

1 Grenzen der KI

Theoretisch, praktisch, ethisch

Klaus Mainzer

Abstract: Die technisch-praktischen Grenzen heutiger Computer und KI hängen von den theoretischen Grenzen digitaler Berechenbarkeit ab. Zunächst orientierte sich KI am logischen Schließen der symbolischen Logik. Das statistische Lernen aus großen Datenmassen führte schließlich zum Machine Learning, das die technisch-wirtschaftlichen Durchbrüche der heutigen KI dominiert. Dazu gehören auch Chatbots wie ChatGPT, die täuschend echt Texte und Unterhaltungen auf anspruchsvollem Niveau generieren, aber auch gefährliche Fehlinformationen und Diskriminierungen in den Medien verbreiten können. Was fehlt, ist logisches und kausales Begründen und Verstehen.

Einige Grenzen digitaler Berechenbarkeit, Entscheidbarkeit und Lösbarkeit von Problemen lassen sich theoretisch jenseits der Turing-Berechenbarkeit überwinden. Neuartige analoge Rechnerstrukturen, die am menschlichen Gehirn orientiert sind (z. B. memristive Systeme), ermöglichen ihre technisch-praktische Umsetzung. Ebenso lässt sich zunächst theoretisch beweisen, dass im Formalismus der Quantenmechanik Probleme in polynomaler Zeit lösbar werden, die in den Grenzen klassischer Turing-Berechenbarkeit nicht-polynomial sind und damit praktisch unlösbar. Die technisch-praktische Umsetzung erfordert allerdings den Quantencomputer.

Daher wird eine Erweiterung zu einer hybriden KI gefordert, die digitales, analoges und Quanten-Computing in neuromorphen Rechnerstrukturen verbindet. Anwendungen werden sowohl für Roboter als auch in intelligente Infrastrukturen erörtert. Ziel ist eine Künstliche Intelligenz als Dienstleistung am Menschen, um Vertrauen in KI durch Technikgestaltung zu erreichen. Ethische Orientierung der KI als humane Dienstleistung sind Menschenwürde und Autonomie, wie sie in den Grundrechten demokratischer Verfassungen gefordert werden.

1.1 Grenzen digitaler Berechenbarkeit und Künstlicher Intelligenz

Traditionell wurde KI (Künstliche Intelligenz) als Simulation intelligenten menschlichen Denkens und Handelns aufgefasst. Alan Turings schlug daher in seiner Definition der KI von 1950 ein Simulationsspiel (Turingtest) vor, wonach ein technisches

System ›intelligent‹ genannt werden sollte, wenn es in seinen Antworten, Reaktionen und seinem Verhalten nicht von einem Menschen unterschieden werden kann.

1936 hatte Turing bereits theoretisch definiert, was ein digitaler Computer überhaupt ist. Sein Konzept der Turingmaschine besteht aus einem (nach beiden Seiten) unbegrenztem Band, auf dessen Felder Symbole eines endlichen Alphabets nach einfachen Befehlen gedruckt, gelöscht, verschoben und gelesen werden können. Das Band dient also als Speicher und die Lese- und Druckvorrichtung als Prozessor. Im einfachsten Fall handelt es sich bei den Symbolen um die Bits 0 und 1, mit denen alle digitalen Informationen kodiert werden können. Eine Turingmaschine heißt deterministisch, wenn die Ausführung der Befehle nacheinander eindeutig festgelegt ist. Nach der Churchschen These kann jeder digitale Algorithmus durch eine Turingmaschine simuliert werden. Danach ist die Turingmaschine Repräsentant eines digitalen Algorithmus überhaupt.

Technisch wurde Turings theoretisches Konzept der Turingmaschine in den 1940er Jahren sowohl durch Zuses als auch von Neumanns programmgesteuertem digitalen elektronischen Computer realisiert. Die theoretisch auf Turing zurückgehende Trennung von Prozessor und Speicher wird als von Neumann-Architektur bezeichnet und bis heute in digitalen Computern vom Smartphone bis zum Supercomputer verwendet. Universell nannte Turing eine Turingmaschine dann, wenn sie alle möglichen Computer mit ihren Maschinencodes simulieren kann. Eine universelle Turingmaschine ist daher das theoretische Konzept eines digitalen Vielzweck-Computers.

Mit der Turingmaschine lag erstmals eine logisch-mathematisch präzise Definition eines digitalen Algorithmus überhaupt vor. Noch Hilbert war Anfang des 20. Jahrhunderts der Auffassung von Leibniz, dass alle logisch-mathematischen Probleme im Prinzip entscheidbar seien. Das Halteproblem für Turingmaschinen ist ein Beispiel für ein prinzipiell unentscheidbares Problem: Prinzipiell gibt es kein allgemeines Entscheidungsverfahren, ob eine beliebige Turingmaschine für einen beliebigen Input nach endlich vielen Schritten stoppt oder nicht. Die Annahme eines solchen Verfahrens würde der Existenz einer prinzipiell nicht berechenbaren reellen Zahl widersprechen.

Daran wird deutlich, wie die Grenzen von mathematischen Entscheidungsverfahren mit dem Konzept digitaler Berechenbarkeit und digitaler Computer zusammenhängen. Ebenso folgt Gödels Unvollständigkeit direkt aus der Unentscheidbarkeit des Halteproblems. Gäbe es ein vollständiges formales System mit formalen Beweisen für alle mathematischen Wahrheiten, dann gäbe es ein Entscheidungsverfahren, ob ein beliebiges Computerprogramm stoppt oder nicht.

Für die Grenzen digitalen Computing wurden Komplexitätsklassen zur Unterscheidung von P-, NP-, NP-harten und NP-vollständigen Problemen eingeführt.¹ Da die Leistungen von Künstlicher Intelligenz von unterschiedlichen Algorithmenklassen abhängen, ist die Berechenbarkeits- und Komplexitätstheorie grundlegend. In Alltagsentscheidungen greifen wir häufig auf Hintergrundwissen zurück, das wir selbst nicht entscheiden und beweisen können. Das gilt auch in einer arbeitsteiligen

¹ Mainzer (2019): 36.

Forschung, in der wir uns häufig des Wissens von Nachbardisziplinen bedienen, ohne selbst dieses Wissen entscheiden zu können. Häufig handelt es sich dabei auch um angenommene Hypothesen. Diese Art von erweiterter Intelligenz lässt sich in Algorithmen abbilden. Turing hatte dazu das Konzept einer »Orakel-Maschine« eingeführt. In diesem Fall greift eine Turingmaschine zusätzlich auf eine Instanz (Orakel) zurück, die Fragen beantwortet, ohne selbst dieses Wissen entscheiden zu können.

1.2 Grenzen der symbolischen und subsymbolischen KI

1.2.1 Symbolische KI: Logik und Deduktion

Auf der Grundlage digitaler Berechenbarkeit orientierte sich KI in einer ersten Phase an formalen (symbolischen) Kalkülen der Logik, mit denen Problemlösungen regelbasiert abgeleitet werden können. Man spricht deshalb auch von symbolischer KI. Ein typisches Beispiel ist das automatische Beweisen mit logischen Deduktionen, die sich mit Computerprogrammen realisieren lassen. Dabei wird der Beweis der Allgemeingültigkeit eines logischen Schlusses durch ein logisches Widerlegungsverfahrens gefunden. Bei der Resolutionsmethode wird die entsprechende Formel mit Annahme des Gegenteils der Allgemeingültigkeit systematisch nach logischen Regeln aufgelöst und auf einen Widerspruch geführt. In der Aussagenlogik terminieren alle Resolutionsalgorithmen, in der Prädikatenlogik nur in Teilsystemen.²

Automatisierung bedeutet bis zu einem bestimmten Grad auch Autonomie, da Computerprogramme die Beweistätigkeit eines Mathematikers übernehmen. Wissensbasierte Expertensysteme sind Computerprogramme, die Wissen über ein spezielles Gebiet speichern und ansammeln, aus dem Wissen automatisch Schlussfolgerungen ziehen, um zu konkreten Problemen des Gebietes Lösungen anzubieten. Im Unterschied zu menschlichen Experteninnen und Experten ist das Wissen eines Expertensystems aber auf eine spezialisierte Informationsbasis beschränkt ohne allgemeines und strukturelles Wissen über die Welt.³

Um ein Expertensystem zu bauen, muss das Wissen der Expertin oder des Experten in Regeln gefasst werden, in eine Programmiersprache übersetzt und mit einer Problemlösungsstrategie bearbeitet werden. Die Architektur eines Expertensystems besteht daher aus den folgenden Komponenten: Wissensbasis, Problemlösungskomponente (Ableitungssystem), Erklärungskomponente, Wissenserwerb, Dialogkomponente. In dieser Architektur werden zugleich die Grenzen symbolischer KI deutlich: Fähigkeiten, die nicht oder nur schwer symbolisch erfasst und regelbasiert simuliert werden können, bleiben der symbolischen KI verschlossen.

² Robinson (1965).

³ Puppe (1988); Mainzer (1990).

1.2.2 Subsymbolische KI: Statistik und Induktion

Sensorische und motorische Fähigkeiten werden nicht aus Lehrbuchwissen logisch abgeleitet, sondern aus Beispielen erlernt, trainiert und eingeübt. So lernen wir, uns motorisch zu bewegen und in einer Vielzahl sensorischer Daten Muster und Zusammenhänge zu erkennen, an denen wir unser Handeln und Entscheiden orientieren können. Da diese Fähigkeiten nicht von ihrer symbolischen Repräsentation abhängen, spricht man auch von subsymbolischer KI. An die Stelle der formalen Schlüsse der Logik tritt nun die Statistik der Daten. Beim statistischen Lernen sollen allgemeine Abhängigkeiten und Zusammenhänge aus endlich vielen Beobachtungsdaten durch Algorithmen abgeleitet werden⁴. An die Stelle der Deduktion in der symbolischen KI tritt also in der subsymbolischen KI die Induktion. Dazu können wir uns ein naturwissenschaftliches Experiment vorstellen, bei dem in einer Serie von veränderten Bedingungen (Inputs) entsprechende Ergebnisse (Outputs) folgen. In der Medizin könnte es sich um eine Person handeln, die auf Medikamente in bestimmter Weise reagiert.

Dabei nehmen wir an, dass die entsprechenden Paare von Input- und Outputdaten unabhängig durch dasselbe Zufallsexperiment erzeugt werden. Statistisch sagt man deshalb, dass die endliche Folge von Beobachtungsdaten $(x_1, y_1), \dots, (x_n, y_n)$ mit Inputs x_i und Outputs y_i ($i = 1, \dots, n$) durch Zufallsvariablen $(X_1, Y_1), \dots, (X_n, Y_n)$ realisiert wird, denen eine Wahrscheinlichkeitsverteilung $P_{X,Y}$ zugrunde liegt. Algorithmen sollen nun Eigenschaften der Wahrscheinlichkeitsverteilung $P_{X,Y}$ ableiten. Ein Beispiel wäre die Erwartungswahrscheinlichkeit, mit der für einen gegebenen Input ein entsprechender Output auftritt. Es kann sich aber auch um eine Klassifikationsaufgabe handeln: Eine Datenmenge soll auf zwei Klassen aufgeteilt werden. Mit welcher Wahrscheinlichkeit gehört ein Element der Datenmenge (Input) eher zu der einen oder anderen Klasse (Output)? Wir sprechen in diesem Fall auch von binärer Mustererkennung.

Die derzeitigen Erfolge des Machine Learning scheinen die These zu bestätigen, dass es auf möglichst große Datenmengen ankommt, die mit immer stärkerer Computerpower bearbeitet werden. Die erkannten Regularitäten hängen dann aber nur von der Wahrscheinlichkeitsverteilung der statistischen Daten ab.

In der Automatisierung statistischen Lernens nehmen neuronale Netze mit Lernalgorithmen eine Schlüsselrolle ein. Neuronale Netze sind vereinfachte Rechenmodelle nach dem Vorbild des menschlichen Gehirns, in denen Neuronen mit Synapsen verbunden sind. Die Intensität der neurochemischen Signale, die zwischen den Neuronen ausgesendet werden, sind im Modell durch Zahlen als Synapsengewichte repräsentiert. Probabilistische Netzwerke haben experimentell eine große Ähnlichkeit mit biologischen neuronalen Netzen. Werden Zellen entfernt oder einzelne Synapsengewichte um kleine Beträge verändert, erweisen sie sich als fehlertolerant gegenüber kleineren Störungen wie das menschliche Gehirn z. B. bei kleineren Unfallschäden. Das menschliche Gehirn arbeitet mit Schichten paralleler Signalverarbeitung. So sind

⁴ Vapnik (1998).

z. B. zwischen einer sensorischen Inputschicht und einer motorischen Outputschicht interne Zwischenschritte neuronaler Signalverarbeitung geschaltet, die nicht mit der Außenwelt in Verbindung stehen.

Tatsächlich lässt sich auch in technischen neuronalen Netzen die Repräsentations- und Problemlösungskapazität steigern, indem verschiedene lernfähige Schichten mit möglichst vielen Neuronen zwischengeschaltet werden. Die erste Schicht erhält das Eingabemuster. Jedes Neuron dieser Schicht hat Verbindungen zu jedem Neuron der nächsten Schicht. Aktivierungsfunktionen sorgen für die Übertragung von Signalen und aktivieren die nachgeordneten Neuronen („Feuern“). Die Hintereinanderschaltung setzt sich fort, bis die letzte Schicht erreicht ist und ein Aktivitätsmuster abgibt.⁵

Wir sprechen von überwachten Lernverfahren, wenn der zu lernende Prototyp (z. B. die Wiedererkennung eines Musters) bekannt ist und die jeweiligen Fehlerabweichungen daran gemessen werden können. Ein Lernalgorithmus muss die synaptischen Gewichte so lange verändern, bis ein Aktivitätsmuster in der Outputschicht herauskommt, das möglichst wenig vom Prototyp abweicht.

Ein effektives Verfahren besteht darin, für jedes Neuron der Outputschicht die Fehlerabweichung von tatsächlichem und gewünschtem Output zu berechnen und dann über die Schichten des Netzwerks zurückzuverfolgen. Wir sprechen dann von einem Backpropagation-Algorithmus. Die Absicht ist, durch genügend viele Lernschritte für ein Vorgabemuster den Fehler auf Null bzw. vernachlässigbar kleine Werte zu vermindern.

1.2.3 Anwendungsbeispiel von statistischem Lernen in KI-Chatbots

Ein spektakuläres Anwendungsbeispiel subsymbolischer KI sind Chatbots wie ChatGPT (Generative Pre-trained Transformer), der wegen seiner verblüffenden Möglichkeiten als automatischer Textgenerator seit dem 30. November 2022 innerhalb weniger Tage mit Millionen von Nutzerinnen und Nutzern mehr Followers hatte als soziale Medien wie z. B. Instagram und Spotify. ChatGPT kann Texte von Schulaufgaben auf gymnasialem Niveau bis zu Texten von Seminararbeiten mittleren universitären Niveaus generieren. Auf der Grundlage eines „Large Language Models“ kann man sich mit diesem KI-Programm über Businesspläne unterhalten oder das Schreiben eines Lieds, Gedichts oder Romanfragmenten in einem bestimmten Stil in Auftrag geben.⁶

Tatsächlich beruht das Sprachmodell von ChatGPT auf einer gewaltigen Menge von Texten (Big Data), die dem System von Menschen eintrainiert wurden. Es handelt sich also um ein Beispiel des Machine Learning auf der Grundlage von statistischer Lerntheorie und Mustererkennung, wie es im vorherigen Abschnitt erklärt wurde. Das ehrgeizige Ziel ist dabei, eine zentrale Grenze der symbolischen KI zu

5 Hornik/Stinchcombe/White (1989).

6 Kushwaha (2022).

überwinden, die in ihren wissensbasierten Expertensystemen auf das Fachwissen von Spezialistinnen und Spezialisten (z. B. medizinisches Fachwissen in einer speziellen medizinischen Disziplin) beschränkt war, sofern es in logisch-regelbasierte Formeln übersetzt werden konnte.

Eine Expertin oder ein Experte wie z. B. eine Medizinerin oder ein Mediziner verfügt aber auch über ein unbewusstes Können aufgrund von Erfahrung und Hintergrundwissen, das über das Lehrbuchwissen hinaus in sein Handeln und Entscheiden einfließt. Was in den Expertensystemen fehlte, war das allgemeine Hintergrundwissen und die Erfahrung und Intuition eines Menschen. Mit Steigerung der Rechenleistung und der Bewältigung großer Datenmassen mit Modellen statistischen Lernens wird nun das Ziel verfolgt, auch das allgemeine ‚Weltwissen‘ von uns Menschen auf die Maschine zu bringen.

Dazu wird der Chatbot mit Texten aus Nachrichten, Büchern, sozialen Medien, online-Foren, Bildern, Filmen und gesprochenen Sprachtexten trainiert. Algorithmen dienen dazu, aus den Trainingsdaten zu lernen. Der Chatbot reproduziert Muster, die er in den gespeicherten Daten erkennt. Das geschieht nach den gleichen Verfahren, mit denen bei der Gesichtserkennung Personenbilder aus Bilddateien erkannt werden. Die reproduzierten Texte werden mit trainierten Beispieldaten verglichen und so durch Algorithmen des verstärkenden Lernens (reinforcement learning) schrittweise verbessert. Dabei können auch Korrekturen durchgeführt werden, wenn Korrelationen der eintrainierten Daten z. B. zu Diskriminierungen führen. Ähnlich wie bei indoktrinierten Menschen, kann ein solches Fehlverhalten aufgrund des Umfangs der eintrainierten Datenmengen nie ausgeschlossen werden. Da diese Chatbots in den sozialen Medien auf große Akzeptanz stoßen, können sie auch gefährliche Desinformationen hervorrufen.

Letztlich ist auch ChatGPT nichts anderes als eine stochastische Maschine, die Daten, Texte, Bilder und Gesprochenes mit Mustererkennungsalgorithmen re kombiniert und rekonfiguriert. Machine Learning berechnet statistische Erwartungswahrscheinlichkeiten von Daten und digitale Simulationen. Aufgrund der modernen Rechnertechnologien, die gewaltige Datenmengen speichern und schnelle Lernalgorithmen anwenden können, entstehen allerdings verblüffende Ergebnisse, die einen großen Teil von menschlichem Hintergrundwissen und Intuition simulieren. Damit zeigt sich aber auch, auf welchen Mechanismen unsere Konversations- und Kulturwelten beruhen – Reproduktionen und Rekombinationen von Mustern, die weitgehend von Maschinen übernommen werden können. Selbst die Sozial-, Kultur- und Geisteswissenschaften sind davor nicht gefeit, vom Journalismus ganz abgesehen.

Wittgenstein nannte das ‚Sprachspiele‘, die nach bestimmten Regeln funktionieren. Das Originelle besteht häufig nur in einer kleinen Veränderung und Variante der gewohnten Sprachspiele und ‚Narrative‘. Im Machine Learning ist mittlerweile die Rede von ‚stochastischen Papageien‘. Positiv gewendet eignet sich daher der ChatGPT zur Entlarvung der Mechanismen von Kulturbetrieb und Journalismus. Dort wird man anspruchsvoller werden müssen, um nicht durch Maschinen ersetzt zu werden.

Was der ChatGPT aber überhaupt nicht kann, ist einfachste Mathematik. Hier helfen »kluge« Sprechblasen und ihre Assoziationen, Rekombinationen und stochastischen Variationen nicht. ChatGPT kennt Zahlen nur, wenn sie von eintrainierten Texten extrahiert werden können. So könnte die Definition einer Primzahl reproduziert werden, wenn dieser Text irgendwo im Speicher von ChatGPT auftaucht. Aber daraus Schlüsse ziehen und entscheiden, ob eine vorliegende Zahl eine Primzahl ist oder nicht, kann ChatGPT nicht. Logisches und kausales Denken sind ihm fremd. Es wird geraten und assoziiert.⁷ ChatGPT kann auch keine Computerprogramme schreiben, sondern imitiert und rekombiniert nur gespeicherte Vorlagen und Fragmente durch Raten – das aber auf verbüffend hohem Niveau, das selbst von »gebildeten« Menschen nicht zu unterscheiden ist. ChatGPT besteht daher in vielen Anwendungsbereichen den Turingtest.

1.2.4 Vom statistischen zum kausalen Lernen

Statistisches Lernen und Schließen aus Daten reichen also nicht aus. Wir müssen vielmehr die kausalen Zusammenhänge von Ursachen und Wirkungen hinter den Messdaten erkennen.⁸ Diese kausalen Zusammenhänge hängen von den Gesetzen der jeweiligen Anwendungsdomeäne unserer Forschungsmethoden ab, also den Gesetzen der Physik, den Gesetzen der Biochemie und des Zellwachstums im Beispiel der Krebsforschung, etc. Wäre es anders, könnten wir mit den Methoden des statistischen Lernens und Schließen bereits die Probleme dieser Welt lösen:

Statistisches Lernen und Schließen ohne kausales Domänenwissen ist blind – bei noch so großer Datenmenge (Big Data) und Rechenpower!

Die Auseinandersetzung zwischen probabilistischem und kausalem Denken ist keineswegs neu, sondern wurde erkenntnistheoretisch bereits in der Philosophie des 18. Jahrhunderts zwischen David Hume (1711–1776) und Immanuel Kant (1724–1804) ausgefochten. Nach Hume beruht alle Erkenntnis auf sinnlichen Eindrücken (Daten), die psychologisch »assoziiert« werden. Es gibt danach keine Kausalitätsgesetze von Ursache und Wirkung, sondern nur Assoziationen von Eindrücken (z. B. Blitz und Donner), die mit (statistischer) Häufigkeit »gewohnheitsmäßig« korreliert werden.⁹ Nach Kant sind Kausalitätsgesetze als vernunftmäßig gebildete Hypothesen möglich, die experimentell überprüft werden können. Ihre Bildung beruht nicht auf psychologischen Assoziationen, sondern auf der Kategorie der Kausalität,¹⁰ die für Vorhersagen auf der Grundlage von Erfahrung angenommen und überprüft werden kann. Nach Kant ist dieses Verfahren seit Galileo Galilei in der Physik in Gebrauch, die so erst zur Wissenschaft wurde.

7 Mainzer/Kahle (2022).

8 Pearl (2009).

9 Hume (1993): 95.

10 Kant 1900ff., KrV B 106.

Neben der Statistik der Daten bedarf es also zusätzlicher Gesetzes- und Strukturannahmen der Anwendungsdomänen, die durch Experimente und Interventionen überprüft werden. Kausale Erklärungsmodelle (z. B. das Planetenmodell oder ein Tumormodell) erfüllen die Gesetzes- und Strukturannahmen einer Theorie (z. B. Newtons Gravitationstheorie oder die Gesetze der Zellbiologie):

Beim kausalen Schließen werden Eigenschaften von Daten und Beobachtungen aus Kausalmodellen, d. h. Gesetzesannahmen von Ursachen und Wirkungen, abgeleitet. Kausales Schließen ermöglicht damit, die Wirkungen von Interventionen oder Datenveränderungen (z. B. durch Experimente) zu bestimmen.

Kausales Lernen versucht umgekehrt, ein Kausalmodell aus Beobachtungen, Messdaten und Interventionen (z. B. Experimente) abzuleiten, die zusätzliche Gesetzes- und Strukturannahmen voraussetzen.

In der Praxis verwenden neuronale Netze bereits Millionen von Elementen und Milliarden von synaptischen Verbindungen. Solche komplexen Systeme können zwar nach den Gesetzen der statistischen Physik untersucht werden, um globale Aussagen über Trend- und Konvergenzverhalten des gesamten Systems abzuleiten. Die Zahl der empirischen Parameter der einzelnen Elemente ist jedoch unter Umständen so groß, dass keine lokalen Ursachen ausgemacht werden können. Für Fragen der Verantwortlichkeit ist die Ursachenbestimmung jedoch zentral. Das neuronale Netz bleibt für uns eine »Black Box«. Vom ingenieurwissenschaftlichen Standpunkt aus sprechen Autorinnen und Autoren daher von einem »dunklen Geheimnis im Zentrum der KI des Machine Learning: »[...] even the engineers who designed [the machine learning-based system] may struggle to isolate the reason for any single action«¹¹.

Zwei verschiedene Ansätze im Software Engineering sind denkbar:

1. Testen zeigt nur (zufällig) gefundene Fehler, aber nicht alle anderen möglichen.
2. Zur grundsätzlichen Vermeidung müsste eine formale Verifikation des neuronalen Netzes und seiner zugrundeliegenden kausalen Abläufe durchgeführt werden.

Zusammengefasst folgt: Machine Learning mit digitalen neuronalen Netzen funktioniert, aber wir können die Abläufe in den neuronalen Netzen nicht im Einzelnen verstehen und kontrollieren. Heutige Techniken des Machine Learning beruhen meistens nur auf statistischem Lernen, aber das reicht nicht für sicherheitskritische Systeme. Daher sollte Machine Learning mit Beweisassistenten und kausalem Lernen verbunden werden.¹²

¹¹ Knight (2017).

¹² Mainzer (2020a).

1.3 Jenseits digitaler Berechenbarkeit und KI

1.3.1 Grenzen analoger Berechenbarkeit

Probleme, die nicht Turing-berechenbar sind, können auf technischen digitalen Computern nicht gelöst werden. Ein Grund ist die diskrete Struktur von theoretischen Turingmaschinen und technisch-praktischen digitalen Computern. Tatsächlich sind viele Probleme des Alltags kontinuierlicher (stetiger) Natur. So werden z. B. Druck- und Wärmeempfindungen durch stetige Zu- und Abnahme wahrgenommen. Viele Messgrößen werden durch stetige Messanzeichen registriert. Historische Beispiele sind das Ablesen der Uhrzeit auf dem Kreis eines Ziffernblatts oder astronomischer Abläufe in einem mechanischen Planetenmodell. In der Technik wurden elektrische, mechanische und hydraulische Signale durch stetige Größen modelliert. Wegen der Analogie der Modellgrößen zu den stetigen Signalen der Natur spricht man auch von analogen Modellgrößen. Analogrechner berechnen spezielle Abläufe in Natur und Technik in analogen Modellen.

Digitale Computer können stetige Größen nur approximativ durch diskrete Signale approximieren. Häufig reichen in der Praxis solche Approximationen aus. Seit den 1970er Jahren erwiesen sich digitale Computer für diese technisch-praktischen Zwecke als schneller und effektiver als Analogrechner. Der Siegeszug der Digitalrechner begann und die Analogrechner galten weitgehend als historisch überholt und nur noch für sehr spezielle Aufgaben sinnvoll. Bereits am Beginn des digitalen Zeitalters stellte allerdings John von Neumann als Konstrukteur des ersten programmgesteuerten digitalen Computers in der Hixon Symposium Lecture (1948) fest, dass digitale Computer bestenfalls Approximationen sind, aber nie die vollständige analoge Welt erfassen können.¹³

John von Neumann war nicht nur digitaler Computerpionier, sondern zugleich einer der brillanten mathematischen Grundlagentheoretiker des 20. Jahrhunderts in mathematischer Logik, Quantenmechanik, reeller und komplexer Analysis. Digitale Computer nach dem Vorbild der Turingmaschine berechnen nur endliche Größen wie ganze und rationale Zahlen (›Brüche‹). Reelle Zahlen werden mathematisch aber als Dezimalbuchentwicklungen durch unendliche Folgen von rationalen Zahlen definiert. Daher heißen reelle Zahlen Turing-berechenbar, wenn sie mit endlichen Abschnitten von (Turing-)berechenbaren Folgen rationaler Zahlen effektiv approximiert werden können.

Roger Penrose kritisierte diese Turing-berechenbare Analysis, weil in diesem Fall eine reelle (oder komplexe) Zahl Bit für Bit in eine Turing-Maschine eingegeben werden muss: Probleme entstehen, wenn man bei unendlichen Dezimalbruchentwicklungen wie 4,10327... in endlicher Zeit entscheiden will, ob zwei Zahlen gleich sind.¹⁴

Die Berechenbarkeit reeller und komplexer Analysis erfordert daher das Modell eines Analogrechners, der reelle Zahlen vollständig als analoge Größen erfasst

13 Neumann (1951).

14 Penrose (1991): 124.

und nicht nur in digitalen Approximationen.¹⁵ Theoretisch zeigt sich, dass auf dieser Grundlage Probleme entscheidbar werden, deren Berechenbarkeit auf digitaler Grundlage ausgeschlossen ist. Dabei handelt es sich z. B. um technisch-praktische Optimierungsprobleme und Probleme der Mustererkennung in der KI (*>inverse Probleme<*), deren Lösbarkeit an der Beschränkung der digitalen Turing-Berechenbarkeit scheitert. Die Aufgabe besteht also darin,

1. theoretisch eine mathematische Theorie analoger Berechenbarkeit für die reelle Analysis zu entwickeln, auf deren Grundlage dieser Probleme lösbar, entscheidbar und berechenbar werden,
2. technisch-praktisch eine Realisierung reeller und komplexer Berechenbarkeit in der Hardware passender Analogrechner zu entwickeln.

In dieser Weise lässt sich auch die Entscheidbarkeit und Berechenbarkeit von (reellen) Differentialgleichungen für Anwendungen in Natur-, Technik- und Wirtschaftswissenschaften bestimmen.¹⁶ Ein grundlegender Unterschied der Entscheidbarkeit für digitale und reelle Anwendungen war theoretisch seit der Mitte des letzten Jahrhunderts durch zentrale Beweise der Logiker Kurt Gödel und Alfred Tarski bekannt. Während Gödel bereits 1931 zeigte, dass es über den ganzen Zahlen definierbare Mengen gibt, die nicht entscheidbar sind, bewies Tarski 1951 allgemein, dass jede über den reellen Zahlen definierbare Menge auch entscheidbar (in beschränkter Zeit) ist.

Ein grundlegender Unterschied in der digitalen und analogen (reellen) Welt zeigt sich in den Anwendungen der Künstlichen Intelligenz.¹⁷ Bereits 1956 bewies der amerikanische Logiker und Mathematiker S.C. Kleene, dass einfache neuronale Netze (McCulloch-Pitts Netze) mit ganzen Zahlen als synaptischen Gewichten nur dieselbe Klasse von einfachen (regulären) formalen Sprachen erkennen kann wie endliche Automaten. Demgegenüber erkennen (rekurrente) neuronale Netze mit rationalen Zahlen als synaptischen Gewichten dieselbe Klasse von Sprachen wie Turingmaschinen, nämlich Sprachen auf der Grundlage von Chomsky-Grammatiken. Erweitert man die Gewichte neuronaler Netze auf reelle Zahlen, erhält man analoge Netze. Sie erkennen dieselbe Klasse von Sprachen wie entsprechend um Orakel erweiterte Turing-Maschinen in polynomialer Zeit jenseits der Turing-Berechenbarkeit.¹⁸

Bei praktischen Anwendungen in der Medizin zeigt Machine Learning auf analoger Grundlage ebenfalls ein erheblich größeres Potential als auf digitaler Grundlage. Ein klassisches Beispiel sind die inversen Probleme bei der medizinischen Bildgebung. Invers besagt hier, dass z. B. bei der medizinischen Bildgebung durch MRI (magnetic resonance imaging) oder CT (computed tomography) das abgebildete Organ auf ana-

¹⁵ Mainzer (2018).

¹⁶ Blum/Shub/Smale (1989).

¹⁷ Mainzer (2018): 241ff.

¹⁸ Siegelmann/Sontag (1994).