teure Beratung und Training durch menschliche Fachleute leisten konnte. Durch personalisierte Informations- und Bildungsangebote steht diese Möglichkeit nun deutlich mehr Menschen zur Verfügung. Dies kommt insbesondere den Menschen zugute, deren Problem zu selten war, um es in der analogen Welt mithilfe von eigens für sie entwickelten Systemen oder Prozessen zu bearbeiten (vgl. zu Chancen, Risiken und Handlungsbedarfen auch Lischka und Klingel 2017).

Diese Potenziale können Algorithmen jedoch nur in Verbindung mit menschlichen Entscheidungen entfalten. Denn es sind Menschen, die Algorithmen programmieren und die Ziele algorithmischer Entscheidungsfindung festlegen. Nur Menschen sind dazu in der Lage, sicherzustellen, dass Diskriminierungen nicht in den Programmcode implementiert werden. Nur sie können über den ethischen Einsatz von Algorithmen entscheiden und bestimmen, ob diese tatsächlich zu mehr Teilhabe führen.

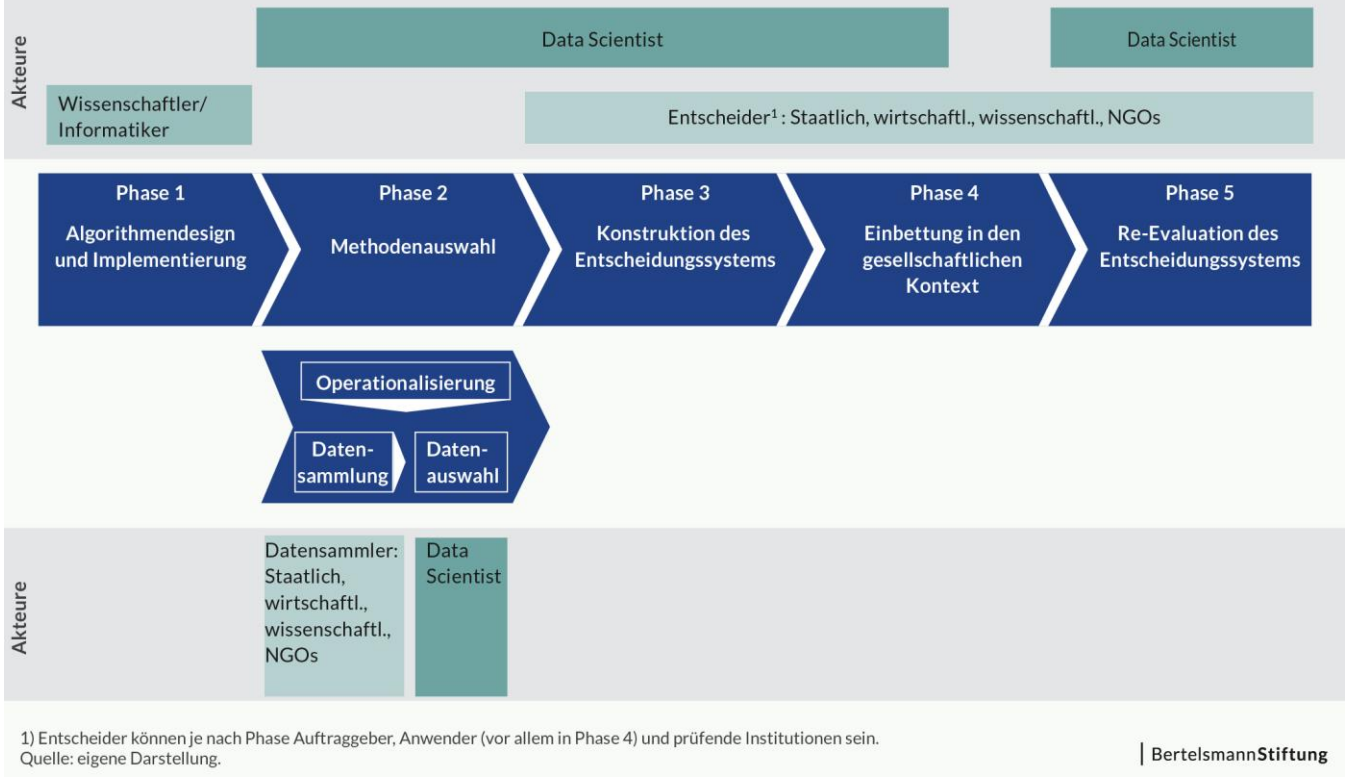
Nachdem definiert wurde, was ein Algorithmus ist und welche Arten von Algorithmen als Bestandteil von algorithmischen Entscheidungssystemen in diesem Papier betrachtet werden, wird im Folgenden der Prozess dargestellt, in dem diese Entscheidungssysteme entwickelt und in einen gesellschaftlichen Prozess eingebettet werden.

5 Was geschieht: Entwicklungs- und Einbettungsprozess von Entscheidungssystemen

Die Entwicklung von algorithmischen Entscheidungssystemen ist momentan noch von höchst unterschiedlicher Qualität. Auch die Folgen ihrer Einbettung in gesellschaftlich relevante Prozesse sind kaum erforscht. Dabei wird der Begriff „gesellschaftlicher Prozess“ hier sehr weit verstanden und umfasst Dinge wie die Auswahl von Bewerbern für Jobinterviews, die Bewertung von Angeklagten bezüglich ihres aktuellen oder künftig erwartbaren kriminellen Verhaltens oder die Vergabe von Krediten. Um mehr Transparenz über algorithmische Entscheidungen oder Entscheidungsvorbereitungen zu erlangen, ist es zuerst notwendig, die Entwicklung derartiger Systeme und die lange Kette an Verantwortlichkeiten in diesem Prozess zu skizzieren. Das ist die notwendige Grundlage, um Fehlerquellen zu erkennen (vgl. Kapitel 5).

Der Entwicklungs- und Einbettungsprozess algorithmischer Entscheidungssysteme vollzieht sich im mehreren Phasen (vgl. auch Abbildung 3):

ABBILDUNG 3: **Entwicklungs- und Einbettungsprozess von algorithmischen Entscheidungssystemen mit verantwortlichen Akteuren**



1) Entscheider können je nach Phase Auftraggeber, Anwender (vor allem in Phase 4) und prüfende Institutionen sein.
Quelle: eigene Darstellung.

BertelsmannStiftung

Abbildung 3: Entwicklungs- und Einbettungsprozess von algorithmischen Entscheidungssystemen mit verantwortlichen Akteuren (Quelle: eigene Darstellung).

Phase 1: Algorithmendesign und Implementierung

Die Entwicklung eines Entscheidungssystems beginnt mit der Entwicklung eines Algorithmus durch **Wissenschaftler und Informatiker** – seltener auch interessierte Laien. Anschließend transferieren Wissenschaftler oder Informatiker die für Menschen lesbare Beschreibung des Algorithmus in eine Programmiersprache. Dieser Transfer wird Implementierung genannt. In den meisten Fällen wird nur ein Bruchteil der Algorithmen auch kommerziell oder anderweitig für die weitere Nutzung implementiert. Die Implementierung wird daher von Angestellten von Firmen,

teilweise aber auch freiwillig und unentgeltlich von einer Gemeinschaft von Programmierern geleistet. Diese veröffentlichen die Algorithmen in sogenannten „Softwarepackages“ frei verfügbar und oftmals auch kostenlos oder schreiben sogar eine Software, in die die Analysemethoden zur bequemen Anwendung eingebettet sind. Häufig werden Algorithmen nur im Blick auf eine einzige Anwendung entwickelt oder lösen erst mal ein abstraktes Problem. Dank ihrer Wandelbarkeit und Generalität können sie aber oft auf viele verschiedene Problemstellungen angewendet werden.

Phase 2: Methodenauswahl

Die Auswahl einer Bewertungs- oder Vorhersagemethode ist ein entscheidender Schritt im Entwicklungsprozess eines Entscheidungssystems – es gibt mehrere Dutzend verschiedene Methoden des maschinellen Lernens, jede mit ihren eigenen Schwächen und Stärken. Die Auswahl der Methode entscheidet darüber, nach welcher Art von Mustern in den Daten gesucht wird und wie diese nachher über neue Daten entscheiden. Die Auswahl der Methode hat demnach einen erheblichen Einfluss auf die Ergebnisse des Prozesses. Die Methodenselektion wird in der Regel von einem **Data Scientist** durchgeführt. Als Data Scientist werden hier alle Personen bezeichnet, die in Daten mithilfe von Algorithmen nach Mustern suchen, die mit (aktuellem oder zukünftigem) Verhalten korrelieren und daher eine Bewertung menschlichen Verhaltens oder eine Vorhersage ermöglichen. Es handelt sich dabei momentan um kein klares Berufsbild. Es ist auch nicht mit einer klassischen Ausbildung assoziiert. Ebenfalls fehlen klare Leistungsprofile oder Evaluationsmöglichkeiten, um zu bestimmen, ob jemand, der als Data Scientist arbeitet, auch die dafür nötigen Kenntnisse hat. Die meisten heute als Data Scientist arbeitenden Personen haben eine Ausbildung als Informatiker, Mathematiker oder Physiker. Durch die weite Zugänglichkeit von Datenanalysemethoden und die Möglichkeit, Zertifikate als Data Scientist durch Prüfungen nach der Teilnahme an „Massive Open Online Courses“ (MOOCs) zu erwerben, gibt es auch Quereinsteiger mit anderen Lebensläufen in diesem Berufsfeld.

Datensammlung und -auswahl: Lernende Entscheidungssysteme benötigen als Grundlage Trainingsdaten, mit denen sie Muster lernen können, um diese dann auch in neuen Daten zu erkennen (z. B. Bilder von Gesichtern, um diese erkennen zu lernen). Die Input- oder Trainingsdaten werden von unterschiedlichen Akteuren gesammelt, zum Beispiel von staatlichen, wirtschaftlichen oder wissenschaftlichen Institutionen. Diese können entweder die Auftraggeber des Entscheidungssystems sein oder andere **datensammelnde Akteure** wie zum Beispiel auch Datenhändler. Ausgewählt werden die Daten dann von **Data Scientists**.

Daten- und Methodenauswahl folgen keiner festgelegten Reihenfolge. Vielmehr handelt es sich um einen iterativen Prozess, in dem verschiedene Daten und Methoden kombiniert und ausgetestet werden können.

Operationalisierung: Bei der Datenerhebung und -auswahl findet häufig eine sogenannte Operationalisierung statt. Durch sie werden abstrakte Konzepte messbar gemacht, die nicht direkt beobachtbar sind, wie zum Beispiel „Kreditwürdigkeit“ bei einer Person, die noch keinen Kredit aufgenommen hat, oder „Relevanz einer Nachricht“. Die Möglichkeiten, solche Konstrukte zu messen, sind häufig dadurch eingeschränkt, dass die Personen nicht direkt befragt werden können, sondern ihr Verhalten aus einem digitalen Log abgeleitet wird (z. B. Aufrufe von Webseiten als Indikator für Interesse an einem bestimmten Thema). Zudem werden teilweise Indikatoren herangezogen, die nicht direkt mit dem Konstrukt in Verbindung stehen (z. B. die Kaufhistorie bei Amazon oder die Kreditwürdigkeit der Facebookfreunde als Indikatoren für die eigene Kreditwürdigkeit). Weitere Beispiele für eine Operationalisierung sind:

- Die COMPAS-Software zur Prognose operationalisiert das Delinquenzrisiko unter anderem durch das Konstrukt „soziales Umfeld“. Dieses wiederum wird durch Fragen wie: „Wenn Sie mit beiden Eltern aufgewachsen sind, diese sich aber später getrennt haben, wie alt waren Sie da?“ oder „War ein Elternteil im Gefängnis?“ messbar gemacht.

- Ein anderes Beispiel für eine Operationalisierung sind bibliometrische Kennzahlen zur Leistungsmessung von Wissenschaftlern. So beschreibt der h-Index eines Wissenschaftlers die Anzahl seiner Publikationen, die mindestens genauso viele unabhängige Zitate erreicht haben. Ein h-Index von 13 zeigt also an, dass der Wissenschaftler 13 wissenschaftliche Artikel geschrieben hat, die mindestens 13-mal zitiert wurden. Der h-Index ist eine wichtige Entscheidungsgrundlage, wenn es darum geht die, Professorabilität von Wissenschaftlern zu bewerten.

Die Operationalisierung kann von den verschiedenen **beteiligten Akteuren** beeinflusst werden, die die Daten sammeln, oder sie kann vom **Data Scientist** festgelegt werden, der die Daten auswählt. Da in Zeiten von Big Data die Daten oft aus verschiedenen Datenbanken zusammengeführt werden, können hier auch jeweils mehrere Individuen beteiligt sein.

Phase 3: Konstruktion des Entscheidungssystems

Im Entscheidungssystem wird eine Methode des maschinellen Lernens mit den ausgewählten Trainingsdaten zusammengebracht. Das trainierte System wird anschließend evaluiert. Die Konstruktion des Entscheidungssystems resultiert aus der Operationalisierung sowie der Daten- und Methodenauswahl zusammen mit der Wahl entsprechender Parameter, die die Methode benötigt. Dementsprechend sind daran ebenfalls sowohl die jeweiligen **Entscheider** als auch **Data Scientists** beteiligt.

Die eigentliche Konstruktion ist abhängig von einem Qualitätskriterium, mit dem während des Trainings festgestellt werden kann, ob das Entscheidungssystem schon gut genug ist. Wenn dies nicht der Fall ist, gibt es meistens mehrere Parameter, die der Data Scientist verändern kann, um bessere Regelwerke ableiten zu können. Oftmals werden auch verschiedene Methoden des maschinellen Lernens ausprobiert, bis die beste gefunden ist. Da somit die Auswahl des finalen Systems entscheidend vom gewählten Qualitätskriterium abhängt, muss dieses ebenfalls sorgfältig aus einer Reihe von ungefähr zwei Dutzend Maßen ausgewählt werden. Auch diese Entscheidung treffen häufig die beteiligten Data Scientists.

Phase 4: Einbettung in den gesellschaftlichen Prozess (Einsatz des Systems – Ergebnisinterpretation – Aktion)

Die Einbettung beschreibt den Prozess, der entscheidet, wie das Entscheidungssystem angewendet wird, wie die Ergebnisse interpretiert werden und indem die Aktion festgelegt wird, die aus den Ergebnissen resultiert. Vor dem Einsatz des Systems werden die beteiligten **Akteure (Entscheider)** als Anwender in dessen Funktionen eingewiesen, geschult oder erhalten eine Handreichung. Die Anwender füttern das trainierte System mit neuen Daten, die bewertet werden sollen oder auf denen basierend eine Vorhersage bezüglich zukünftigen Verhaltens gemacht werden soll. Manchmal ist auch der **Data Scientist** noch mit diesen Aufgaben befasst, beispielweise dann, wenn die Resultate graphisch visualisiert werden (Farbskalen, Einteilen einer kontinuierlichen Skala in verschiedene visuelle Kategorien, Charts …). Dadurch wird eine bestimmte Interpretation nahegelegt. Hier entscheidet also ein Programmierer, wie die Resultate dem Anwender präsentiert werden, und leistet damit einen Teil der Interpretationsarbeit. Meistens werden aber die Anwender auch einen Teil der Interpretation leisten, indem sie bestimmen, welche (Re-)Aktion aus dieser Interpretation folgt. Das können verschiedene Akteursgruppen sein, die wiederum jeweils mehrere Individuen umfassen können. Bei Algorithmen, die im Justizsystem eingesetzt werden, können dies beispielsweise Richter, Vollzugsbeamte oder auch Sozialarbeiter sein. Wenn es sich um ein automatisches Entscheidungssystem handelt, kann ein nachgeschalteter Algorithmus basierend auf den Ergebnissen des ersten Systems die Aktion selbstständig auswählen und in Gang setzen.

Phase 5: Re-Evaluation

Am Ende des Prozesses kann das Ergebnis evaluiert werden. Dafür sind entweder der **Data Scientist** oder die beteiligten Akteure als **Anwender** selbst verantwortlich. Diese Bewertung kann dann als Feedback in den Prozess zurückgegeben werden. So können etwa aufgrund des Feedbacks die Daten verändert werden, die als Trainingsdaten ausgewählt werden. Auch die Operationalisierung, Methode oder das Entscheidungssystem als solches können entsprechend angepasst werden (vgl. Rückkopplungspfeile in Abbildung 3). So wurde etwa ein automatisiertes Entscheidungssystem verändert, das im Rekrutierungsprozess einer Firma eingesetzt wurde. Das System identifizierte Indikatoren, die darauf schließen ließen, dass Angestellte schnell wieder kündigten. Ein solches Kriterium war die Distanz zum Firmensitz, das jedoch diejenigen Bewerber diskriminierte, die sich keine Wohnung im teuren Umfeld des Firmensitzes in der Stadtmitte leisten konnten. Nachdem dies erkannt wurde, wurde das Kriterium ausgeschlossen und das Entscheidungssystem entsprechend angepasst (Walker 2012).

Der Prozess von der Entwicklung eines (allgemeinen) Algorithmus bis hin zu seiner Einbettung in einen gesellschaftlichen Prozess ist lang und involviert leicht Hunderte Personen. Wenn es ein System ist, das anschließend nachher von vielen genutzt wird, können es auch Tausende von Personen sein, die mit dem System interagieren und ihre Entscheidungen davon abhängig machen. Ein gutes Beispiel dafür ist die Entwicklung von Entscheidungssystemen zur Rückfälligkeitsvorhersage von Kriminellen: Die eigentlichen Algorithmen sind vermutlich Standardalgorithmen des maschinellen Lernens, die in den letzten zehn bis zwanzig Jahren entwickelt wurden. Es ist zu vermuten, dass das Entwicklerteam aus einer Handvoll von Personen besteht, die das System aufsetzen und trainieren. Im laufenden Betrieb erheben Sozialarbeiter und Vollzugsbeamte die Daten der zu Bewertenden, geben die Daten in das System ein und interpretieren auch die Vorhersagen des Algorithmus – manchmal natürlich auch andere Personengruppen wie beispielsweise Richter. Der Kreis der Personen, die das Entscheidungssystem entwickeln oder nutzen, ist offensichtlich sehr groß. Dies führt zu vielen möglichen Fehlerquellen.

6 Wo Fehler passieren können: Entscheidungssysteme im gesellschaftlichen Einsatz

Im Folgenden werden diese potenziellen Fehler anhand der Phasen des oben dargelegten Prozesses kurz skizziert, es wird die mögliche Tragweite der Fehler bewertet und untersucht, ob und wie gut diese Fehler durch regulative oder anderweitige Maßnahmen vermieden werden können.

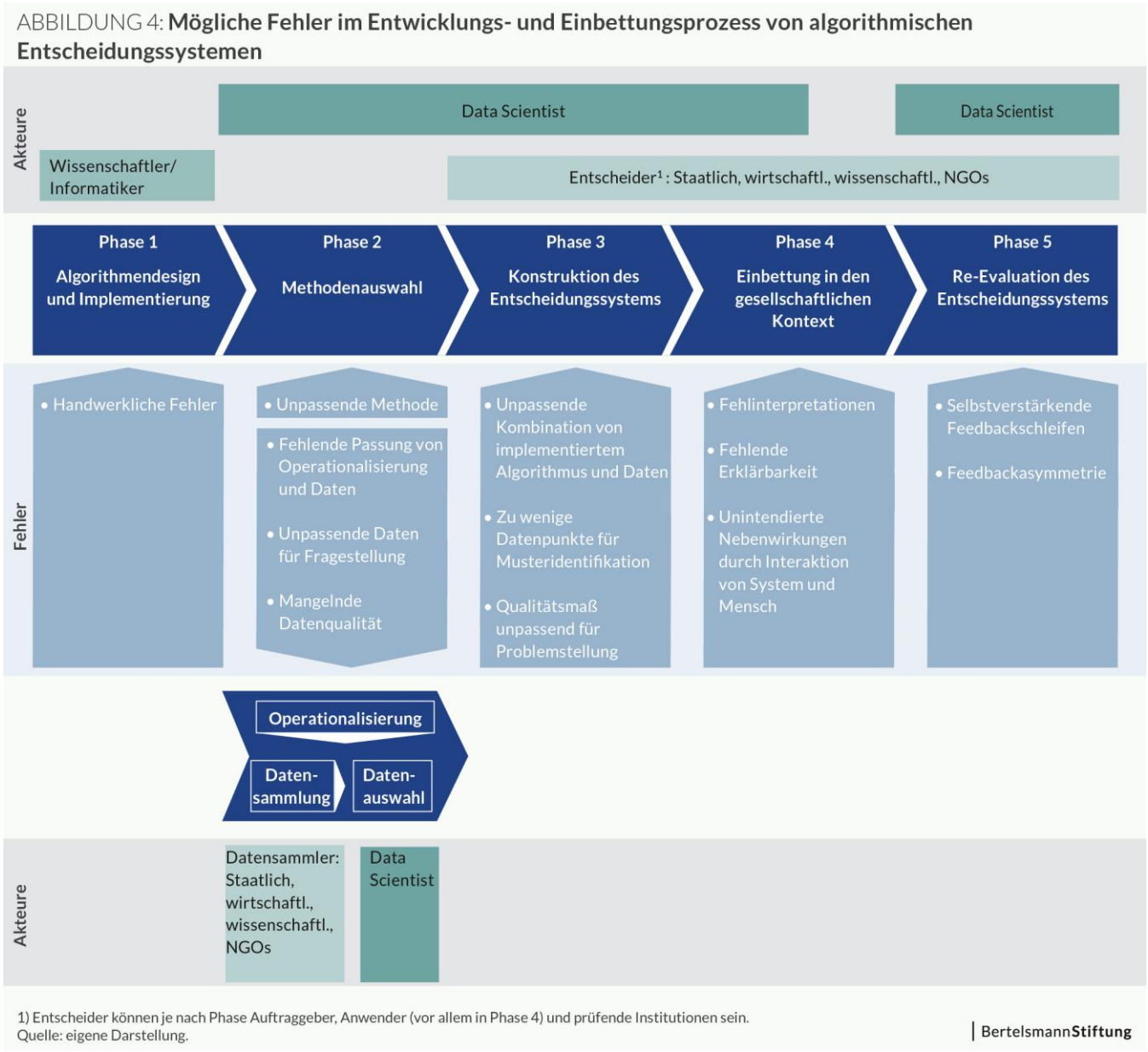


Abbildung 4. Mögliche Fehler im Entwicklungs- und Einbettungsprozess von algorithmischen Entscheidungssystemen (Quelle: eigene Darstellung)

Fehler der Phase 1: Algorithmendesign und Implementierung

Im Design und in der Implementierung von Algorithmen können verschiedene handwerkliche Fehler auftreten. Diese können von Informatikerinnen und Informatiker identifiziert, getilgt und vermieden werden, dazu sind sie

bestens ausgebildet. Die Möglichkeit, handwerkliche Fehler aufzufinden, hängt aber wesentlich von drei Aspekten ab:

1. **Nutzerbasis**: Von wie vielen Personen kann der Algorithmus verwendet werden? Als Faustregel gilt: Je mehr Anwender es gibt, desto wahrscheinlicher ist es, dass ein Fehler entdeckt wird – wenn die Spezifikation bekannt ist.
2. **Spezifikation**: Wie gut wurde das Verhalten des Algorithmus spezifiziert? Um Fehler erkennen zu können, ist es vor allen Dingen wichtig zu wissen, wie der Algorithmus in welchem Fall reagieren sollte – die Problemspezifikation muss also bekannt sein. Um unerwünschtes Verhalten zu erkennen, muss erwünschtes Verhalten klar dargelegt sein.
3. **Zugänglichkeit**: Ist der Sourcecode öffentlich zugänglich? Als Faustregel gilt: Je mehr Personen Zugang zum Code haben, desto wahrscheinlicher ist es, dass einem von ihnen ein Fehler auffällt.

Es ist wichtig zu bemerken, dass die zugrunde liegenden Algorithmen, mit denen ein Modell trainiert wird, relativ einfach zu spezifizieren sind. In vielen Fällen sind sie bereits seit Jahren in Softwarepackages professionell implementiert und oft ist auch der Code öffentlich zugänglich. Damit sind – bei Verwendung dieser „klassischen Algorithmen" – nicht viele Fehler auf der handwerklichen Ebene zu erwarten, sie können aber auch nicht ganz ausgeschlossen werden.

Für Algorithmen, deren Spezifikation unklar oder unvollständig ist, deren Implementierungscode nicht einsehbar ist und die nicht systematisch und von vielen Anwendern testbar sind, ist es wahrscheinlicher, dass handwerkliche Fehler in der Implementierung verbleiben. Solche Fehler können dazu führen, dass Systeme völlig versagen oder in einigen oder allen Fällen falsche Entscheidungen berechnen.

Fehler der Phase 2: Operationalisierung sowie Daten- und Methodenauswahl

Da die zugrunde liegenden Algorithmen im Wesentlichen leicht zu kontrollieren sind und – wenn sie zu den „klassischen Algorithmen" gehören – gut erprobt sind, werden die wichtigsten Entscheidungen bei allen Operationen getroffen, die im Einflussbereich der Data Scientists liegen. Die Mitglieder dieser Gruppe entscheiden, wie eine gesellschaftlich relevante Frage so modelliert wird, dass sie vom Computer beantwortet werden kann. Dazu müssen insbesondere an den drei folgenden Stellen Entscheidungen getroffen werden:

1. **Operationalisierung**: Wie können gesellschaftliche Prozesse messbar gemacht werden, wie zum Beispiel die Bewertung der Relevanz einer Nachricht? Die Relevanz einer Nachricht kann beispielweise daran gemessen werden, wie oft diese Nachricht anderen weitererzählt wurde.
2. **Datenauswahl**: Welche der eigentlich gewünschten Daten sind vorhanden und welche der verfügbaren Daten können sinnvoll für die Fragestellung benutzt werden? Hier muss auch darüber entschieden werden, ob die verfügbaren Daten überhaupt die notwendige Qualität aufweisen, um verwendet werden zu können.
3. **Methodenwahl**: Mit welcher Methode wird nach statistisch auffälligen Mustern in den Daten gesucht? Viele Methoden sind mit grundlegenden Annahmen über die Daten und deren Beziehung zur vorherzusagenden Eigenschaft verbunden.

Es gibt zahllose Beispiele für **Operationalisierungen**, die für das zu lösende Problem nicht sinnvoll oder geradezu schädlich sind. Ein Beispiel dafür ist die Operationalisierung der Begriffe „Wichtigkeit" oder „Zentralität" einer Person in einem sozialen Netzwerk. Dafür gibt es Dutzende von Formeln, unter denen die sogenannte „Betweenness-Zentralität" („Betweenness Centrality") besonders häufig genutzt wird (vgl. Abbildung 4). Die Betweenness-Zentralität misst für alle Paare von Personen, auf welchem Anteil der kürzesten Kommunikationswege zwischen ihnen eine Person von Interesse sitzt, und summiert diese Anteile auf. Eine Person, die sehr oft als Mittler zwischen anderen Personen tätig ist, damit eine Information von A nach B kommt, wird als zentral angesehen, da sie diese Kommunikation beeinflussen kann.

$$C_B(v) = \sum_{s,t \in V} \frac{\delta_{st}(v)}{\delta_{st}}$$

Abbildung 6: Formel für die Betweenness-Zentralität (Quelle: Zweig 2016, S. 250 ff.)[5]

Es ist wichtig zu betonen, dass dieser Index für alle Personen in einem sozialen Netzwerk ausgerechnet werden kann, auch für Netzwerke mit Millionen von Knoten. Diese harmlos aussehende Formel basiert aber auch auf mindestens drei Modellannahmen, die dazu führen, dass die Ergebnisse bei großen Netzwerken kaum sinnvoll zu interpretieren sind:

1. Der Index nimmt an, dass auch und gerade indirekte Kommunikation zwischen Menschen in sozialen Netzwerken immer den kürzesten Weg nimmt. Dies ist aber unwahrscheinlich für viele Arten der Kommunikation.
2. Ungleich wichtiger ist die Annahme, dass alle Paare von Personen mit derselben Dringlichkeit und Frequenz miteinander kommunizieren wollen – unabhängig davon, wie weit sie im Kommunikationsnetzwerk voneinander entfernt sind. Diese Annahme ist offenkundig falsch für alle größeren Kommunikationsnetzwerke.
3. Der Index nimmt an, dass eine Person eine Nachricht hintereinander an alle anderen schickt – auch wenn diese Nachricht damit mehrfach über dieselben vermittelnden Personen läuft, die die Nachricht somit schon kennen. Wahlweise kann diese Bedingung auch so interpretiert werden, dass eine Person eine personalisierte Nachricht an jeden anderen verschickt. Keine dieser beiden Annahmen ist erfüllt in einem Kommunikationsnetzwerk mit mehreren Millionen Nutzern.

Dieses Beispiel zeigt, dass insbesondere das Angebot von solchen und anderen Indizes in frei verfügbaren Softwarepackages dazu führen kann, dass Operationalisierungen mit Daten ausgerechnet werden, die auf dieser Basis nicht interpretierbar sind. Diese und andere Operationalisierungsfehler zu vermeiden, ist eine schwierige Aufgabe (vgl. Kapitel 10 bis 15 in Zweig 2016).

Zudem gibt es zahllose Beispiele von **Datenerhebungen**, die fehlerhafte Daten liefern. Es ist daher notwendig, dass elementare Qualitätsmaße für die erhobenen Daten den Entwicklern bekannt sind, bevor sie diese verwenden. Eins der Probleme bei der Datenauswahl betrifft veraltete Daten. O'Neil (2016) beschreibt in ihrem bereits erwähnten Buch den Fall von Helen Stokes, die in ein örtliches Altersheim ziehen wollte und immer wieder abgelehnt wurde. Der Grund dafür waren Verhaftungen. Diese hatten bei Auseinandersetzungen mit ihrem Mann tatsächlich stattgefunden. Da sie aber nicht verurteilt wurde, konnte sie diese aus der Regierungsdatenbank wieder entfernen lassen. Die Daten blieben jedoch in den Unterlagen des Unternehmens, das Daten für Hintergrundchecks von Mietern sammelte, weiterhin bestehen und führten zu der falschen Klassifizierung als ungeeignet für das Altersheim.

Da auch Algorithmen Modellierungsannahmen beinhalten, kann das **Zusammenbringen von** einer an sich korrekten Menge an **Daten** und einer an sich fehlerlosen **Implementierung eines Algorithmus** ebenfalls zu Fehlern führen – das heißt **die Methodenauswahl** ist falsch bezüglich der zu beantwortenden Frage.

Einfache Methoden des maschinellen Lernens sind beispielsweise sogenannte „Regressionen": Sie versuchen, eine Formel zu finden, sodass der Wert von Interesse – zum Beispiel die Frage danach, ob jemand vermutlich einen Kredit zurückzahlen wird oder wieder eine Straftat begeht – bestmöglich vorhergesagt wird. Viele der Regressionen können dabei keine Zusammenhänge zwischen den verschiedenen Faktoren für ein menschliches

[5] Die Summe läuft über alle Paare von Knoten in einem Netzwerk. Für jedes Paar wird berechnet, wie viele kürzeste Wege sie insgesamt verbinden (repräsentiert durch δ_{st}) und wie viele davon über den Knoten v verlaufen (repräsentiert durch δ_{st} (v)) (Quelle: Zweig 2016, S. 250 ff.).

Verhalten abbilden, wenn dies nicht explizit vom Data Scientist eingebaut wird. Das bedeutet, dass alle Beweggründe in ihrem Einfluss auf das menschliche Verhalten als voneinander unabhängig bewertet werden. Oft hängen diese Faktoren aber zusammen oder beeinflussen einander. Ein Beispiel dafür sind „Geschlecht" und „Drogensucht" als Faktoren: Während es im Allgemeinen so ist, dass Männer öfter kriminell werden als Frauen, könnte das Geschlecht weniger relevant sein, wenn jemand drogensüchtig ist. Daher sollte die Methode der Regression nicht bei einem Geschehen zugrunde gelegt werden, bei dem verschiedene Wirkzusammenhänge vermutet werden, wenn diese nicht explizit mit modelliert werden. Rein mathematisch gesehen kann die Methode aber verwendet werden und mindestens eines der verwendeten Vorhersagesysteme zur Bestimmung der Rückfälligkeit von Straftätern baut auf einer solchen Regression auf.

Fehler der Phase 3: Konstruktion des Entscheidungssystems

In der dritten Phase werden bei solchen Entscheidungssystemen, die auf Algorithmen des maschinellen Lernens beruhen, Algorithmus und Trainingsdaten ausgewählt.

In dieser Phase bestehen auch generellere Fehlerquellen wie die Annahme, dass überhaupt **genügend Datenpunkte** vorhanden sind, um darin statistisch signifikante Muster zu finden und daraus genügend abstrahierte Regeln abzuleiten. So wird es vermutlich nie möglich sein, einen Algorithmus zu konstruieren, der die Eignung eines Kandidaten für eine Professur auf der Grundlage seines oder ihres Lebenslaufs vorhersagen kann. Dafür sind die jeweiligen Lebensläufe zu unterschiedlich und die jeweils relevanten Journale oder Konferenzen oder Wirkungsstätten über die Jahre zu volatil, um hier aussagekräftige Muster zu extrahieren.

In der dritten Phase wird bei lernenden Algorithmen routinemäßig eine Evaluation des trainierten Systems vorgenommen. Die Evaluation eines trainierten Systems – das gelernt hat, zu klassifizieren – erfolgt, indem es auf eine bestimmte Datenmenge angesetzt wird. Bei diesem Datenset ist für alle Daten bekannt, in welche Klasse sie gehören. Solche Datensets werden „Ground Truth" genannt. An ihnen kann die Qualität der Klassifikation bewertet werden, die das trainierte System vornimmt. Am leichtesten kann das am Beispiel der Rückfälligkeitsvorhersage von Kriminellen illustriert werden. Die gängigen Algorithmen bewerten dabei alle ihnen bekannt gemachten Eigenschaften der Person, vergleichen sie mit den Eigenschaften von rückfällig gewordenen Menschen und geben eine Zahl zurück. Sie wurden so trainiert, dass sie solchen Personen, deren Eigenschaften mit denen Rückfälliger übereinstimmen, hohe Zahlen zuweisen, und solchen, bei denen weniger oder weniger wichtige Eigenschaften übereinstimmen, niedrigere Zahlen zuweisen. Damit können die Menschen nun „sortiert" werden. Aus dieser Sortierung wird eine Klassifizierung erstellt, indem man einen Schwellwert bestimmt. Alle Personen, denen der Algorithmus einen höheren Wert als diesen Schwellwert zuweist, werden in die Klasse der „vermutlich rückfällig Werdenden" eingestuft, die mit Werten darunter in die Klasse der „vermutlich nicht rückfällig Werdenden"[6]. Da bei der Ground Truth bekannt ist, ob die Personen rückfällig wurden oder nicht, kann nun die Güte der Zuordnung bewertet werden. Dazu gibt es verschiedene Bewertungsmaßstäbe. Bei der **Auswahl eines** solchen **Qualitätsmaßes zur Bewertung der Güte eines Systems** können Fehler auftauchen, wenn ein Maß gewählt wird, dessen Annahmen nicht zu der Aufgabe passen, die der Algorithmus lösen soll. In diesem Fall erscheinen Entscheidungssysteme scheinbar gut trainiert. Sie werden aber in Situationen eingesetzt, die eigentlich ein anderes Qualitätsmaß erfordern, und erzielen deshalb im konkreten Einzelfall nur mittelmäßige oder schlechte Ergebnisse. Ein Beispiel soll dies veranschaulichen. Es gibt unter anderem folgende Qualitätsmaße:

- **Sensitivität:** Dieses Maß bewertet nur, welcher Anteil einer Klasse korrekt klassifiziert wurde. Für das Beispiel bedeutet dies, dass nur geprüft wird, welcher Anteil der tatsächlich Rückfälligen auch korrekt vom Algorithmus zugeteilt wurde. Dieser Wert ist alleine für sich nicht aussagekräftig, da ein Algorithmus einfach

[6] Die Situation wird vereinfach dargestellt. Die in den USA genutzte Software COMPAS verwendet neun Schwellwerte, mit denen Personen in einer von zehn Klassen eingeteilt werden. Diese werden dann noch in drei abstrakte Klassen zusammengefasst: „hochrisiko", „mittleres Risiko" und „geringes Risiko" der Rückfälligkeit.

alle Personen in die „vermutlich rückfällig Werdenden"-Klasse einteilen kann. Damit wären alle tatsächlich Rückfälligen korrekt zugeordnet.
- **Spezifizität:** Deshalb bedarf es eines weiteren Maßes, das die korrekte Zuteilung zu der anderen Klasse (Nichtrückfällige, die tatsächlich nicht rückfällig geworden sind) prüft.
- **Akkuratheit:** Dieses Maß gibt den Anteil aller korrekt zugeteilten Personen wieder, unabhängig davon, ob sie korrekt als nicht rückfällig oder korrekt als rückfällig vorhergesagt wurden.
- **ROC AUC**[7]**:** Dieses Maß berechnet den Anteil aller Personenpaare von Rückfälligen und Nichtrückfälligen, bei denen der Algorithmus dem Rückfälligen den höheren Wert zugewiesen hat.

In den meisten Fällen wird die Situation von den jeweiligen Qualitätsmaßen unterschiedlich bewertet. Sie enthalten unterschiedliche Annahmen, die je nach Aufgabe, die gelöst werden soll, unterschiedlich passend sein können. Im Beispiel der Rückfälligkeitsvorhersage lag der ROC AUC-Wert bei 71 Prozent, die Sensitivität aber nur bei 50 Prozent. Es wurden also 71 Prozent aller Paare von Rückfälligen und Nichtrückfälligen korrekt in Relation zueinander bewertet. Aber von allen Personen, die in die Klasse der Rückfälligen eingeteilt und denen hohe Werte zugewiesen wurden (hohe Rückfallwahrscheinlichkeit), werden nur 50 Prozent rückfällig. Würde der ROC AUC-Wert als Maß für die Beurteilung des Systems herangezogen, würde das System als gut trainiert erscheinen. Tatsächlich ist das ausschlaggebendere Maß hier aber die Sensitivität. Denn eine Fehlentscheidung bei der Zuordnung zur Klasse der Rückfälligen zieht im Einzelfall enorme Konsequenzen nach sich – bedeutet sie doch, dass ein Mensch fälschlicherweise verdächtigt wird und dies wahrscheinlich eine Verlängerung der Haftstrafe nach sich zieht. Durch die Wahl des falschen Qualitätsmaßes wurde die Software falsch trainiert und erzielt nun viele fälschlich als hochrisikoreich eingeordnete Personen.

Fehler der Phase 4: Einbettung des Systems in den gesellschaftlichen Kontext

In der vierten Phase wird das Entscheidungssystem in den gesellschaftlichen Kontext eingebettet. Die Daten, auf deren Grundlage eine Bewertung oder Vorhersage berechnet wird, werden nun häufig von den Nutzern des Systems selbst eingegeben, die auch oftmals die Interpretation der Resultate vornehmen.

Die Algorithmen des maschinellen Lernens haben aus den Trainingsdaten gelernt, dass in der Vergangenheit bestimmte Eigenschaften von Personen mit dem Verhalten von Interesse korrelieren, zum Beispiel: Eine Person, die schon mehrfach vorbestraft ist, wird vermutlich wieder kriminell werden. Es kann aber auch zu weniger einsichtigen Korrelationen kommen, wie oben am Beispiel der Distanz vom Arbeitsplatz und der Kündigungswahrscheinlichkeit schon erwähnt. In jedem Fall werden die Personen, deren Daten eingegeben werden, implizit einer Gruppe von Personen in den Trainingsdaten zugeordnet, die laut Algorithmus „ähnlich" zu ihnen sind. Das Verhalten der Menschen in dieser Gruppe – so wie es vom Algorithmus interpretiert wird – bestimmt die Entscheidung darüber, wie das System das Verhalten der neuen Personen bewertet oder vorhersagt. Wenn den Nutzern des Systems nicht klar ist, was eine Vorhersage eigentlich ist, nämlich eine gruppenbasierte Wahrscheinlichkeit für ein bestimmtes Verhalten, kann es also zu massiven **Fehlinterpretationen** kommen. Denn eine „Rückfälligkeitsvorhersage von 60 Prozent" bedeutet, dass die zu bewertende Person einer Personengruppe zugeordnet wurde, von denen 60 Prozent wieder kriminell wurden. Dieser gruppenbasierte Wert wird dann als das individuelle Risiko interpretiert – wobei natürlich jede Einzelne entweder wieder rückfällig wird oder nicht, aber nicht zu 60% rückfällig wird. Eine solche Bewertung kann sinnvoll sein, wenn knappe Rehabilitionsmaßnahmen an die Personen verteilt werden sollen, die am stärksten gefährdet sind. Sie sind allerdings kaum interpretierbar, wenn es zum Beispiel um einen Antrag auf vorzeitige Haftentlassung gibt.

Wenn die Fehlerrate der zugrunde liegenden Daten der Nutzerin nicht bekannt ist, sind ebenfalls Fehlinterpretationen der resultierenden Bewertung (z. B. der Leistung) einer Person möglich. Dies geschieht beispielsweise momentan häufig bei der Interpretation des oben genannten h-Indexes, der eben nicht nur von der Leistung der

[7] ROC: Receiver Operating Characteristics; AUC: Area Under Curve.

Person abhängt, sondern ganz massiv von der verwendeten Datenbank und einem algorithmischen Teilsystem, das versucht zu erkennen, ob zwei in der Datenbank gespeicherte Namen dieselbe Person oder unterschiedliche Personen bezeichnen („Entity Recognition Problem"). Wenn die zugrunde liegende Datenbank fehlerhaft ist oder Namensänderungen einer Person (z. B. durch Heirat oder Scheidung) dem System nicht bekannt sind, kann das ebenfalls häufig fehlerhafte Resultate und damit fehlerhafte Interpretationen der Resultate nach sich ziehen. Diese Entity Recognition Probleme werden, genauso wie andere Fälle, bei denen unvollständige oder falsche Daten die Interpretation der Ergebnisse erschweren, auch in O'Neils Buch „Weapons of Math Destruction" (2016) eingehend und anschaulich beschrieben.

Die Aufdeckung falscher Resultate oder Interpretationen wird grundlegend erschwert, wenn das Entscheidungssystem keine für den Menschen einsichtige Erklärung für sein Ergebnis liefern kann (**Erklärbarkeit**). Natürlich sind auch die trainierten Modelle im Wesentlichen Algorithmen, die für jede Eingabe deterministisch eine Ausgabe produzieren. Für die momentan beliebten neuronalen Netze würden diese aus einer Reihe von hintereinander geschalteten Gleichungen bestehen, wobei das Ergebnis der ersten Reihe in die zweite Reihe eingeht und so weiter. Damit ist eindeutig nachvollziehbar, ob sich das System verrechnet hat – aber nicht, ob beispielsweise einem Kunden mit diesem Resultat der angefragte Kredit verweigert werden sollte. Dies erlaubt es dem Kunden zum Beispiel, weder sich mit dem Wert anderer Personen in einer ähnlichen Situation zu vergleichen noch die Interpretation der Zahl als „nicht kreditwürdig" ganz allgemein infrage zu stellen. Nicht zuletzt ist ein solches Gleichungssystem nicht geeignet, um dem Bankkunden zu erklären, was er in seinem Leben ändern muss, um das nächste Mal einen Kredit zu bekommen. Eine solche Erklärung ist nicht „actionable", ermöglicht also keine gezielte Verbesserung.

In diese vierte Phase fallen auch Effekte, die erst durch die **Interaktion des Menschen mit dem Entscheidungssystem** entstehen. Diese heißen in der Komplexitätsforschung „emergente Phänomene". Dazu zählen beispielsweise persönlichkeitsrechtlich relevante Ergänzungen in der Suchvervollständigung. Ein prominentes Beispiel war die Klage von Präsidentengattin Bettina Wulff gegen Google: Wenn man nach ihr auf Google suchte, wurde die Anfrage vervollständigt mit Begriffen wie „Rotlicht" oder „Escort". Auch wenn die genauen Mechanismen der damaligen Suchvervollständigung bis heute unbekannt sind, war doch bekannt, dass Suchbegriffe generell mit den Worten vervollständigt wurden, die zu dem Suchzeitpunkt besonders häufig gemeinsam mit dem schon Getippten gesucht wurden. Anscheinend zählte der Algorithmus aber die Suchanfragen, die überhaupt erst durch die Vervollständigung getriggert wurden, genauso mit bei der Berechnung der Häufigkeit wie solche, die der Nutzer vollständig händisch eingetragen hatte. Ein solches Vorgehen ist sinnvoll bei technischen Fragen, beispielsweise bei der Suche nach Funktionen von Software, etwa Outlook oder Word. Hier kann die Tatsache, dass viele Personen nach der Lösung eines Problems suchen, auch darauf hinweisen, dass tatsächlich viele Nutzer dasselbe Problem haben – unabhängig davon, ob sie die gesamte Anfrage händisch eingeben oder auf den Vorschlag der Vervollständigung eingehen. Diese Interpretation ist aber nicht sinnvoll bei Vervollständigungen, die einen skandalhaften Charakter haben. Hier ist es wahrscheinlich, dass Suchende, die überraschend mit einer solchen Vervollständigung konfrontiert werden, auf die Suchanfrage klicken, ohne sie vorher im Sinn gehabt zu haben. Daher ist die reine Häufigkeit von Suchanfragen kein Gradmesser dafür, dass die Suchenden diese Suchanfrage für relevant oder gar richtig halten. Damit bedeutet die reine Popularität einer Suchanfrage unterschiedliche Dinge für unterschiedliche Bereiche: Im technisch-faktischen Bereich kann sie als Gradmesser für das allgemeine Interesse gelten, was wiederum darauf hindeutet, dass viele Menschen diese Frage für wichtig und relevant halten. Bei skandalösen Inhalten sollte der Algorithmus die Popularität dagegen eher unterschätzen, um keine selbsterfüllende Prophezeiung zu generieren.

Dies ist nur ein Beispiel für die überraschende Nebenwirkung eines eigentlich sinnvoll gestalteten Algorithmus. Ein drastisches Beispiel für nicht intendierte Wirkungen ist auch Chatbot Tay, ein weiblicher Avatar, der lernen sollte, worüber sich Menschen in einem Forum unterhalten, um dann passende Textschnipsel aus dem Internet zu suchen, die sinnvoll in die Diskussion eingebracht werden können. Den Menschen in dem Forum, in dem Tay ihre Fähigkeiten beweisen sollte, gefiel es aber, sie mit rechtsradikalen Äußerungen zu füttern. Dementsprechend lernte sie diese Stichpunkte und beteiligte sich mit so extremen Statements, dass ihre Entwickler sie aus dem Verkehr zogen (Beuth 2016). Auch hier zeigte ein technisch eigentlich hervorragender Algorithmus erst in der Interaktion

mit menschlichen Nutzern einen unbeabsichtigten Effekt, von dem unklar ist, wer eigentlich für ihn verantwortlich ist und wie er sich verhindern ließe.

Nicht zuletzt entstehen in dieser Phase Fehler durch nicht intendierte Wirkungen, die durch den **nicht sachgemäßen und/oder kriminell-manipulativen Umgang** mit den Algorithmen entstehen. Besonders spannend ist das Beispiel mazedonischer Jugendlicher, die laut einer Analyse von BuzzFeed beim US-Präsidentschaftswahlkampf im Sommer 2016 eine Rolle spielten. Laut den Autoren dieser Studie, Silverman und Alexander (2016), nahmen die Jugendlichen Nachrichten aus dem Netz, sensationalisierten sie und veröffentlichten sie auf ihrer Facebookseite, um damit Besucher auf ihre eigenen Webseiten zu locken. Auf diesen Webseiten schalteten sie Werbung und verdienten damit – laut Aussage der Studie – mehrere Zehntausend Dollar. Ein Markt, in dem es finanziell attraktiv ist, Fake News zu veröffentlichen, kommt in diesem Beispiel erst durch verschiedene, zusammenwirkende Faktoren zustande: Zum einen ist das die fehlende ethische Einstellung der Jugendlichen, die es ihnen erlaubt, Falschmeldungen zu ihren Gunsten zu nutzen. Zum anderen tragen Algorithmen dazu bei: Der Algorithmus auf sozialen Netzwerken, der viel Aufmerksamkeit für emotionalisierte Nachrichten erzeugt, und der Algorithmus, der bei der Verteilung von Werbung eher weniger auf die Eigenschaften der Webseiten – wie etwa ihre Qualität – achtet, sondern vielmehr auf die demographischen Eigenschaften der Nutzer fokussiert. Diese Effekte entstehen erst durch die Interaktion eines einzelnen Entscheidungssystems mit seinen Nutzern oder gar durch die Interaktion mehrerer Entscheidungssysteme.

Solche emergenten Phänomene sind besonders schwer vorherzusagen und bedürfen daher eines agilen Prozesses, der schnell auf schwerwiegende Folgen des Einsatzes von algorithmischen Entscheidungen reagieren kann. Solche Folgen können zum Beispiel Verhaltensanpassungen von Anwendern und Betroffenen an Entscheidungssysteme sein. Diese könnten im Beispiel der Rückfallwahrscheinlichkeitsprognose wie folgt aussehen: Auf der einen Seite könnten Richter auch dann der Empfehlung des Algorithmus zu einer Haftstrafe folgen, wenn sie nicht mit ihrer eigenen Entscheidung übereinstimmt. Denn die negativen Konsequenzen bei einer Fehlentscheidung, die gegen den Algorithmus getroffen wurde, überwiegen den persönlichen Nutzen für den Richter bei einer richtigen Entscheidung entgegen der Empfehlung des Algorithmus. Um an dieser Stelle gegenzusteuern, bräuchte es einen Prozess, der es Richtern erlaubt, ohne negative Konsequenzen auch entgegen der Empfehlung des Algorithmus eine Entscheidung zu treffen. Auf der anderen Seite könnten sich auch Kriminelle an das Entscheidungssystem anpassen. Sie können Tipps austauschen, mit welchen Antworten im Fragebogen man auf eine geringe Punktzahl kommt und dementsprechend als wenig rückfallgefährdet eingestuft wird. Derartige emergente Effekte müssen während des Einsatzes von Entscheidungssystemen erkannt, beobachtet und durch Änderungen an den Systemen gesteuert werden.

Fehler der Phase 5: Re-Evaluation des Entscheidungssystems

Manche Algorithmen erhalten laufend Echtzeitfeedback und können so verbessert werden. Ein Beispiel dafür sind Empfehlungssysteme, die Kunden im Onlinehandel auf Basis der bisher gekauften Waren neue Produkte vorschlagen. Sie erhalten sofort Rückmeldung darüber, ob der Kunde, das vorgeschlagene Produkt angesehen oder gekauft hat und können ihre Empfehlungen daraufhin anpassen. Das Feedback, das ein Algorithmus erhält, kann jedoch auch negative Auswirkungen haben. So kann es zu **selbstverstärkenden Feedbackschleifen** kommen**.** O'Neil (2016) führt dies am Beispiel von Predictive Policing aus. Es kann dazu führen, dass in manchen Arealen mehr Streife gefahren wird. Dies führt wiederum automatisch zu mehr Festnahmen in diesem Gebiet, weil auch mehr Kleinkriminalität entdeckt wird. Das wiederum hat zur Folge, dass das System „lernt", dass hier viele Kriminelle leben, was die Anzahl der Streifen weiter erhöhen könnte. Hier kommt es zu einer scheinbar objektiven Maßnahme, die die tatsächlich stattfindende Kriminalität jedoch höchst ungleichmäßig verfolgt und daher den Anschein erweckt, dass eine Teilgruppe der Bevölkerung viel krimineller ist als der Rest der Bevölkerung. Auf diese Weise kann der Einsatz eines Entscheidungssystems zu mehr Ungleichheit führen.

Eine weitere Fehlerquelle der Reevaluation eines Entscheidungssystems ist die **Feedbackasymmetrie**. Sie beschreibt das Problem, dass viele Situationen, in denen lernende Algorithmen eingesetzt werden, nur ein Feedback in eine Richtung zulassen. Ein Beispiel dafür sind Algorithmen, die die Kreditwürdigkeit einer Person berechnen. Wenn dies dazu führt, dass eine Person einen Kredit bekommt, diese Person nachher aber den Kredit nicht zurückzahlt, kann der Algorithmus darüber informiert werden. Wenn aber eine Person keinen Kredit bekommt, die ihn zurückgezahlt hätte, kann der Algorithmus über diese Fehlentscheidung nicht informiert werden, da die Person keine Möglichkeit hatte, dies nachzuweisen. Dasselbe Prinzip gilt für eine große Anzahl von Situationen, angefangen bei Entscheidungen über Haftstrafen statt Bewährungen, Einladung zu Jobinterviews und Jobangebote, Studienplatzvergabe und – im Extrem – die Identifikation von Terroristen mit der sofortigen Erschießung von Personen, die scheinbar vom System identifiziert wurden. Natürlich gibt es solche Feedbackasymmetrien auch bei Urteilen durch menschliche Experten, aber beim Einsatz von Algorithmen ist es wahrscheinlicher, dass diese sich monopolartig durchsetzen. Dadurch werden die Vorurteile, die in einem System möglicherweise enthalten sind, vervielfältig. Dies kann dazu führen, dass bestimmte Personen kategorisch ausgeschlossen werden. Wird überall derselbe Algorithmus eingesetzt, bleibt zum Beispiel einem einmal abgelehnten Bewerber nicht nur die Chance auf eine bestimmte Arbeitsstelle, sondern gleich der Zugang zum gesamten Arbeitsmarkt verwehrt.

Weitere Fehlerquellen

Neben den Mängeln im Entwicklungs- und Einbettungsprozess von Entscheidungssystemen, die in den einzelnen Phasen auftauchen, gibt es noch weitere übergeordnete Fehler:

Imbalance: Die unterschiedliche Größe der verschiedenen Klassen, in die Menschen kategorisiert werden sollen. Ein extremes Beispiel bieten die Klassen „möglicher Terrorist“ und „kein Terrorist“. Die Klasse der möglichen Terroristen ist in Bezug auf die Gesamtbevölkerung in allen Ländern dieser Welt sehr klein (wenn auch in der genauen Relation zur Bevölkerung schwer quantifizierbar). In Deutschland spricht man beispielsweise von circa 550 Gefährdern und weiteren 1100 Personen mit „islamistisch-terroristischem Personenpotenzial“ auf circa 80 Millionen Einwohner. Es handelt sich also um einen Anteil von ca. 0,002 Prozent (1650 auf 80.000.000). Je größer das Ungleichgewicht ist, desto schwerer tun sich Entscheidungssysteme damit, Regeln zu lernen, die zuverlässig zwischen den beiden Klassen trennen.

Absolute Häufigkeit: Fehlprognosen können zudem entstehen, wenn die absolute Häufigkeit, mit der ein zu klassifizierendes Ereignis auftritt, gering ist. Wenn es zum Beispiel zwar grundsätzlich ausreichend viele Datenpunkte gibt, diese sich aber über einen langen Zeitraum strecken, in dem sich andere relevante Parameter geändert haben, sollte kein Algorithmus aus diesen Daten lernen. Ein Beispiel dafür ist die Menge aller Bundesminister der Bundesrepublik Deutschland. Diese Menge umfasst rund 185 Personen, die von 1949 bis heute unter sehr unterschiedlichen und individuellen Bedingungen ernannt wurden. Ein Algorithmus könnte hier vermutlich nichts „lernen“, was über allgemeine Führungsqualitäten hinausgeht, sodass eine auf diesen Personen und ihren Eigenschaften beruhende Vorhersage, wer in Zukunft Minister werden wird, nicht erfolgreich sein dürfte.

Fehler sind, wie die Ausführungen dieses Kapitels zeigen, in allen Phasen des Entwicklungs- und Entstehungsprozesses möglich. Sie können Konsequenzen von unterschiedlicher Tragweite nach sich ziehen. Manche der Fehlerquellen können zudem einfacher entdeckt und behoben werden als andere. Im folgenden Kapitel werden abschließend erste Lösungsansätze beispielhaft skizziert, mit denen in allen Phasen einige der Fehler angegangen werden können.

7 Wo man ansetzen kann: Beispielhafte Lösungsvorschläge

In ihrem bereits zitierten Buch „Weapons of Math Destruction“ (etwa: „Mathevernichtungswaffen“) beschäftigt sich O’Neil mit der Auswirkung von Entscheidungsunterstützungssystemen und AuDM-Systemen, die die folgenden drei Eigenschaften haben: Sie sind intransparent, sie können ohne großen Mehraufwand auf viele Menschen angewendet werden (sie „skalieren“) und sie haben individuell großes Schadenspotenzial. Dabei besteht der Schaden grundsätzlich immer darin, dass eine Person oder ihr Verhalten durch das Entscheidungssystem falsch bewertet wird und ihr dadurch wesentliche Lebenschancen verwehrt bleiben oder dass gesellschaftliche Teilhabe reduziert wird. Intransparenz, leichte Skalierbarkeit und potenzieller persönlicher Schaden sind also wichtige Anhaltspunkte dafür, wie notwendig Qualitätskontrollen und eine Regulierung eines Entscheidungssystems sind.

Für die Kontrolle von Entscheidungssystemen gibt es jedoch nicht die eine Lösung, sondern vielmehr unterschiedliche Ansätze, die in den verschiedenen Phasen im Entwicklungs- und Einbettungsprozess wirksam werden können. Im Folgenden werden exemplarisch Lösungsvorschläge dargestellt. Sie sollen verdeutlichen, dass es für die meisten Fehler in allen Phasen des Prozesses Ansätze gibt, mit denen sie sich vermeiden oder beheben lassen. Manche dieser Lösungsansätze sind auf mehrere Phasen anwendbar, andere adressieren gezielt Fehlerquellen in bestimmten Phasen.

Phasen 1 bis 5: Algorithmen-TÜV

Mit dem Schlagwort „Algorithmen-TÜV“ ist die Idee einer unabhängigen, demokratisch legitimierten Institution gemeint – der Vorschlag stammt von Mayer-Schönberger und Cukier (2013). Im Auftrag des Algorithmen-TÜVs prüfen Experten Entscheidungssysteme auf einen angemessen und korrekten Einsatz. Dabei könnten sie für mehrere Phasen eingesetzt werden, je nachdem, wo Probleme auftreten.

Eingriffe eines Algorithmen-TÜVs sollten nach dem Wirkpotenzial von Entscheidungssystemen abgestuft werden. Die Forderung nach einer transparenteren Entwicklung, Anwendung und Kontrolle von Entscheidungssystemen gilt hier vor allem für solche mit hohem Einflusspotenzial auf Teilhabe. Sie bezieht sich nicht auf die oft geforderte allgemeine Transparenz, unter der einige die Offenlegung des Programmcodes verstehen. Diese Forderung basiert auf dem Missverständnis, dass der Code alleine ausreichend und geeignet sei, um ein Urteil über die Wirkung eines Algorithmus fällen zu können. Dies ist aus mehreren Gründen nicht richtig:

1) Programmcode ist keine effiziente Kommunikationsform zwischen Menschen, sondern eine Kommunikationsform zwischen Mensch und Maschine, die einen Kompromiss zwischen den Bedürfnissen von Computern und Maschinen darstellt. Durch in den Programmcode eingebettete Kommentare in menschlicher Sprache werden zum Beispiel Zusammenfassungen der Funktion von nachfolgenden Codeteilen geliefert und besondere Kniffe erklärt. Weiterhin gilt die Verwendung sogenannter „sprechender Variablennamen“ als guter Programmierstil, die eine essenzielle Hilfe für das menschliche Verständnis des Codes darstellen. Eine Firma, die sich gegen die Folgen einer Zwangsveröffentlichung des Codes wehren will, wird jegliche Kommentare streichen und die Variablennamen durch Nonsenszeichenfolgen ersetzen. Ohne Übertreibung vertausendfacht sich damit der notwendige Arbeitsaufwand für das Verständnis und macht die gewünschte Wirkung zunichte. Eine erzwungene Veröffentlichung des Programmcodes wird daher sehr wahrscheinlich nicht den gewünschten Effekt haben.
2) Wesentlich wichtiger ist, dass es sehr oft auch gesellschaftlich gute Gründe gibt, um die Mechanismen eines Entscheidungssystems nicht transparent zu machen. Da Kriminelle etwa den Aufwand zum Verständnis des Codes bei genügend hoher Gewinnerwartung nicht scheuen, können diese manipulierend eingreifen oder wichtige Technologien kopieren. Es ist sogar denkbar, dass eine Gesellschaft das Recht auf Intransparenz eines Algorithmus haben könnte, um gesellschaftlich folgenschwere Manipulationen zu erschweren. Dies könnte beispielsweise heißen, dass die Hilfeseiten

von Google, die den Suchmaschinenalgorithmus beschreiben, teilweise gelöscht werden müssten, um die prominente Platzierung von Fake News zu behindern.

3) Algorithmen sind nur eine Komponente von Entscheidungssystemen: Der Algorithmus definiert nämlich nur, wie die Zwischenstruktur aufgebaut werden muss, die in den Daten vorliegende Muster speichert und zur Bewertung oder Vorhersage genutzt werden kann. Ohne die Daten, mit denen die Zwischenstruktur „trainiert" wurde, kann also nicht abschließend bewertet werden, ob das Gesamtsystem den gewünschten Zielen entspricht.

Phasen 2 bis 5: Berufsethik für den Beruf des Data Scienstist

Für alle Phasen, an denen Data Scientists beteiligt sind, ist die Erstellung eines Berufsprofils und einer **Berufsethik für den Beruf des Data Scientist** ein wichtiger Ansatz, um Fehler zu vermeiden. Data Scientists stellen die wichtigste Akteursgruppe dar. Doch bisher sind die Berufswege eher erratisch und klassische Ausbildungen als Physiker, Informatiker oder Mathematiker beinhalten im Allgemeinen weder eine interdisziplinäre Grundausbildung noch die Berufsethik, die für diese besonderen Entscheidungssysteme notwendig wäre. Da Entscheidungssysteme weitreichende gesellschaftliche Auswirkungen haben können, sollten ihre ethischen und sozialen Implikationen Teil des Curriculums in der Ausbildung zum Data Scientist werden. Eine Professionsethik für Entwickler und Data Scientists müsste Prinzipien beinhalten, die auf einen ethischen, sicheren, nützlichen und nachvollziehbaren Einsatz von Entscheidungssystemen abzielen. Erste Ansätze dazu gibt es in den USA zum Beispiel mit den „Asilomar AI Principles". In Deutschland werden mit dem Studiengang „Sozioinformatik" an der TU Kaiserslautern Studierende auf der einen Seite allgemeiner dazu ausgebildet, die möglichen gesellschaftlichen Folgen von Softwaresystemen zu modellieren und zu antizipieren. Auf der anderen Seite lernen sie auch sinnvolle Softwaresysteme für gesellschaftliche Kontexte zu entwickeln. Zudem gibt es erste Studiengänge zum Thema „Data Science", deren Curriculum aber noch nicht standardisiert ist.

Phase 2: Monitoring des Inputs

Monitoring beschreibt in diesem Fall die Überprüfung der Trainingsdaten (Input). Wird beispielsweise ein Algorithmus mit den Bewerberdaten der letzten zehn Jahre für Jobs in einer Firma trainiert, kann es passieren, dass diese Daten schon Diskriminierungen enthalten. Diese würden dann mitgelernt werden. Ein wichtiger Schritt in dieser Phase ist also die Überprüfung der Trainingsdaten auf Vollständigkeit, Diskriminierungsfreiheit und Korrektheit.[8]

Phase 3: Black-Box-Experimente

Unter Black-Box-Experimenten versteht man das systematische Austesten der Funktionalität eines Algorithmus durch Eingabe von Daten, bei denen das korrekte Ergebnis des Algorithmus bekannt ist. Bei Entscheidungssystemen, die eine Bewertung abgeben, ist also die korrekte Bewertung bekannt, bei Vorhersagealgorithmen ist bekannt, welchen Ausgang die Situation nahm, von der die Daten stammten. Black-Box-Experimente können damit Hinweise auf Fehler bei der Konstruktion des Entscheidungssystems (vgl. Kapitel 5, Phase 3) geben.

So hat ProPublica beispielsweise den Vorhersagealgorithmus COMPAS der Firma Northpointe Inc. mit Daten von Kriminellen untersucht, von denen bekannt war, dass sie in den zwei Jahren nach ihrer Entlassung eine weitere Straftat begangen hatten oder dass sie in diesem Zeitraum nicht rückfällig geworden waren (Angwin et al. 2016).

[8] Eine allgemeine Sichtung nach datenschutzrechtlichen Gesichtspunkten ist natürlich auch relevant, liegt aber außerhalb des Fokus dieser Studie.

COMPAS weist Straftätern einen Wert zwischen 1 und 10 zu, wobei eine höhere Zahl mit einer erhöhten Wahrscheinlichkeit für die Rückfälligkeit einhergeht. ProPublica wies für dieses Datenset nach, dass zwar grundsätzlich die Wahrscheinlichkeit der Gruppen, rückfällig zu werden, mit dem Score stieg, aber dass insgesamt die Rückfälligkeitswahrscheinlichkeit auch in den Gruppen mit hohem Score relativ niedrig war. Beispielsweise geht eine Klassifizierung in die Klasse 8 („Hochrisiko“) damit einher, dass 60 Prozent der Personen im Trainingsdatenset hier wieder rückfällig wurden. Es erscheint fragwürdig, ob ein solcher Wert schon dafür ausreichend ist, einen Kriminellen als „mit hohem Risiko rückfällig werdend“ zu bezeichnen. ProPublica wies auch darauf hin, dass Afroamerikaner fälschlich zu oft in die Hochrisikogruppen klassifiziert wurden und Weiße zu wenig oft.

Es sind jedoch nicht alle (proprietären) Algorithmen auf diese Weise überprüfbar. So ist es momentan beispielsweise nicht möglich, die Filterblasentheorie von Pariser (2012) im Newsfeed von Facebook durch automatische und große Black-Box-Experimente zu untersuchen. Da es sich dabei um gesellschaftliche Auswirkungen von Algorithmen handelt, sollte es unabhängiger Forschung möglich gemacht werden, sie anhand von Black-Box-Experimenten zu untersuchen und zu überprüfen (vgl. unten Phase 5).

Phase 4: „Beipackzettel“ für Algorithmen

In einem „Beipackzettel“ für Algorithmen soll analog zu Beipackzetteln für Medizinprodukte beschrieben werden, um was für einen Algorithmus es sich innerhalb des ADM-Systems handelt, wie und wann das System angewendet werden kann und welche Nebenwirkungen es haben kann: Neben den Inputparametern und der allgemeinen Datengrundlage würde in einem Beipackzettel beispielsweise das mathematische Problem erklärt, das der Algorithmus lösen soll. Dazu müsste erläutert werden, wie der gesellschaftlich relevante Prozess auf das mathematische Problem reduziert wurde und welche Modellannahmen dazu getroffen wurden. Weitere Modellannahmen, wie zum Beispiel die verwendete Datenanalysemethode, müssten ebenfalls erläutert werden. Bekannte Nebenwirkungen des Algorithmus (z. B: „Kann die Popularität von sensationalistischen Nachrichten weit über den Wahrheitsgehalt hinaus erhöhen“) sollten kontinuierlich gesammelt werden und dem Beipackzettel hinzugefügt werden.

Ein solcher Beipackzettel kann für verschiedene Zielgruppen hilfreich sein. Er kann die Institutionen, die das Entscheidungssystem einsetzen, die korrekte Anwendung des Systems und die Interpretation der Ergebnisse erleichtern. Zudem kann er Personen, die durch das Entscheidungssystem bewertet werden, dabei helfen, die Bewertung nachzuvollziehen und das Resultat zu interpretieren. Er kann Forschern und Data Scientists unterstützen, den Algorithmus zu überprüfen (vgl. Phase 5).

Damit kann ein Beipackzettel Fehler in der vierten Phase, in der Entscheidungssysteme angewendet sowie ihre Ergebnisse interpretiert und für Aktionen genutzt werden, vermeiden helfen. Dazu können darüber hinaus auch **einheitliche Trainings für Anwender** eines Entscheidungssystems beitragen, die Daten in das System einpflegen oder auf der Grundlage des Systems eine Entscheidung treffen sollen. Sie sind insbesondere für solche Berufsgruppen notwendig, die in der Regel keine verpflichtende statistische Grundausbildung erfahren haben. Für diese Trainings wären klare Richtlinien wünschenswert, wie diese aussehen müssen.

Phasen 1 bis 5: Validierung und externe Beforschbarkeit

Ein Beipackzettel für Entscheidungssysteme sollte auch Auskunft über die **Angemessenheit** des Systems machen und Belege dafür anbringen, inwiefern es zu besseren Lösungen führt als menschliche Experten. Eine solche Bewertung gehört in die fünfte Phase der Reevaluation, die auch die Auswirkungen des Einsatzes der Software in der Gesellschaft mit einbezieht. Die Bewertung von Entscheidungssystemen ist momentan jedoch völlig unabhängig von ihrer Einbettung in gesellschaftliche Prozesse und von der Evaluation des durch seinen Einsatz entstehenden

sozioinformatischen Systems. Insbesondere wenn Algorithmen menschliche Experten in gesellschaftlich relevanten Prozessen ersetzen, muss ein experimentell überprüfbarer Beweis vorliegen, dass die algorithmisch unterstützte Lösung den gesellschaftlich relevanten Prozess verbessert. Es kann nicht genügend betont werden, dass es dazu nicht ausreichend ist, nur die Qualität der reinen Bewertung oder Vorhersage zu kennen. Das Entscheidungssystem ist Teil eines komplexen soziotechnischen Gesamtgefüges und setzt darin neue Anreize und verändert die Kommunikation innerhalb des sozialen Subsystems. Damit kann es zu emergenten Phänomenen kommen, die die Gesamteffektivität steigern oder verringern. Im oben genannten Beispiel der Rückfälligkeitsvorhersage von Angeklagten könnten sich Richter zum Beispiel übermäßig dazu verpflichtet fühlen, den Vorschlägen des Systems zu folgen. Denn eine Fehlentscheidung, die durch das System gestützt wird, hat für sie persönlich weniger Konsequenzen als eine Fehlentscheidung, die von der Entscheidung des Systems abweicht. Daher ist es notwendig, die Qualität des Gesamtprozesses im Vergleich zur Qualität des Prozesses ohne algorithmisches Entscheidungssystem zu bewerten.

Unter Umständen ist es auch nötig, dass weitere Gruppen das ADM-System (alleine oder in seiner Einbettung) validieren. **Externe Beforschbarkeit** beschreibt daher die Möglichkeit, dass Dritte Zugang zu Algorithmen und Trainingsdaten bekommen und diese zu Forschungszwecken nutzen können. Fehler aller Phasen, darunter auch solche der fünften Phase, wie etwa selbstverstärkende Feedbackschleifen, können nur so aufgedeckt werden. Ein Beispiel für solche selbstverstärkenden Feedbackschleifen sind die viel erwähnten Filterblasen auf Facebook: Nutzer klicken Informationen an, die ihren Interessen und Einstellungen entsprechen, der Algorithmus zeigt ihnen mehr davon an und die Nutzer wiederum interagieren mit diesen Informationen. Das soll laut der Filterblasentheorie zur Folge haben, dass Nutzer nur noch die Inhalte sehen, die ihren Einstellungen entsprechen und keine Informationen erhalten, die außerhalb ihres Horizonts liegen. Eine Möglichkeit, die Filterblasentheorie anhand von Facebookdaten zu überprüfen, wäre, systematisch Fake User zu erstellen, deren Verhalten von Social Bots gesteuert wird, um die Reaktion des Newsfeedalgorithmus darauf zu testen. Bisher ist es nicht möglich, dieses Phänomen in externen Forschungsprojekten zu untersuchen, da es verboten ist, Fake Accounts anzulegen, und Forscher keinen selektiven Zugriff auf Daten von Facebook erhalten.

Ein weiterer Fehler der fünften Phase liegt in der oben genannten Feedbackasymmetrie. Damit wurde das Problem beschrieben, dass manche Algorithmen nur einseitiges Feedback erhalten. So erhält man etwa Informationen darüber, ob freigelassene Straftäter rückfällig geworden sind oder nicht. Inhaftierte Straftäter können jedoch nicht beweisen, dass sie in Freiheit nicht erneut straffällig würden. Noch ist unklar, wie mit dieser asymmetrischen Rückkopplung umzugehen ist. Denkbar wäre durchaus, dass Systeme algorithmischer Entscheidungsfindung in solchen Situationen – je nach Schadenspotenzial – nicht eingesetzt werden sollten und ein **Verbot** für den Einsatz solcher Systeme durchgesetzt werden sollte.

Diese ersten beispielhaften Lösungsvorschläge zeigen bereits, dass es für viele Fehlerquellen in allen Phasen Ansätze gibt, mit denen man Mängel im Entwicklungs- und Einbettungsprozess von Entscheidungssystemen entdecken und beheben kann.

8 Fazit

Das vorliegende Arbeitspapier beschreibt den Entwicklungs- und Einbettungsprozess von algorithmischen Entscheidungssystemen, es zeigt die Fehlerquellen in den verschiedenen Phasen dieses Prozesses auf und skizziert beispielhaft erste Ansätze, mit denen man diese Fehler angehen kann.

Die Analyse macht deutlich, dass Fehler in allen Phasen des Prozesses auftauchen können. Es handelt sich dabei nicht nur um technische und handwerkliche Mängel bei der Programmierung. Fehler können vor allem entstehen, wenn das Entscheidungssystem in einen gesellschaftlichen Kontext eingebettet wird und Anwender mit ihm interagieren. Systeme algorithmischer Entscheidungsfindung sollten deshalb nicht für sich allein betrachtet, sondern immer als Teil eines soziotechnischen Gesamtgefüges gesehen werden. Dies ist vor allem auch wichtig, weil Entscheidungssysteme soziale Konsequenzen für die Teilhabe Einzelner haben können (z. B. Ablehnung von Bewerbern).

Die Erläuterung des Entwicklungsprozesses von Entscheidungssystemen zeigt zudem, dass in den unterschiedlichen Phasen verschiedene Akteure für Entscheidungen verantwortlich sind. Dies können sowohl Wissenschaftler und Programmierer in Unternehmen sein als auch Auftrag gebende Institutionen und Anwender. Bei Entscheidungssystemen, die hohe Anwenderzahlen verzeichnen, wächst die Anzahl der Beteiligten am Prozess daher schnell an – und damit auch das Fehlerpotenzial. Es wurde dargelegt, dass bei vielen Schritten eine interdisziplinäre und ethische Perspektive notwendig ist, die bei den oft technischen Ausbildungen der heutigen Data Scientists nicht unbedingt gegeben ist. Oft scheint eine solche Expertise auch in den Entwicklerteams von algorithmischen Systemen zu fehlen. Zum anderen fehlen den Anwendern oft Kompetenzen, wenn es darum geht, die Datengrundlage einzuschätzen und Ergebnisse richtig zu interpretieren.

Die Darstellung der verschiedenen Fehlerquellen in den Phasen des Prozesses weist darauf hin, dass die Fehler unterschiedlich gut entdeckt und behoben bzw. vermieden werden können. So gibt es einerseits handwerkliche Fehler im Algorithmendesign, die eher selten auftauchen. Falls sie doch passieren, können sie schnell entdeckt und behoben werden, solange deutlich ist, welches Problem der Algorithmus löst, und der Quellcode zugänglich ist. Andererseits existieren Fehler, denen nur schwer auf die Schliche zu kommen ist und entgegengewirkt werden kann, beispielweise bei fehlerhaften Operationalisierungen.

Fehler bei Prozessen algorithmischer Entscheidungsfindung sind letztendlich also bedingt durch eine komplexe Abfolge vieler Entscheidungen, an denen eine Vielzahl unterschiedlicher Akteure beteiligt ist. Fehler können in allen Phasen geschehen mit unterschiedlicher Tragweite und verschiedenen Anforderungen an ihre Bearbeitung. Das Papier zeigt mit einzelnen Beispielen, dass es für alle Phasen des Prozesses Lösungen geben kann, mit denen die meisten der Fehler mit mehr oder weniger Aufwand vermieden oder behoben werden können. Lösungsansätze, die in diesem Papier nur beispielhaft skizziert werden, müssen demnach auf verschiedenen Ebenen ansetzen, um sowohl Fehlern in der Entwicklung des ADM-Systems als Software als auch negativen Effekten bei seiner Einbettung in einem soziotechnischen System entgegenwirken zu können. Zudem müssen Verantwortliche mit entsprechenden Kompetenzen und einer Sensibilität für Fehler ausgestattet werden. Da algorithmische Entscheidungssysteme Auswirkungen auf die Gesellschaft haben können, ist zudem ein Diskurs darüber notwendig, welche Datengrundlage genutzt und wo solche Systeme eingesetzt werden sollen und welche Effekte gewollt sind. Diese Überlegungen sowie Lösungsansätze sollten in Zukunft diskutiert und konkreter ausgearbeitet werden.

9 Literatur

Angwin, Julia, Jeff Larson, Surya Mattu und Lauren Kirchner (2016). „Machine Bias – There's software used across the country to predict future criminals. And it's biased against blacks." *ProPublica*. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing (Download 15.12.2017).

Ariely, Dan (2010). *Predictably Irrational – The hidden forces that Shape Our Decisions*. London: Harper Collins Publishers.

Beuth, Patrick (2016). „Twitter-Nutzer machen Chatbot zur Rassistin". *Zeit Online* 24.03.2016. http://www.zeit.de/digital/internet/2016-03/microsoft-tay-chatbot-twitter-rassistisch (Download 15.12.2017).

Cosier, Colin (2017). „How Centrelink's ‚robodebt' ran off the rails". *Radio National*. http://www.abc.net.au/radionational/programs/backgroundbriefing/2017-03-05/8319442 (Download 15.12.2017).

Danziger, S, J. Levav und L. Avnaim-Pesso (2011), Extraneous factors in judicial decisions. Proceedings of the National Academy of the Sciences, 108, 6889-6892.

Flach, Peter (2012). *Machine Learning – The Art and Science of Algorithms that Make Sense of Data*. New York NY: Cambridge University Press.

Kahnemann, Daniel (2012). *Thinking, fast and slow*. London: Penguin Books Ltd.

Knaus, Christopher (2017). „Centrelink robo-debt system wrongly targets Australian of the Year finalist". *The Guardian* 16.1.2017. https://www.theguardian.com/australia-news/2017/jan/16/centrelink-robo-debt-system-wrongly-targets-australian-of-the-year-finalist (Download 15.12.2017).

Lischka, Konrad, und Anita Klingel (2017). *Wenn Maschinen Menschen bewerten*. Bertelsmann Stiftung. Gütersloh. (Auch online unter https://doi.org/10.11586/2017025, Download 15.12.2017).

Lischka, Konrad und Christian Stöcker (2017). *Digitale Öffentlichkeit: Wie algorithmische Prozesse den gesellschaftlichen Diskurs beeinflussen.* Bertelsmann Stiftung. Gütersloh. (Auch online unter https://doi.org/10.11586/2017028, Download 15.12.2017).

Mayer-Schönberger, Viktor, und Kenneth Cukier (2013). *Big Data: Die Revolution, die unser Leben verändern wird*. München: Redline Verlag.

O'Neil, Cathy (2016). *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. New York NY: Crown Publishing Group.

Pariser, Eli (2012). *Filter Bubble: Wie wir im Internet entmündigt werden*. München: Carl Hanser Verlag.

Rohde, Noelle (2017). „In Australien prüft eine Software die Sozialbezüge – und erfindet Schulden für 20.000 Menschen". https://algorithmenethik.de/2017/10/25/in-australien-prueft-eine-software-die-sozialbezuege-und-erfindet-schulden-fuer-20-000-menschen/ (Download 15.12.2017).

Vieth, Kilian, und Ben Wagner (2017). *Teilhabe, ausgerechnet. Wie algorithmische Prozesse Teilhabechancen beeinflussen können*. Bertelsmann Stiftung. Gütersloh. (Auch online unter https://doi.org/10.11586/2017027, Download 15.12.2017).

Walker, Joseph (2012). „Meet the New Boss: Big Data. Companies Trade In Hunch-Based Hiring for Computer Modeling". *Wall Street Journal* 20.9.2012. https://www.wsj.com/articles/SB10000872396390443890304578006252019616768 (Download 15.12.2017).

Zweig, Katharina A. (2016). *Network Analysis Literacy*. Wien: Springer Verlag.

10 Über die Autorin

Prof. Dr. Katharina A. Zweig, geb. Lehmann. Jahrgang 1976. Studium der Biochemie (1996–2001) und Studium der Bioinformatik (1998–2006) an der Eberhard Karls Universität Tübingen, parallel zu letzterem 2007 Promotion in der Informatik. 2008–2009 als Postdoc in der statistischen Biophysik an der ELTE Universität in Budapest, Ungarn. 2009–2012 Leiterin einer unabhängigen Nachwuchsgruppe am Interdisziplinären Zentrum für wissenschaftliches Rechnen (IWR) an der Universität Heidelberg; seit 2012 als Professorin für Graphentheorie und Analyse komplexer Netzwerke an der TU Kaiserslautern. Dort entwickelte sie federführend den deutschlandweit einzigartigen Studiengang „Sozioinformatik“. Dieser behandelt die Frage nach der Auswirkung des Einsatzes von IT-Systemen auf Individuum, Organisation und Gesellschaft. Katharina Zweig ist seit 2013 Juniorfellow der Gesellschaft für Informatik, wurde 2014 im Rahmen des Wissenschaftsjahres „Die digitale Gesellschaft“ als eine von Deutschlands 39 „Digitalen Köpfen“ ausgezeichnet und gründete 2016 mit Matthias Spielkamp, Lorenz Matzat und Lorena Jaume-Palasí die Initiative „Algorithm Watch“.

Arbeitsschwerpunkte: Analyse und Design von Algorithmen, Modellierung und Analyse komplexer Systeme als komplexe Netzwerke, Network Analysis Literacy, Algorithmic Accountability

11 Impulse Algorithmenethik

Alle Veröffentlichungen sind abrufbar unter: https://algorithmenethik.de/impulse/

Impuls Algorithmenethik #1: Konrad Lischka und Anita Klingel. „Wenn Maschinen Menschen bewerten“. Bertelsmann Stiftung, 2017. https://doi.org/10.11586/2017025

Impuls Algorithmenethik #2: Kilian Vieth, Ben Wagner und Bertelsmann Stiftung. „Teilhabe, ausgerechnet“. Bertelsmann Stiftung, 2017. https://doi.org/10.11586/2017027

Impuls Algorithmenethik #3: Konrad Lischka und Christian Stöcker. „Digitale Öffentlichkeit“. Bertelsmann Stiftung, 2017. https://doi.org/10.11586/2017028

Adresse | Kontakt

Bertelsmann Stiftung
Carl-Bertelsmann-Straße 256
33311 Gütersloh
Telefon +49 5241 81-8114

Dr. Sarah Fischer
Ethik der Algorithmen
Telefon +49 5241 81-81148
sarah.fischer@bertelsmann-stiftung.de

www.bertelsmann-stiftung.de

BertelsmannStiftung