

Neuronale Netze selbst programmieren

Ein verständlicher Einstieg mit Python

» Hier geht's
direkt
zum Buch

DAS VORWORT

Die Suche nach intelligenten Maschinen

Seit Tausenden von Jahren versuchen Menschen zu verstehen, wie Denkprozesse funktionieren und wie sie sich mit irgendwelchen Maschinen nachbilden lassen – als Denkmaschinen.

Wir geben uns nicht zufrieden mit mechanischen oder elektronischen Maschinen, die uns bei einfachen Arbeiten unter die Arme greifen – Feuersteine, die Feuer anzünden, Flaschenzüge, die schwere Steine heben, und Taschenrechner, die vorgegebene Rechenaufgaben lösen.

Stattdessen möchten wir anspruchsvollere und komplexere Aufgaben automatisieren, beispielsweise ähnliche Fotografien klassifizieren, kranke von gesunden Zellen unterscheiden oder sogar ein gepflegtes Schachspiel erleben. Solche Aufgaben verlangen anscheinend nach menschlicher Intelligenz oder zumindest nach einer geheimnisvollen, tiefer verborgenen Fähigkeit des menschlichen Gehirns, die man in einfachen Maschinen wie Taschenrechnern nicht findet.

Die Idee einer Maschine mit dieser menschenähnlichen Intelligenz ist so verlockend und übermächtig, dass unsere Kultur viele Fantasien und Ängste dazu entwickelt hat – man denke an den äußerst leistungsstarken, doch letztendlich bedrohlichen HAL 9000 aus dem Film *2001: Odyssee im Weltraum* von Stanley Kubrick, die verrückten *Terminator*-Roboter und das sprechende KITT-Auto mit cooler Persönlichkeit aus der klassischen TV-Serie *Knight Rider*.

Als Gary Kasparov, der damals amtierende Schachweltmeister und Großmeister, 1997 durch den IBM-Computer Deep Blue geschlagen wurde, fürchteten wir das Potenzial der Maschinenintelligenz genau so, wie wir diese historische Errungenschaft gefeiert hatten.

Unser Verlangen nach intelligenten Maschinen ist so stark, dass schon so mancher der Versuchung erlegen war, eine solche Maschine vorzutäuschen. Im berühmigten Schachtürken, einer mechanischen Schachmaschine, war lediglich eine Person im Inneren eines Schanks versteckt!

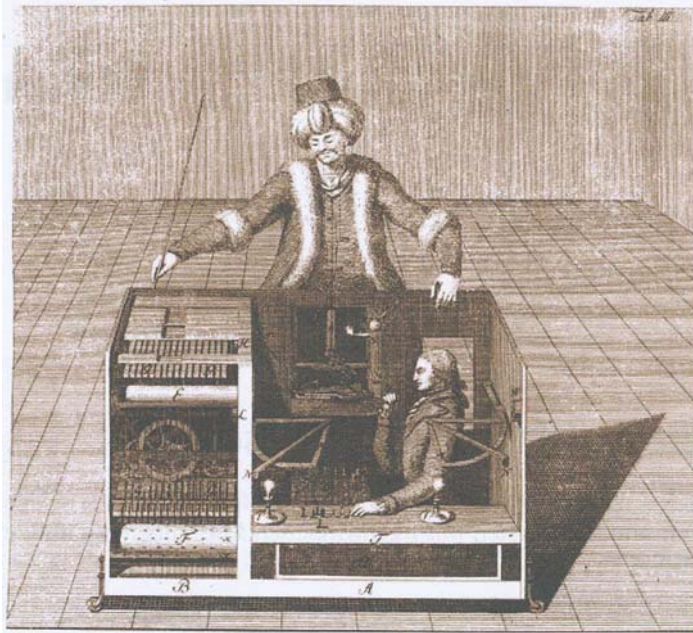


Abbildung E-1: Der Schachtürke – eine »Maschine«, die Schach spielen konnte

Ein neues goldenes Zeitalter – von der Natur inspiriert

Optimismus und Ambitionen in Bezug auf künstliche Intelligenz hatten in den 1950er-Jahren einen Höhenflug, als das Thema formalisiert wurde. Erste Erfolge zeigten sich bei Computern, die einfache Spiele beherrschten und Lehrsätze beweisen konnten. Manche waren davon überzeugt, dass innerhalb eines Jahrzehnts Maschinen erscheinen würden, die das Niveau menschlicher Intelligenz erreichten.

Doch die künstliche Intelligenz erwies sich als harter Brocken, und der Fortschritt stagnierte. Die 1970er-Jahre standen unter schlechten Vorzeichen in Bezug auf künstliche Intelligenz, begleitet von Mittelkürzungen und zurückgehendem Interesse.

Es schien so, als würden Maschinen aus kalter, fest verdrahteter Logik, die mit absoluten Einsen und Nullen arbeiten, niemals in der Lage sein, die nuancenreichen organischen, manchmal unscharfen Denkprozesse von biologischen Gehirnen zu erreichen.

Nach einer Periode ohne wesentliche Fortschritte tauchte eine sehr vielversprechende Idee auf, um die Suche nach Maschinenintelligenz aus ihrer Starre zu befreien. Warum versucht man nicht, künstliche Gehirne zu bauen, indem man kopiert, wie reale biologische Gehirne arbeiten? Reale Gehirne mit Neuronen an-

stelle von Logikgattern, weichere biodynamische Schlussfolgerungen anstelle von kalten, harten, schwarz-weißen, absolutistischen herkömmlichen Algorithmen.

Die Wissenschaftler wurden inspiriert von der scheinbaren Einfachheit eines Bienen- oder Taubengehirns verglichen mit den komplexen Aufgaben, die sie bewältigen konnten. Gehirne, die nur einen Bruchteil eines Gramms wiegen, sind offenbar zu komplizierten Aktionen fähig, zum Beispiel zur Flugsteuerung und Anpassung an den Wind, dazu, Nahrung und Beutetiere zu identifizieren und schnell zu entscheiden, ob gekämpft oder geflüchtet werden muss. Könnten Computer, die jetzt über massive billige Ressourcen verfügen, diese Gehirne nachbilden und verbessern? Eine Biene hat etwa 950.000 Neuronen – könnten heutige Computer mit Speicherkapazitäten im Gigabyte- und Terabyte-Bereich die Bienen übertreffen?

Doch mit den herkömmlichen Konzepten zur Problemlösung können diese Computer trotz der massiven Speicherkapazitäten und superschnellen Prozessoren nicht das erreichen, wozu die relativ winzigen Gehirne von Vögeln und Bienen in der Lage sind.

Neuronale Netze haben sich aus diesem Drang nach biologisch inspiriertem intelligentem Computing herausgebildet – und sind in der Folge zu den leistungsfähigsten und nützlichsten Methoden auf dem Gebiet der künstlichen Intelligenz geworden. Heute ist Google DeepMind zu fantastischen Dingen fähig, beispielsweise von sich aus zu lernen, wie Videospiele gespielt werden, und zum ersten Mal wurde der Weltmeister im unglaublich komplexen Spiel Go geschlagen. Als wesentlicher Bestandteil der Architektur dienen neuronale Netze. Neuronale Netze bilden bereits den Kern vieler Alltagstechnologien – wie der automatischen Nummernschilderkennung und der Decodierung von handschriftlichen Postleitzahlen auf handgeschriebenen Briefen.

Dieses Buch erläutert, was neuronale Netze sind, wie sie funktionieren und wie Sie eigene neuronale Netze erstellen können, die sich für die Erkennung von handgeschriebenen Zeichen trainieren lassen. Eine solche Aufgabe ist mit herkömmlichen Ansätzen der Rechentechnik äußerst schwer zu realisieren.

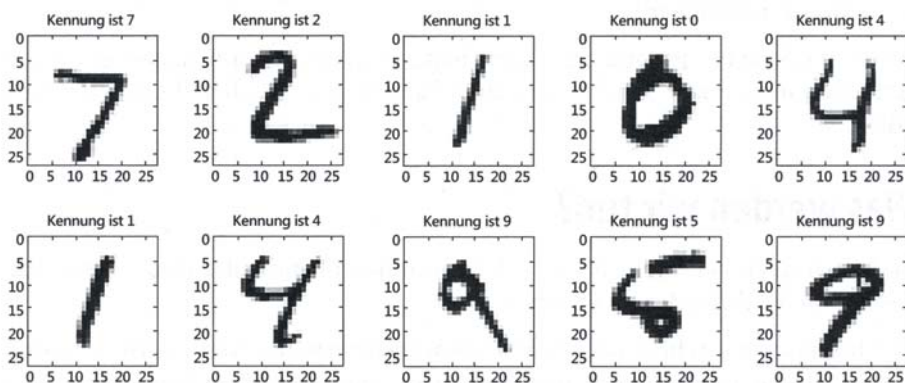


Abbildung E-2: Handgeschriebene Ziffern, die automatisch erkannt werden sollen

An wen richtet sich dieses Buch?

Dieses Buch ist für alle gedacht, die an neuronalen Netzen interessiert sind und eigene Netze erstellen und einsetzen möchten. Es richtet sich an diejenigen, die sich mit den ziemlich einfachen, aber doch spannenden mathematischen Ideen, die den Kern der Arbeitsweise von neuronalen Netzen bilden, auseinandersetzen wollen.

Leserinnen und Leser müssen keine Experten der Mathematik oder Informatik sein. Spezielle Kenntnisse oder mathematische Fähigkeiten, die über die Schulmathematik hinausgehen, sind nicht erforderlich. Um Ihre eigenen neuronalen Netze zu erstellen, kommen Sie mit den vier Grundrechenarten aus. Die schwierigste Operation ist die Gradientenanalyse – doch selbst dieses Konzept wird so erklärt, dass es von den meisten verstanden werden kann.

Interessierten Lesern oder Studentinnen kann dieses Buch als Ausgangspunkt dienen, um spannende Exkursionen in die Welt der künstlichen Intelligenz zu unternehmen. Hat man einmal die Grundprinzipien neuronaler Netze verinnerlicht, kann man die Kernideen auf viele unterschiedliche Probleme anwenden.

Dozentinnen und Dozenten können dieses Buch als leicht verständliche Erläuterung neuronaler Netze und ihrer Implementierung nutzen, um Studentinnen und Studenten zu begeistern und anzuregen, künstliche Intelligenz mit nur wenigen Codezeilen einer Programmiersprache selbst zu erkunden. Die Funktionsfähigkeit des Codes wurde getestet mit einem Raspberry Pi, einem kleinen und preiswerten Computer, der in Schulen und im Studium sehr beliebt ist.

Schön wäre es gewesen, wenn es zu meiner Zeit ein solches Buch gegeben hätte, als ich als Teenager mühevoll herausfinden wollte, wie diese leistungsfähigen und dennoch geheimnisvollen neuronalen Netze funktionieren. Zwar konnte man ihnen in Büchern, Filmen und Zeitschriften begegnen, doch zu jener Zeit konnte ich nur schwierige akademische Artikel finden, die sich an Leute richteten, die schon Mathematiker waren oder sich zumindest in der mathematischen Ausdrucksweise auskannten.

Wenn mir doch nur jemand das Thema hätte so erläutern können, dass es ein einigermaßen neugieriger Schüler verstehen kann! Dieses Buch soll das jetzt nachholen.

Was werden wir tun?

In diesem Buch begeben wir uns auf Tour, um ein neuronales Netz zu erstellen, das handschriftliche Ziffern erkennen kann.

Wir beginnen mit sehr einfachen Vorhersageneuronen und verbessern sie sukzessive, wenn wir an ihre Grenzen stoßen. Dabei legen wir kurze Pausen ein, wenn

erforderliche mathematische Konzepte eingeführt werden, um zu verstehen, wie neuronale Netze lernen und Problemlösungen vorhersagen können.

Auf unserer Tour begegnen wir vielen mathematischen Konzepten. Dazu gehören zum Beispiel Funktionen, einfache lineare Klassifizierer, iterative Verfeinerung, Matrixmultiplikation, Gradientenrechnung, Optimierung nach dem Gradientenverfahren und sogar geometrische Drehungen. Alle diese Konzepte werden aber wirklich leicht verständlich erläutert, und über einfache Schulmathematik hinaus sind weder Vorkenntnisse noch fachliche Erfahrungen erforderlich.

Nach dem erfolgreichen Aufbau unseres ersten neuronalen Netzes entwickeln wir es in verschiedene Richtungen weiter. So verwenden wir Bildverarbeitung, um unser maschinelles Lernen zu verbessern, ohne auf zusätzliche Trainingsdaten zurückzugreifen. Wir werfen sogar einen kurzen Blick in das Gedächtnis unseres neuronalen Netzes, um zu erfahren, ob es etwas von seinen inneren Lernmechanismen preisgibt – etwas, das viele Bücher nicht zeigen!

Außerdem werden wir Python, eine einfache, nützliche und beliebte Programmiersprache, kennenlernen, wenn wir unser neuronales Netz schrittweise aufbauen. Auch hierfür werden keine vorhandenen Programmierkenntnisse vorausgesetzt oder benötigt.

Wie gehen wir vor?

Dieses Buch hat vor allem das Ziel, die Konzepte hinter neuronalen Netzen möglichst vielen Menschen zugänglich zu machen. Das bedeutet, dass wir bei einem neuen Gedanken immer von etwas Bekanntem ausgehen und es in kleinen, einfachen Schritten weiterentwickeln. Auf diese Weise gelangen wir zu einem Punkt, an dem wir interessante und spannende Erkenntnisse über die neuronalen Netze mitnehmen können.

Um den Faden nicht zu verlieren, widerstehen wir der Versuchung, alles zu diskutieren, was nicht unbedingt erforderlich ist, um ein eigenes neuronales Netz zu erstellen. Es wird aber auf interessante Zusammenhänge und Berührungspunkte hingewiesen. Wenn Sie sich näher damit befassen möchten, nur zu!

Dieses Buch betrachtet nicht alle möglichen Optimierungen und Verfeinerungen an neuronalen Netzen. Es gibt viele davon, doch sie würden hier vom eigentlichen Zweck ablenken – nämlich die wesentlichen Ideen in möglichst einfacher und überschaubarer Weise vorzustellen.

Gegliedert ist dieses Buch in vier Kapitel:

- In *Kapitel 1* gehen wir behutsam die mathematischen Konzepte durch, die in einfachen neuronalen Netzen wirken. Wir verzichten zunächst bewusst auf Computerprogramme, um nicht von den Kerngedanken abzulenken.

- In *Kapitel 2* lernen wir gerade so viel Python, um unser eigenes neuronales Netz implementieren zu können. Wir trainieren es darauf, handgeschriebene Ziffern zu erkennen, und wir testen seine Erkennungsleistung.
- In *Kapitel 3* gehen wir weiter, als es für das Verständnis einfacher neuronaler Netze erforderlich ist – just for fun. Wir probieren Ideen aus, um die Leistung unseres neuronalen Netzes weiter zu steigern, und wir werfen auch einen Blick in das Innere eines trainierten Netzes, um zu erkunden, wie das Netz intern lernt und Entscheidungen für die Ausgabe trifft.
- In *Kapitel 4* überführen wir unser neuronales Netz in ein typisches professionelles Szenario, indem wir es mithilfe von PyTorch erstellen, einem in der Branche führenden Framework. Wir sammeln einige praktische Erfahrungen mit der Funktionalität, die uns das Framework PyTorch bietet.

Und keine Angst: Alle hier verwendeten Softwaretools sind *kostenlos* und *Open Source*, sodass Sie dafür nichts bezahlen müssen. Zudem brauchen Sie keinen teuren Computer, um eigene neuronale Netze zu erstellen. Der gesamte Code in diesem Buch ist auf einem sehr preiswerten (ca. 5 Euro teuren) Raspberry Pi Zero getestet worden.

Anmerkung des Autors

Ich hätte mein Ziel verfehlt, wenn ich Ihnen keinen Eindruck davon vermittelt hätte, wie spannend und überraschend Mathematik und Informatik sein können.

Ich hätte versagt, wenn ich Ihnen nicht gezeigt hätte, wie unglaublich leistungsstark Schulmathematik und einfache Computerrezepte sein können – indem man mit eigener künstlicher Intelligenz die Lernfähigkeiten menschlicher Gehirne nachbildet.

Ich hätte versagt, wenn ich Ihnen nicht die Zuversicht und den Wunsch mitgegeben hätte, das unglaublich umfangreiche Gebiet der künstlichen Intelligenz weiter zu erkunden.

Ihr Feedback zu diesem Buch ist mir sehr willkommen. Schreiben Sie dazu den Verlag O'REILLY unter *kommentar@oreilly.de* an.

Den im Buch verwendeten Code finden Sie auf GitHub unter:

<https://github.com/makeyourownneuralnetwork/makeyourownneuralnetwork>