

Ethische Deliberation für agile Softwareprozesse: EDAP-Schema

Von Niina Zuber, Severin Kacianka, Alexander Pretschner und Julian Nida-Rümelin

Ethisch entscheiden

Der Mensch sieht sich alltäglich mit der Frage konfrontiert, was er tun soll: Was eine richtige, angemessene, vernünftige oder gute Handlung sei. Mit zunehmender Verbreitung von Maschinen und Algorithmen, die Aktionen ohne menschliche Aufsicht ausführen, stellt sich diese Frage insbesondere auch bei der Entwicklung und dem Einsatz solcher Systeme, gerade weil diese direkten Einfluss sowohl auf das individuelle Leben jedes Einzelnen als auch die Gesellschaft insgesamt haben. Die Ethik soll hierauf Antworten finden, da sie diejenige Wissenschaft ist, die sich mit Fragen der Normativität beschäftigt. Insofern fällt sie Urteile über das, was sein soll, und sagt nichts darüber aus, was ist (Henning, 2019). Im Gegensatz zur Jurisprudenz regulieren ethische Urteile Handlungen nicht *ex post*, sondern dienen bereits *ex ante* der Strukturierung von Handlungszusammenhängen. Viele Alltagssituationen bedürfen keiner normativen Deliberation, da das erwünschte Verhalten, die angemessene Handlung, eindeutig – zumindest für den entsprechenden Zeitpunkt – bestimmbar ist: Viele Rechenregeln, Verkehrsregeln oder Höflichkeitsregeln fallen hierunter. Wir wissen, ohne darüber nachdenken zu müssen, was getan werden soll, beziehungsweise welches Verhalten erstrebenswert ist.¹ Diese dispositionellen Verhaltensweisen sind akzeptiert, internalisiert und bedürfen in jenem Moment keiner weiteren Reflexion – was jedoch nicht ausschließt, dass sie zu einem späteren Zeitpunkt überdacht und verändert werden können. Lebensweltlicher Komplexität wird somit durch angeeignete und gefestigte Dispositionen – also das, was in der aristotelischen Tradition als *aretai* (Tugenden) bezeichnet wird – begegnet (MacIntyre, 1981; Nussbaum, 1999; Foot, 2003; Nida-Rümelin, 2019; Vallor, 2016).

Anders verhält es sich bei Handlungskonflikten, also in Situationen, in denen unklar ist, welche Handlung, welcher Zustand oder welches Ziel erstrebenswert ist. Aufgabe der normativen Ethik ist es dann, zu reflektieren und zu begründen, welcher Handlungsgrund Orientierung bieten soll. Die Begründung ist Resultat einer ethischen Deliberation, welche normative Argumente systematisiert und zuletzt zu einem normativen Urteil führt: So gilt es, die Zulässigkeit oder Zumutbarkeit von Entscheidungen zu er- und begründen. In der ethischen

¹ Diese Regeln lassen sich auch Maschinen relativ einfach beibringen.

Deliberation werden Handlungsgründe gewichtet und unterschiedliche Handlungsoptionen abgewogen. Letztendlich muss die Entscheidung beziehungsweise die Handlungsintention wohlbegründet² sein, um als vernünftig gelten zu können (Lord, 2018; Kiesewetter, 2017; Wedgewood, 2017; Nida-Rümelin, 2020). Die Informatik kann diese normativen Entscheidungen dann in der Entwicklung von technischen Artefakten berücksichtigen und praktisch in ihrem Design umsetzen.

Dabei lassen sich normative Urteile nicht auf epistemische Tatsachen reduzieren. Ein Mehr an Wissen führt nicht notwendigerweise zu mehr Klarheit in der Entscheidungsfindung beziehungsweise zu einem besseren Urteil. Anders ausgedrückt: Gewissheit lässt sich nicht allein in Daten finden, weshalb Algorithmen allein zur Entscheidungsfindung nicht ausreichen, unter anderem gilt das für die weitverbreiteten Machine-Learning-Algorithmen, welche Korrelationen in Daten ausfindig machen oder die Dimensionalität von Daten reduzieren können. Diese Grenze der Algorithmisierung ist auch der Unterbestimmtheit normativer Handlungsgründe geschuldet. Diese Unterbestimmtheit tritt bei fundamentalen Normenkonflikten besonders deutlich zutage, ist aber auch die Folge der deontologischen Verfasstheit lebensweltlicher Moral und juridischer Beurteilung: Denn im Gegensatz zu konsequentialistischen (utilitaristischen) Systemen gibt es in deontologischen Systemen keine generelle Auflösung praktischer Konflikte durch Optimierung. Zwar sind durchaus Fälle denkbar, in denen der Handlungsgrund unmittelbar erkenntlich scheint, beispielsweise in der Pflicht, Verletzten zu helfen. Aber selbst hier müssen situativ-evaluative Perspektiven in der Abwägung der jeweiligen Einzelsituation Berücksichtigung finden, wie zum Beispiel Zeitdruck, Gefährdung des Hilfeleistenden etc. Entscheidungen bleiben somit kontingent. Es lässt sich eben nicht bei allen Tatsachen eindeutig, das heißt notwendig, bestimmen, welcher Handlungsgrund gelten soll. Sie entziehen sich somit einer Berechenbarkeit, weshalb folglich beim Entstehungsprozess technischer Objekte angesetzt werden muss. Zudem lassen sich normativ uneindeutige Situationen nicht nur durch Wissenserwerb lösen, denn sie fordern zusätzlich praktische Kompetenzen heraus, wie Entscheidungsstärke oder andere Charaktermerkmale. Die Handlungsanleitung.

² Maschinen basieren auf grundlegenden Annahmen der klassischen Entscheidungs- und Rationalitätstheorie, sodass wir von „rationalen Maschinen-Konstrukten“ sprechen können. Allerdings können wir nicht von „vernünftigen Maschinen“ sprechen, welche Entscheidungen begründen, wohlbegründete Ausnahmen machen oder sich wider ihr Programm verhalten können (vgl. Nida-Rümelin/Weidenfeld, 2018; besonders Kapitel „Autonomie und Determinismus in der digitalen Welt“).

beziehungsweise Normativität erschöpft sich nicht in juridischen Gesetzen³ oder sozialen Normen. Daher ist es nicht weiter verwunderlich, dass im Fokus des aktuellen öffentlichen Diskurses gerade Algorithmen stehen, die durch ihre enorme Reichweite viele Menschen betreffen und beeinflussen können und so eine große normative Kraft entwickeln. Der Mensch steht alltäglich in Situationen, in denen er entscheiden muss und in denen er auch richtig, angemessen und gut entscheiden will. Hierbei kommt auch Werten beziehungsweise Werthaltungen eine herausragende Rolle zu, denn sie können konative Einstellungen konstituieren, das heißt, sie können Intentionen hervorbringen, welche ausdrücken, wie wir uns in dieser oder jener Handlungssituation entscheiden *wollen*. Individuell für gut befundene und im Laufe des Lebens internalisierte Werte können also Handlungsgründe als besonders wohlbegründet erscheinen lassen, sodass diese Gründe einen zwanglosen Zwang für die Ausführung der Handlung ausüben. Folglich kann eine ethische Deliberation als ein Denkmodus, ein Abwägungsprozess, verstanden werden, der zu jenen normativ adäquaten Handlungen führen soll.

Normative Gründe müssen keine moralischen Gründe sein, aber alle – auch moralische – Gründe sind normativ, das heißt, sie können mit mehr oder weniger Berechtigung zur Realisierung drängen. Ökonomische und juristische Tatsachen oder soziale Regeln können ebenso normative Gründe konstituieren und unseren Handlungen Orientierung bieten. Konflikte entstehen, wenn sich Gründe gegenseitig ausschließen. So kann ein ökonomischer Grund einen moralischen Grund ausschließen oder zwei ökonomische Gründe weisen in unterschiedliche Richtungen. Von moralischen Konflikten muss man insbesondere dann sprechen, wenn sie die individuelle(n) Handlungsfreiheit(en) und Rechte anderer tangieren – wenn also eine Handlungsausführung Freiheit(en) und Rechte beschneidet (Nida-Rümelin, 2016).

Maschinen ethisch bauen

Hinter den Schlagwörtern „robot ethics“ (Lin/Jenkins/Abney, 2017), „moral machines“ (Wallach, 2010), aber auch „value sensitive design“ (Friedman/Hendry, 2019) verbirgt sich die Fragestellung, welche Regeln gelten sollen, um ethische Maschinen entwickeln zu können und wie diese ausgestaltet sein müssen, damit sie eingehalten werden können. Wenn von ethischen

³ Gesetze entstehen als Reaktion auf Schieflagen, somit a posteriori. Sobald sie ein Gesetz sind, fließen sie als Norm in die Handlungswahl mit ein.

Maschinen gesprochen wird, meinen wir Maschinen beziehungsweise Softwaresysteme, die juristischen, kulturellen und moralischen Standards gerecht werden. Dies sind Systeme, in denen vom Entwicklerteam Bewertungen im Systemdesign reflektiert, beabsichtigt und inkludiert werden, sodass die Maschine die Bewertung der Entwickler*innen implementiert. Damit sind also keine Systeme gemeint, die Architekturen und Algorithmen um eine ethische Dimension, in Form einer Variablen oder einer Regel, zu erweitern versuchen, um die menschliche Fähigkeit der normativen Deliberation zu modellieren (Conitzer/Sinnott-Armstrong/et al., 2017; Misselhorn, 2018). Der Wunsch nach einem Softwaresystem, das selbstständig ein ethisch abgesichertes Resultat liefert, indem es die menschliche Deliberationsfähigkeit nachzuahmen versucht, übersteigt den heutigen wissenschaftlichen Forschungsstand bei Weitem. Softwaresysteme, die einen Rekrutierungsprozess an einer Universität steuern, Drohnen, die im Krieg den Gegner beschießen, oder Softwaresysteme, welche die sozialen Absicherungssysteme vor Betrug schützen sollen (O’Neil, 2017; Eubanks, 2018; Noble, 2018), werden nicht dadurch zu ethischen Systemen, dass ein um eine ethische Regel erweiterter Algorithmus implementiert oder eine architektonische Struktur entwickelt wurde, welche Datenpunkte nach bestimmten Regeln induktiv und deduktiv clustert und subsumiert. Die Erwartungshaltung, dass ethische Maschinen quasi von selbst ein ethisch wünschenswertes Ziel erreichen beziehungsweise dieses sich sogar selbst zu setzen vermögen, weil wir sie zur Ethik programmiert hätten, übersieht die zentrale menschliche Leistung der ethischen Deliberation und Entscheidungsfreiheit: eben die menschliche Kompetenz, in einer unsicheren, komplexen Lebenswelt wohlbegründete Entscheidungen treffen zu können.

Allein, dass in der Gestaltung technischer Objekte technische Dimensionen im Hinblick auf gesellschaftliche, legale, ökonomische, ästhetische und moralische Kontexte kontinuierlich bedacht werden müssen, verdeutlicht die Schwierigkeit, ethische Diffizilität zu reduzieren oder gar an Maschinen zu delegieren. Das Front- sowie das Backend-Design, die Nutzer-Bedienung oder die gesellschaftlichen Implikationen, die durch einen Systemeinsatz verursacht werden, müssen in unterschiedlicher Dringlichkeit je individuell und in ihrer Kompositionalität berücksichtigt werden. Negative Externalitäten, die beispielsweise durch *airbnb* auf dem Wohnungsmarkt entstanden sind (Lee, 2016), kann man nicht durch eine einmalige Implementierung eines um ein moralisches Prinzip angereicherten Algorithmus abwenden beziehungsweise abfedern, der dann sozusagen selbstständig aus gegebenen Daten die moralisch perfekte Lösung berechnet. Die Denkleistung, das Nachdenken über die Ausgestaltung sinnstiftender Lebensformen kann nicht technologisch substituiert werden – und

im Sinne eines Digitalen Humanismus (Nida-Rümelin/Weidenfeld, 2018) ist dieser Versuch auch nicht wünschenswert.

Beispiel: Machine Learning

Dies lässt sich gut am sogenannten Maschinellen Lernen (Machine Learning, ML) zeigen, worunter Methoden verstanden werden, die in Trainingsdaten Muster ausfindig machen und anschließend neue, unbekannte Datensätze nach diesen Mustern klassifizieren. Wichtig ist, dass es sich hierbei um algorithmisch aufgebaute, statistische Modelle handelt, die Daten extrahieren und ordnen, wobei das Zustandekommen des Resultats in der Regel nicht einmal vom Entwicklerteam vollständig nachvollzogen werden kann. Auch aus technischer Sicht bietet Machine Learning somit keine Funktionalität, um ethische Entscheidungen treffen zu können. Selbst scheinbar einfache Sachverhalte können bei ML-Algorithmen unerwartete Ergebnisse liefern. Ein einfaches Beispiel ist die Meinungsfreiheit: Folgt man John Stuart Mill (1869), so soll in einer freien Gesellschaft Widerspruch zur herrschenden Meinung geduldet und nach Henry David Thoreau (1849) sogar unterstützt werden. Wenn man sich jetzt ein abstraktes Beispiel vorstellt, in dem die Mehrheit sagt, etwas sei „100“, und nur ein einziges Individuum vertritt die Ansicht, es sei „-1“, hat ein linearer ML-Algorithmus zwei Möglichkeiten: Entweder wird „-1“ als Outlier behandelt; man ignoriert folglich den Widerspruch und unterdrückt diese Meinung, oder man bildet irgendeine Form von Mittelwert, sodass der Konsens ein Wert wäre, der nahe bei „100“ läge, aber diesen nie erreicht. Nicht-lineare Algorithmen wie neuronale Netze oder sogenannte Decision Trees, würden, falls keine weiteren Eigenschaften des Individuums identifiziert werden können, nur die Mehrheitsmeinung abbilden können. Deshalb wird beim Trainieren solcher Algorithmen versucht, sogenanntes Overfitting, also eine Überanpassung des Modells an die Trainingsdaten, zu verhindern. Ein Algorithmus, mithilfe dessen wir ausweisen, dass alle außer einem Individuum etwas mit „100“ bezeichnen, jenes andere Individuum aber mit „-1“, ist also aus Gründen der Generalität des Modells nicht erstrebenswert. Aktuell verbreitete ML-Algorithmen sind somit nicht in der Lage, eine als ethisch angesehene Meinungsvielfalt im Sinne von John Stuart Mill adäquat abzubilden.

Machine Learning (Mohri/Rostamizadeh/Talwalkar, 2018) ist im Wesentlichen eine Sammlung von informatischen Methoden, die aus existierenden oder vergangenen Datenpunkten Vorhersagen über zukünftige Datenpunkte treffen. Dieses Vorgehen teilen alle

Methoden, weshalb es im Kern darum geht, dass möglichst geschickt eine mehrdimensionale Kurve durch die meistens vieldimensionalen Daten gelegt wird, mit der Erwartung, dass eben diese Kurve auch neue Datenpunkte in die „richtige“ Klasse einsortiert. Besonders erfolgreich ist Machine Learning bei Aufgaben, in denen es um die Klassifikation von Daten geht (zum Beispiel Bild- oder Spracherkennung), beim Erkennen von Regressionen (zum Beispiel die Preisentwicklung von Produkten) und Reihenfolgen (zum Beispiel die relevanteste Webseite für eine Suchanfrage) sowie beim Clustering von Daten (zum Beispiel Personen in bestimmte Gruppen einzuteilen).

Machine-Learning-Verfahren sind also rein statistisch zu verstehen und sind nicht in irgendeiner Form (künstlich) intelligent. Im Gegenteil: Pearl zeigt sehr anschaulich, dass aktuelle Verfahren nicht in der Lage sind, über einfache Korrelationen hinauszugehen (Pearl/Mackenzie, 2018, 27ff). Sie sind zwar exzellent darin, Muster und Assoziationen in Daten zu erkennen und diese Muster zu „lernen“, allerdings ist es nicht möglich, mit ihnen kontrafaktische Fragen zu beantworten. Pearls Beispiel aufgreifend ist es somit sehr leicht, einen ML-Algorithmus einzusetzen, um zu erkennen, welche Produkte in einem Supermarkt zusammen gekauft werden. Allerdings ist es unmöglich, mit einem solchen Algorithmus die Frage zu beantworten, mit welcher Wahrscheinlichkeit ein Käufer von Produkt A dieses auch gekauft hätte, wenn es doppelt so teuer gewesen wäre. Gerade solche Fragen nach alternativen Handlungsoptionen, nach anderen Begründungsstrukturen, sind für eine ethische Deliberation essenziell. Der Grund hierfür ist, dass Daten alleine diesen Sachverhalt nicht erklärbar machen können, sondern es einer Theorie oder eines Modells der Welt und ihrer kausalen Zusammenhänge bedarf – in diesem Falle einer soliden ökonomischen Theorie und einer Modellierung derselben. Diese Formen kausaler Schlüsse sind bis dato noch dem Menschen vorbehalten.⁴

Softwaresysteme, gleichgültig wie smart der Algorithmus ist, führen keine Abwägung von Gründen durch; sie können nicht selbstständig zwischen unterschiedlichen Denk- und Abwägungsmodi wechseln und ihr Vorgehen rechtfertigen, um letztendlich sinnstiftend wirksam zu werden. Softwaresysteme sind keine autonomen, moralischen Akteure: Sie bleiben immer ausführend. Vernünftige, das heißt ethisch erstrebenswerte Entscheidungen zu treffen, bleibt also dem Menschen vorbehalten, weshalb gerade bestehende Entwicklungsprozesse normative Abwägungsverfahren auch auf Entwicklerebene integrieren müssen.

⁴ Auch wenn es dazu schon Forschungen gibt, z. B. Dasgupta et al. (2019).

Code of Conducts und Ethik-Canvas: *Ethical Deliberation for Agile Processes* (EDAP-Schema)

Es gibt bereits Prozesse beziehungsweise Erweiterungen für bestehende Systementwicklungsverfahren hinsichtlich *Safety*- und *Security*-Anforderungen, die dafür sorgen sollen, dass ein System sowohl *safe* als auch *secure* entwickelt wird. Tatsächlich macht man eine automatische Tür *safe*, indem man technisch sicherstellt, dass sie keine Gliedmaßen einzwängt. Dafür muss die Tür eben nicht über eine Intuition à la „meine Umwelt sicherer machen“ verfügen, sondern einzig gemäß bestimmter Sicherheitskriterien gebaut werden. In Analogie hierzu kann und soll ein ML-Algorithmus keine ethischen Entscheidungen treffen, sondern realisiert die Bewertungen des Entwicklerteams. Der aktuelle Forschungsstand erlaubt keine Entwicklung von eigenständig „reflektierenden“ und „Theorie-bildenden“ Algorithmen. Es ist schon von daher unabdingbar, dass Entwickler*innen sowie Administrator*innen und Nutzer*innen⁶ bei der Entwicklung und auch während des Softwareeinsatzes die Frage nach wünschenswerten Zielen stellen. Folglich muss die Vereinbarkeit von Algorithmen, Datenquellen und Inputs, wie zum Beispiel Steuerbefehle, mit dem, was als ethisch wünschenswert gilt, bedacht werden. Dies bedeutet auch, dass Entwickler*innen sich selbstkritisch fragen müssen, ob ihr geleisteter Beitrag mit ihrem Berufsethos⁷ in Einklang steht, das heißt unter anderem, ob die Testfälle wirklich gut und ausreichend sind, aber zugleich auch, welche Verantwortung sie im Entwicklungsprozess übernehmen können und müssen. Der Umgang mit datenbasierten Technologien erfordert gerade aufgrund der vielfältigen zentralen Arbeitsprozesse unterschiedliche Kompetenzen mit verschiedenen Anforderungen – von der Entwicklung eines Programms und der Speicherung der Daten über die Wartung von technischen Artefakten bis hin zur umsatzzielenden Markteinführung –, weshalb eine Betrachtung der einzelnen Tätigkeiten und ihrer Bedeutung für ein nachhaltiges Produkt

⁵ Im Deutschen werden die Begriffe „Safety“ und „Security“ oft unter dem Begriff „Sicherheit“ zusammengefasst. Tatsächlich bezeichnet „Safety“ die „funktionale Sicherheit“, also die Eigenschaft eines Systems, keinen Schaden anzurichten, während „Security“ für „Informationssicherheit“ steht, also für die Eigenschaft eines Systems, die Vertraulichkeit, Integrität und Verfügbarkeit von Daten und Funktionen zu gewährleisten.

⁶ Entwickler*innen arbeiten an einem System, bis es „in Produktion geht“. Anschließend übernehmen Administrator*innen die Pflege und Wartung des Systems. Unter dem Begriff „DevOps“ (engl. *Development* und *Operations*) versteht man einen Managementansatz, bei dem diese strikte Trennung aufgeweicht wird. Nutzer*innen des Systems verwenden die zur Verfügung gestellten Funktionalitäten.

⁷ Vgl. dazu z. B. die Leitlinien der Gesellschaft für Informatik: <https://gi.de/ueber-uns/organisation/unsere-ethischen-leitlinien> (zuletzt aufgerufen am 4.2.2020)

notwendig ist, denn jene Arbeitsteilung kann schnell in eine Verantwortungsdiffusion führen (Battaglia/Mukerji/Nida-Rümelin, 2014). Zwar diffundiert die Verantwortung nicht ausschließlich zwischen Unternehmer*innen und Entwickler*innen, denn ein verantwortungsvoller Umgang mit dem technischen Produkt ist auch eine moralische Pflicht der Nutzer*innen. Die Endverbraucher*innen können das Produkt missbrauchen, sie können es schlicht falsch bedienen oder die Anwendung missverstehen. In diesem Beitrag möchten wir uns jedoch auf die ethische Verantwortung der Softwareentwickler*innen beschränken. Hierfür müssen die verschiedenen Aufgabenbereiche in ethischen Leitfäden adressiert werden, damit normative Belange erkannt und anschließend auch an der dafür angemessenen Stelle berücksichtigt werden können. Nur so kann ein Berufsethos gelebt und verinnerlicht werden. Die „Swiss Alliance for Data-based Services“ verfolgt gerade diesen Ansatz, indem sie einen ethischen Code formuliert, der sich am Lebenszyklus des technischen Produkts orientiert, um gezielt Arbeitsrollen und Verantwortungen mit ethischen Kernfragen verknüpfen zu können (Loi/Heitz/et al., 2019). Die Mehrheit der Codes of Conduct zahlreicher privater und öffentlicher Allianzen⁸ sowie der fünf großen Techgiganten⁹ basieren jedoch auf einem jeweils individuell differenzierten Wertekanon. Diese entsprechend exklusiven Bewertungslisten unterscheiden sich in ihrer Konkretisierung und Hierarchisierung der ethisch wünschenswerten Ausgestaltung technischer Objekte. Die Divergenzen lassen sich schon damit erklären, dass weder zwischen Arbeitsprozessen oder der eingesetzten Technologie (wohl zwischen Machine Learning und regelbasierten Systemen) noch zwischen Produktkategorien unterschieden wird, obgleich die normativ wünschenswerte Ausrichtung autonomer Waffensysteme einer anderen Schwerpunktsetzung in ihrer ethischen Ausrichtung bedarf als die Fertigung einer betriebswirtschaftlichen Controlling-Software.

Die Werte und ethischen Prinzipien, die diese schier unüberschaubare Anzahl an Codes of Conducts und Guidelines zusammentragen, sind bisher in keiner Literaturstudie untersucht und statistisch ausgewertet worden.¹⁰ Zwar enthalten die meisten Kodizes wesentliche Werte, wie

⁸ Z. B. private Zusammenschlüsse bzw. Organisationen wie das *IEEE (Institute of Electrical and Electronics Engineers)*, die *Partnership on AI* (gegründet 2016 als Zusammenschluss von Industrie und Non-Profit-Organisationen mit akademischen Institutionen, z. B. IBM, Googles DeepMind, Microsoft; Apple kam 2017 hinzu) oder *Open AI* (2015 gegründet als Non-Profit-Organisation zur Erforschung von KI; größter Geldgeber ist Elon Musk). Aber auch globale und öffentliche Allianzen wie das *Future of Life Institute* mit den *Asilomar AI Principles* (23 Prinzipien für einen ethischen Umgang mit KI, beschlossen auf der Asilomar-Konferenz im Jahr 2017 und von 1.273 KI/Robotik-Forscher*innen und 2.541 Menschen unterzeichnet).

⁹ Apple, Amazon, Facebook, Google und Microsoft.

¹⁰ Die Autor*innen dieses Beitrags werden im Jahr 2020 eine deskriptive, auf Künstliche Intelligenz gestützte Werteanalyse vornehmen.

zum Beispiel Wohlfahrt und Autonomie, ebenso wie ethische Prinzipien, zum Beispiel konsequentialistische oder deontologische Abwägungsregeln, allerdings ohne konkrete Anwendungsmöglichkeiten aufzuzeigen oder Unterstützung in der Transferleistung anzubieten. Diesem Sachverhalt versucht man mithilfe von ethischen Schemata zu begegnen, wie sie beispielsweise vom *Open Data Institute*¹¹, dem *Center for Humane Technology*¹² oder von *The Ethics Canvas*¹³ entwickelt werden. Diese wiederum fungieren als Aktionselemente, die zum Nachdenken auffordern und auf die Multidimensionalität technischer Objekte aufmerksam machen: Sie sollen helfen, Querverbindungen und Wechselwirkungen komplexer Zusammenhänge zwischen dem technischen Objekt und seinem Ökosystem ersichtlich werden zu lassen, indem sie beispielsweise psychologische Zustände, Stakeholder-Belange, rechtswissenschaftliche Grundlagen und Policies abfragen, denn erst die Einbettung in einen Kontext eines technischen Artefakts führt zu unbeabsichtigten Distortionen oder begründet Zielkonflikte.¹⁴ Diese Fragenkataloge unterstützen jedoch weder bei einer Werte-Priorisierung oder einer Konfliktbewältigung noch bei der Formulierung von Verhältnismäßigkeiten oder gar bei der Einbettung ethischer Deliberationszyklen in betriebswirtschaftliche Arbeitsprozesse. Sie bleiben auf einer eher deskriptiven Wertefindungsebene, *ohne eine normative Urteilsfindung systematisch zu adressieren*. Es ist daher wichtig, dass zwischen einer Phase der deskriptiven Werte-Auffindung und einer Phase der kritischen Werte-Überprüfung sowie einer Übertragung von Werten und Prinzipien in technische Objekte unterschieden wird. *Values in Design* ist genau das Forschungsgebiet, welches sich mit der Einbindung ethischer Werte in technische Objekte beschäftigt.¹⁵ Somit rücken der Design- und der Entwicklungsprozess in den Mittelpunkt der Betrachtung, und die ethische Evaluation reduziert sich nicht auf Stakeholder- und Werteanalysen, auch wenn diesen eine zentrale Rolle zukommen muss. Mithilfe von deskriptiven System- und Werteanalysen können erst Wertekonflikte lokalisiert und im Anschluss daran Werte-trade-offs, Werte-Spannungen oder Werte-Konflikte einer wissenschaftlichen, normativen Betrachtung unterzogen werden.¹⁶ Essenziell für die Transferleistung und Implementierung ethisch wünschenswerter Aspekte in

¹¹ *Open Data Institute* <https://theodi.org/article/data-ethics-canvas/> (zuletzt aufgerufen am 4.2.2020).

¹² *Center for Humane Technology* <https://humanetech.com/> (zuletzt aufgerufen am 4.2.2020).

¹³ *The Ethics Canvas* <https://www.ethicscanvas.org/download/handbook.pdf> (zuletzt aufgerufen am 4.2.2020).

¹⁴ Im Jahr 2020 werden die Autor*innen dieses Beitrags im Rahmen des Projekts „Ethik in der agilen Softwareentwicklung“ des Bayerischen Forschungsinstituts für digitale Transformation (bid) die verschiedenen ethischen Schemata wissenschaftlich auswerten.

¹⁵ Vgl. dazu Simon (2016a), Friedman/Hendry (2019) und Friedman (1997).

¹⁶ Beispiele, wie eine ethische Werte-Anwendung erfolgen kann, finden sich u. a. in Friedman/Kahn/Borning/Huldtgren (2013), Simon (2016b) und Simon (2012).

technische Artefakte ist also die Verbindung einer deskriptiven Werteanalyse mit einer anschließenden normativen Deliberation hinsichtlich der technischen Zielsetzung, damit wohlbegründete Handlungsempfehlungen gemacht werden können. Ebenjene Schritte gilt es methodisch einzufangen, zu strukturieren und zu systematisieren.

Die Systemanalyse soll gerade den ethischen Deliberationsprozess in die Entwicklung von Softwaresystemen integrieren und sowohl die Systematisierung und Implementierung normativer Urteile als auch deren Überprüfbarkeit strukturieren. Dies führt bestenfalls in wohlbegründete Präferenzrelationen und in definierte Optionen-Räume, was erstens einem innovativen Perspektivenwechsel und zweitens einer humanen Gestaltung unserer Lebensräume dienlich ist. Unser Vorschlag ist, gerade die Vorteile des iterativen, agilen Managementverfahrens zu nutzen, um die gewünschten Resultate zielführend auszurichten: Als Beispiel möchten wir kurz auf Scrum, eines dieser Verfahren, eingehen. Wie andere dieser Management-Methodiken auch (weitere bekannte Methoden sind unter anderem Kanban, Extreme Programming oder Feature-Driven Development) zielt Scrum darauf ab, hierarchisch strukturierte Bürokratien in Arbeitsorganisationen abzubauen, um so auf Veränderungen dynamisch und ohne Zeitverzögerung (also „agil“) (re-)agieren zu können. Im Kern teilt Scrum die Produktentwicklung in kleinere Features auf. Jedes dieser Features soll sich in kurzen (typischerweise zwei Wochen; nicht länger als vier Wochen), abgeschlossenen Iterationen, sogenannten Sprints, fertigstellen lassen. Alle Features werden in das sogenannte Backlog aufgenommen und dann von einer*m Domänen-Expertin*en oder Kundin*en, dem Product Owner, priorisiert. Der Scrum Master wiederum übernimmt die relevante Kommunikation sowohl zwischen dem Product Owner als auch dem Entwicklungsteam und unterstützt Letzteres dabei, möglichst effizient in der Projektumsetzung vorzugehen. Scrum verlangt nach mehreren Arten von Meetings. Im Daily Scrum trifft sich das gesamte Team, um den aktuellen Status abzugleichen, im Sprint Review wird das Produkt reflektiert und die aktuellen Ergebnisse werden dem Product Owner und anderen Stakeholdern präsentiert, und in der Sprint Retrospective wird der Sprint selbst analysiert und nach Prozessverbesserungen gesucht. Es ist denkbar, dass in diesen Prozessen des agilen Arbeitens und der kontinuierlichen Übersicht normative Deliberationseinheiten inkludiert werden können, sodass bereits existierende, unternehmensweite oder gesellschaftliche Erwartungshaltungen abgeglichen und neue

Entwicklungsansätze, Einsatzmöglichkeiten oder Handhabungsmethoden entstehen können.¹⁷ Der gesamte Entwicklungsprozess – von der konzeptionellen Phase bis hin zum finalen Einsatz – wird in die Systematisierung miteinbezogen, sodass eine normativ erstrebenswerte Ausgestaltung des technischen Produkts gewährleistet ist. Softwareentwickler*innen werden dadurch für die normativ angemessene Konzeption sensibilisiert. Die ethische Deliberation wird somit integraler Bestandteil der Produktentwicklung. Die Methode ist offen, das heißt, es ist gerade nicht schon im Vorfeld festgelegt, welches ethisch wünschenswerte Bewertungskriterium oder Prinzip final angelegt werden soll; vielmehr gilt es, dieses anhand der Fakten- und Argumentenlage zu ermitteln. Das EDAP-Schema ermöglicht eine Integration unterschiedlicher Motivationslagen und soll assoziativ-intuitiven Brainstormings entgegenwirken. Die humanistische Grundlage normativer Ethik beschreibt keine Normen, Regeln und Gesetze, wonach Menschen ihr Verhalten de facto ausrichten, sondern fordert auf, darüber nachzudenken, welche Handlung wünschenswert wäre. Für Algorithmen-Designer*innen bedeutet das beispielsweise herauszufinden, welche Outputs geboten oder wünschenswert sein sollen. Eben diese Deliberation, diese Praxis, bedeutet Ethik.

EDAP-Schema¹⁸

Dieser *reasoning*-Prozess – das heißt die normative Deliberation beziehungsweise das Nachdenken darüber, welche technische Ausrichtung wünschenswert ist beziehungsweise welche Ausgestaltung wünschenswert wäre und wie dies technisch implementiert werden kann – soll mit der Anwendung des EDAP-Schemas¹⁹ zielführend und strukturiert erfolgen. Es ist eben dieses Deliberieren, das die Fähigkeit des Menschen, selbst Autor*in ihres*seines Lebens sein zu können, ausmacht (Nida-Rümelin, 2005). Gerade diese Praxis ist zugleich das humanistische Moment, welches als Referenzpunkt verstanden werden muss und das es technisch zu unterstützen und nicht zu untergraben gilt. Und diesen Deliberationsprozess soll das EDAP-Schema unterstützen. Dies gelingt, indem es den assoziativen Deliberationsprozess strukturiert und eine Priorisierung der Handlungsmöglichkeiten vornimmt. Das Verfahren soll

¹⁷ Am bidt forschen die Autor*innen dieses Beitrags im Rahmen des Projekts „Ethik in der agilen Softwareentwicklung“ an der Frage, wie normative Deliberationen in agile Prozessmanagementstrukturen optimal eingebunden werden können.

¹⁸ Vgl. dazu Friedman/Hendry (2019), Nissenbaum (2005), Betzler (2011), Nida-Rümelin (2015), Freeman (2010), Donaldson/Preston (1995), Kohlberg (1996), Hildebrandt/Musholt (2020).

¹⁹ Im Rahmen des Projekts „Ethische Deliberation in agilen Softwareprozessen“ des Bayerischen Forschungsinstituts für digitale Transformation wird dieses Schema erarbeitet. Es handelt sich um das EDAP-Schema 2019, welches die Autor*innen dieses Beitrags derzeit in der Praxis überprüfen.

zu einer einfachen, übersichtlichen und effektiven Handlungsempfehlung führen, die es in der technischen Umsetzung zu berücksichtigen gilt. So lässt sich beispielsweise die Privatsphäre durch den Verzicht auf Daten schützen oder die Autonomie der Nutzer*innen durch eine transparente Gestaltung von Algorithmen stärken. Außerdem kann man auch durch die geschickte Auswahl an Content möglichen extremistischen Tendenzen entgegenwirken. Der Entwicklungsprozess wird nicht gehemmt, vielmehr können aufgrund der normativen Perspektiveneinnahme innovative Ausgestaltungsmöglichkeiten entstehen.

Das EDAP-Schema basiert auf rationalitätstheoretischen, ethischen, ökonomischen und entwicklungspsychologischen Theorien. Es gliedert sich in acht Phasen, die sich an den unterschiedlichen Entwicklungsschritten ausrichten: Zu Beginn stehen die Aufnahme des Gesamtsystems und die Beschreibung des Anforderungsprofils. In der zweiten Phase geht es um eine Bestandsaufnahme derjenigen Werte, die es zu berücksichtigen gilt: Welche Werte und Empfehlungen sind relevant? Hier werden wir uns auch auf unsere deskriptive Analyse von Code of Conducts stützen, um einen soliden Einstieg in das Thema der Ziel- und Wertekonflikte zu gewährleisten. Anschließend wenden wir uns dem konkreten Fall zu, um diejenigen ethischen Werte zu lokalisieren, die es in diesem konkreten technischen Objekt zu berücksichtigen gilt. Dies führt dann direkt in Phase III, in der wünschenswerte Ausrichtungen, die aufgrund verschiedener Biases unterminiert werden, zu erfassen und zu beachten sind. Einen kritischen Umgang mit Werten und möglichen Zielkonflikten thematisiert anschließend Phase IV: Welche Werte stehen in Konflikt zueinander? Was bedeutet das für die Umsetzung? Und sind die vorgefundenen Werte moralisch verpflichtend? Wie stehen moralische Gründe und andere Gründe in Konflikt? In den meisten Fällen kann direkt zu Phase VII übergegangen werden, wobei die Phasen der technischen Ausgestaltung und der Verifikation bereits von Anbeginn in die Überlegungen mit aufgenommen werden müssen. Die Phase der normativ-theoretischen Systemüberprüfung (Phase VI) ist vor allem im Kontext von schwierigen Konflikten hilfreich beziehungsweise im Kontext der Weiterbildung und ethischen Sensibilisierung relevant. Die Phasen sind nicht linear zu betrachten, das heißt, während des Entwicklungsprozesses muss zwischen den Deliberationsphasen gewechselt werden, um so auch die gegenseitigen Wechselwirkungen hervorheben zu können.

Unser EDAP-Schema eignet sich, um die alltäglichen und üblichen Entscheidungen in Entwicklerteams kurz, aber gezielt zu reflektieren und die Tragweite des eigenen Handelns zu

explizieren. Als Gedankenexperiment²⁰ dient uns dabei ein Softwaresystem, das die Gesprächsprotokolle von Callcenter-Mitarbeiter*innen für Trainingszwecke aufzeichnet. Wir setzen voraus, dass die Aufzeichnungen und üblichen Analysen, wie etwa die Gesprächslänge oder das Feedback der Kund*innen, bereits etabliert sind und funktionieren. An dieser Stelle wird das Entwicklerteam gebeten, das System um eine *Sentiment Analysis* der Gespräche²¹ zu erweitern. Ziel ist es, die Schulung der Mitarbeiter*innen weiter zu verbessern und beispielsweise eine Vorselektion der Audiomitschnitte nach aggressiven Gesprächen vorzunehmen. Im Deliberationsschema geben wir nun neben den generellen Leitfragen auch Beispiele, die sich auf unser Gedankenexperiment beziehen.

Phase I: Deskriptive Systemanalyse	
Universum	Beschreibung der Gesamtsituation <i>s. o.</i>
Stakeholder	Wer beeinflusst das technische System oder wird von diesem beeinflusst? <i>Bsp.: Kund*innen, Callcenter-Mitarbeiter*innen, Trainingsverantwortliche ...</i>
Technische Strategien	Welche technischen Möglichkeiten gibt es, auf die Zielsetzung zu reagieren? <i>Bsp.: Sentiment Analysis</i>
Phase II: Deskriptive Werteanalyse²²	
Universum	Welche Werte müssen berücksichtigt werden? Menschenrechte etc. Welche Werte werden als wünschenswert deklariert? Werteanalyse und Clustering von Codes of Conducts / Guidelines. <i>Bsp.: ACM Code of Ethics, IEEE u. a.</i>
Stakeholderanalyse	Welche Werte vertreten die verschiedenen Stakeholder? Corporate Social Responsibility / Digital Responsibility / Investor Relationship / Leitbild etc. <i>Bsp.: Mitarbeiter*innen via Betriebsrat: Führungsebene:</i>

²⁰ Eine detaillierte theoretische Beschreibung wird Ende 2020 veröffentlicht.

²¹ Dabei werden, häufig mittels Machine-Learning-Algorithmen, die Gefühle hinter gesprochener Sprache ermittelt. Für ein Beispiel mit Audio und Video siehe: Zadeh/Chen/Poria//CambriaMorency (2017): Tensor Fusion Network for Multimodal Sentiment Analysis. *arXiv preprint arXiv:1707.07250*.

²² Auswertung von Codes of Conduct und Ethical Guidelines nach ethischen Werten und Theorien mithilfe von inhaltlichen Analysen in Arbeit.

	<i>Trainingsleiter*innen:</i>		
Phase III: Technical Analysis			
Pre-existing bias	Begründen sich in Institutionen, Praktiken, Einstellungen. <i>Bsp: Bisher wurden in der Firma alle verfügbaren Daten – sofern legal – auch genutzt, um die Effizienz und Performance der Mitarbeiter*innen zu messen. Es gab auch einen Vorfall der illegalen Datennutzung.</i>		
Technical bias	Entstehen aufgrund technischer Limitationen und Betrachtungen. <i>Bsp: Die Sentiment Analysis ist nicht präzise und klassifiziert Aussagen immer wieder falsch. Die Genauigkeit von Menschen beträgt 85.7 %, ML-Algorithmen kommen auf bis zu 77.1 %²³</i>		
Emergent bias	Entstehen im Zusammenhang mit Nutzer*innen und durch Einbindung in reale Lebenssituationen. <i>Bsp: Dies würde sich erst im Laufe der Nutzung zeigen.</i>		
Phase IV: Werte- bzw. Zielkonflikte			
Fragestellungen	Evaluative Einstellung: Welche Werte der Stakeholder stehen in einem trade-off, in einem Spannungsverhältnis oder in einem Konflikt zueinander? <i>Bsp: Der Betriebsrat vertritt das Recht auf Privatsphäre der Mitarbeiter*innen und die Trainingsbeauftragten das Gewinninteresse des Unternehmens.</i>		
	Kritische Werte-Reflektion: Wertimmanente Konflikte		
---> volitionale Einstellung: Umsetzung der Werte			
---> Vortheoretische Deliberation: Abwägung und Ordnen auf Basis empirischer Sachverhalte; Explizieren von Handlungsoptionen und Begründung. (Präferenzenrelation). <i>Bsp: Als Entwicklerteam ist uns das Wohl der Firma, die uns beauftragt, wichtiger als ein möglicher, aber unwahrscheinlicher Missbrauch bzw. negative Konsequenzen für die Callcenter-Mitarbeiter*innen. Oder: Wir möchten nicht, dass unser Name mit einem Produkt assoziiert wird, das dazu genutzt wird, die Menschenwürde zu verletzen.</i>			
Phase V: Ethische Systemüberprüfung: Dieser Schritt ist dann notwendig, wenn aus Phase IV keine vernünftige Handlungsempfehlung folgt.			
Fragestellungen	Welchen Werten bzw. welchen Gründen sollen wir Folge leisten?		
	Deontologisch	Konsequentialistisch	Tugendethik/Berufsethos
Argumente Pro	<i>Bsp.: Die Privatsphäre steht unter absolutem Schutz.</i>	<i>Bsp.: Für den Erfolg der Firma kann man kleinere Kompromisse machen.</i>	<i>Bsp.: Einen Auftrag soll man so gut es geht erfüllen.</i>
Argumente Contra	<i>Bsp.: Der Kunde ist immer König.</i>	<i>Bsp.: Ein Datenschutzskandal kann die Firma und auch</i>	<i>Bsp.: Schreibe keine Software, die zur Überwachung</i>

²³ Vgl. dazu Zadeh/Chen/Poria/Cambria/Morency (2017).

		<i>unseren individuellen Ruf langfristig schädigen.</i>	<i>missbraucht werden kann.²⁴</i>
Theoretische Deliberation:	Ist es wünschenswert, dass eine solche Technologie Einzug in unsere Lebenswelt findet?		
Phase VI: Urteilsphase (Kohärenz)			
<p>Sollen (moralische Gründe) / Wollen (ökonomische Gründe etc.) wir überhaupt über eine technische Umsetzung des Features nachdenken? Passen Nutzen und Kosten der Stakeholder zusammen? Wie fügt sich die technische Umsetzung in das Unternehmensleitbild ein (Stichwort: Corporate Digital Responsibility) / Wie gehen wir mit den Urteilen aus Phase IV und ggfs. V um?</p>			
<p>wenn das Urteil negativ ausfällt → Beginn bei Phase II wenn das Urteil positiv ausfällt → Phase VII</p>			
Phase VII: Technische Umsetzbarkeit			
<p>(1) Technisches Problem: Umreißen Sie, ob und wie Sie technisch vorgehen möchten. Übertragung und Integration der Analyse auf das Design: Klassifikation der Nutzer*innen, Frontend-/Oberflächendesign, Backend-Design, Umfeld</p> <p>(2) Können Sie Ihre normativen Urteile technisch umsetzen, d. h., können Sie ein Feature entwickeln, das z. B. die Arbeitszeit trackt, ohne dass die Rechte der Arbeitnehmer*innen verletzt werden? Was kann den Arbeitnehmer*innen zugemutet werden?</p> <p>(3) Ist eine technische Umsetzung machbar, die das normative Urteil widerspiegelt? Welche normativen Aspekte können nicht integriert werden und warum? Was bedeutet das für Sie: Wollen Sie das Feature entwickeln?</p>			
Phase VIII: Verifikation			
<ul style="list-style-type: none"> • Das System wird technisch und empirisch dahingehend überprüft, ob die Standards gewährleistet werden können: Wie könnte eine Überprüfung aussehen? Wie technisch aufwendig ist eine Überprüfung? • Technisch: „Was sind gute Testfälle für das System?“ 			

Ob das System nun gebaut werden soll beziehungsweise wie es gebaut und eingesetzt werden soll, obliegt dem Entwicklerteam. Wichtig ist, dass die Gründe transparent und nachvollziehbar expliziert wurden. Wir gehen davon aus, dass bereits die gemeinsame Durchführung der ethischen Deliberation zu einem verantwortungsvollen Umgang mit Softwaresystemen führt, aufseiten der Unternehmer*innen, der Entwickler*innen, aber auch der Nutzer*innen.²⁵

²⁴ <https://www.gnu.org/proprietary/proprietary-surveillance.en.html>

²⁵ Eine Studie hierzu ist bereits in Planung.

Fazit

Aktuell ist es technisch unmöglich, Maschinen beziehungsweise Algorithmen so zu programmieren, dass diese selbstständig ethisch wünschenswerte Resultate berechnen. Nichtsdestotrotz werden Maschinen immer mehr Teil unserer Lebenswelt werden und tiefer in unsere Gesellschaft und unseren Alltag integriert sein. Aufgrund ihrer enormen Reichweite, ihres Einflusses auf menschliche Entscheidungen haben solche Systeme eine starke normative Kraft. Softwaresysteme lenken beabsichtigt oder unbeabsichtigt Entscheidungen, weshalb sowohl im Entwicklungsprozess als auch im Technologie-Einsatz auf ungewollte, aber auch gewollte Implikationen Rücksicht genommen werden sollte. Aktuell werden diese Design-Entscheidungen oft willkürlich, das heißt meist intuitiv und assoziativ, durch das Zusammenspiel von Entwickler*innen und Technologie getroffen. Sehr oft setzt sich dabei das zumeist implizite ethische Verständnis von Vorgesetzten oder rhetorisch starken Minderheiten durch. Es ist daher ausschlaggebend, diesem häufig unstrukturierten Vorgehen mit einem semistrukturierten, auf wissenschaftlichen Erkenntnissen beruhenden Prozess entgegenzuwirken, um normative Belange zu lokalisieren und – wenn und wo notwendig – zu diskutieren und zu berücksichtigen. Das hier vorgestellte EDAP-Schema fügt sich nahtlos in bestehende Entwicklungsprozesse ein, unterstützt die ethische Deliberation nicht nur des Entwicklerteams, sondern des ganzen Unternehmens – und dies ohne wesentliche Mehrarbeit. Als solches leistet es einen Beitrag zur Debatte um ethische Systeme und bietet einen pragmatischen und praxistauglichen Ansatz, Systeme unter ethischen Gesichtspunkten zu entwickeln.

Literatur

Battaglia, F. / Mukerji, N. / Nida-Rümelin, J. (Hrsg.) (2014): *Technology and Responsibility*. Pisa: Pisa University Press.

Betzler, M. (2011): *Erziehung zur Autonomie der Elternpflicht*. In: *Deutsche Zeitschrift für Philosophie*, 59, Heft 6, 937–953.

Conitzer, V. / Sinnott-Armstrong, W. / et al. (2017): *Moral Decision Making Frameworks for Artificial Intelligence*. In: *Thirty-first AAAI-Conference on Artificial Intelligence*.

Dasgupta, I. / et al. (2019): *Causal Reasoning from Meta-Reinforcement Learning*. *arXiv preprint arXiv:1901.08162*.

- Donaldson, T. / Preston, L. E. (1995): The Stakeholder Theory of the Corporation: Concepts, Evidence, and Implications. In: *Academy of Management Review*, 20(1)1, 65–91.
- Eubanks, V. (2018): *Automating Inequality - How High Tech Tools Profile, Police, and Punish the Poor*. New York: St. Martin's Press.
- Foot, P. (2003): *Virtues and Vices. And Other Essay in Moral Philosophy*. Oxford: Oxford University Press.
- Freeman, R. E. (2010): *Strategic Management: A Stakeholder Approach*. Cambridge: Cambridge University Press.
- Friedman, B. (1997): *Human Values and the Design of Computer Technology*. Stanford: Center for the Study of Language and Information Stanford University.
- Friedman, B. / Kahn, P. H. / Borning, A. / Huldtgren, A. (2013): Value Sensitive Design and Information Systems. In: Doorn, N. / Schuurbiers, D. / Van de Poel, I. / Gorman, M. (Hrsg.): *Early Engagement and New Technologies: Opening up the Laboratory*. Philosophy of Engineering and Technology, vol 16. Dordrecht: Springer.
- Friedman, B. / Hendry, D. G. (2019): *Value Sensitive Design – Shaping Technology with Moral Imagination*. Cambridge, MA: MIT University Press.
- Henning, T. (2019): *Allgemeine Ethik*. Paderborn: Wilhelm Fink Verlag.
- Hildebrandt, F. / Musholt, K. (2020): Teaching Rationality – Sustained Shared Thinking as a Means for Learning to Navigate the Space of Reason. In: *Journal of Philosophy of Education*, Vol. 00, No. 0.
- Kiesewetter, B. (2017): *The Normativity of Rationality*. Oxford: Oxford University Press.
- Kohlberg, L. (1996): *Die Psychologie der Moralentwicklung*. Frankfurt am Main: Suhrkamp.
- Lee, D. (2016): How Airbnb Short-term Rentals Exacerbate Los Angeles's Affordable Housing Crisis: Analysis and Policy Recommendations. In: *Harvard Law & Policy Review*, 10, 229.
- Lin, P. / Ryan, J. / Abney, K. (Hrsg.) (2017): *robot ethics 2.0*. Oxford: Oxford University Press.
- Loi, M. / Heitz, C. / et. al. (2019): Towards an Ethical Code for Data-based Industry. In: 2019 6th Swiss Conference on Data Science. IEEE, 612.
- Lord, E. (2018): *The Importance of Being Rational*. Oxford: Oxford University Press.
- MacIntyre, A. (1981): *After Virtue*. Notre Dame: University Press of Notre Dame.
- Mill, J. S. (Erstveröffentlichung 1869): *Über die Freiheit*. Stuttgart: Reclam Verlag.
- Misselhorn, C. (2018): *Grundfragen der Maschinenethik*. Stuttgart: Reclam Verlag.
- Mohri, M. / Rostamizadeh, A. / Talwalkar, A. (2018): *Foundations of Machine Learning*. Cambridge: MIT University Press.
- Nida-Rümelin, J. (2005): *Über menschliche Freiheit*. Stuttgart: Reclam.
- Nida-Rümelin, J. (2015): Theoretische und angewandte Ethik: Paradigmen, Begründungen, Bereiche. In: *Angewandte Ethik – Die Bereichsethiken und ihre theoretische Fundierung*, 2–88.
- Nida-Rümelin, J. (2016): Gründe und Lebenswelt. In: *Information Philosophie*, Heft 2, 8–19.
- Nida-Rümelin, J. / Weidenfeld, N. (2018): *Digitaler Humanismus – Eine Ethik für das Zeitalter der Künstlichen Intelligenz*. München: Piper.
- Nida-Rümelin, J. (2019): *Die Optimierungsfalle - Philosophie einer humanen Ökonomie*. München: btb Verlag.
- Nida-Rümelin, J. (i. E. 2020): *Theorie einer praktischen Vernunft*. Berlin: De Gruyter.
- Nissenbaum, H. (2005): Values in Technical Design In: *Encyclopedia of Science, Technology, and Ethics*, hrsg. von C. Mitcham. New York: Macmillan, ixvi-ixx.

- Noble, S. U. (2018): *Algorithms of Oppression: How Search Engines Reinforce Racism*. North Yorkshire: Combined Academic Publ.
- Nussbaum, M. C. (1999): *Virtue Ethics: A Misleading Category?* In: *The Journal of Ethics*, 3, Nr. 3, 163–201.
- O’Neil, C. (2017): *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. London: Penguin.
- Pearl, J. / Mackenzie, D. (2018): *The Book of Why: the New Science of Cause and Effect*. New York: Basic Books.
- Simon, J. (2012): *E-Democracy and Values in Information Systems Design*. In: *Proceedings of the XXV World Congress of IVR, Special Workshop on „Legitimacy 2.0: E-democracy and Public Opinion in the Digital Age“*, 40–64.
- Simon, J. (2016a): *Values in Design*. In: Heesen, J. (Hrsg.): *Handbuch Medien- und Informationsethik*. Stuttgart: J. B. Metzler.
- Simon, J. (2016b): *Value-Sensitive Design and Responsible Research and Innovation*. In: Hansson, S.-O. (Hrsg.): *The Ethics of Technology – Methods and Approaches*. London: Rowman & Littlefield International, 219–236.
- Thoreau, H. D. (Erstveröffentlichung 1849): *Civil Disobedience*. CreateSpace Independent Publishing Platform.
- Vallor, S. (2016): *Technology and the Virtues. A Philosophical Guide to a Future Worth Wanting*. New York: Oxford University Press.
- Wallach, W. (2010): *Moral Machines. Teaching Robots Right from Wrong*. New York: Oxford University Press.
- Wedgwood, R. (2017): *The Value of Rationality*. Oxford: Oxford University Press.
- Zadeh, A. / Chen, M. / Poria, S. / Cambria, E. / Morency, L. P. (2017): *Tensor Fusion Network for Multimodal Sentiment Analysis*. *arXiv preprint arXiv:1707.07250*.

