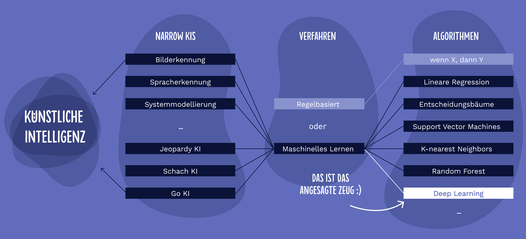
Deep Learning einfach gemacht

Die Komponenten eines Deep-Learning-Systems und was diese verdammten nerdigen Buzzwords eigentlich bedeuten.

Von Iskender Dirik

Künstliche Intelligenz, Maschinelles Lernen, Deep Learning, Katzenvideos. Das kann ganz schön überwältigend sein – wie ein Abendessen mit Deinen Eltern. Am besten zeichnen wir erst mal eine Skizze.



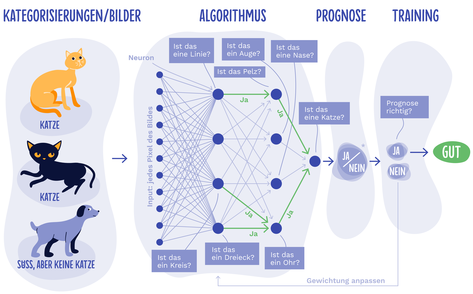
Was ist das eigentlich – KI (und vor allem das I in diesem Kontext)? Es ist nichts weiter als eine Reihe von „Narrow KIs", die kleinere Teilprobleme lösen können. Wie zum Beispiel Jeopardy oder Go spielen oder Bilder von Tieren erkennen. Erinnerst Du Dich an den Typen, der in der Schule neben Dir saß? Der konnte zwar den Zauberwürfel lösen, war dafür aber kein so guter Fußballer. Für die Lösung derart spezifischer, enger Problemstellungen haben wir bislang immer regelbasierte Verfahren genutzt – wenn der andere Schachspieler einen Turm weniger hat, dann versuchst Du, ihm noch mehr Figuren abzunehmen. Heute nutzen wir dafür Algorithmen für maschinelles Lernen. Dazu gehört Deep Learning, das extrem leistungsfähig für Probleme wie Bilderkennung oder die Verarbeitung natürlicher Sprache ist.

Das Wichtigste zuerst: **Die Grundlagen von Deep Learning**

Deep Learning basiert auf künstlichen neuronalen Netzen (KNN) mit mehr als zwei Schichten. Eines der ersten „tiefen" neuronalen Netze besaß drei Zwischenschichten oder „hidden layers" (Veröffentlichung von Hinton und Kollegen im Jahr 2006). „Sehr tiefe" Netze (beispielsweise VGG, der ImageNet-Sieger von 2014) bestehen aus über 16 Zwischenschichten. Ein KNN ist ein von der Biologie inspiriertes Computersystem, das dem menschlichen Gehirn und Nervensystem nachempfunden ist.

Ein neuronales Netz besteht aus unterschiedlichen Abstraktionsschichten. Jede Schicht wiederum besteht aus Knoten, die unterschiedliche Fragen stellen. Je früher die Schicht, desto abstrakter und einfacher die Fragen/Aufgaben. Der Output eines Knotens verrät uns, wie sicher er sich in Bezug auf die Frage ist. Diese Information dient als Input für die nächste Schicht – ein System, das vom Aufbau des menschlichen Gehirns inspiriert ist.

Die Knoten der ersten Schicht nehmen alle Eingabedaten auf, beispielsweise die Pixel von Bildern bei der Bildklassifizierung. Die erste Abstraktionsschicht versucht, Konzepte wie Formen – Dreiecke, Kreise, Linien – zu erkennen. Die nächste Schicht stellt Fragen wie: „Ist dieses Dreieck ein Ohr?" Die letzte Schicht wird als Ausgabeschicht bezeichnet, nimmt all diese Informationen und stellt schließlich die Frage, ob dieses pelzige Etwas mit zwei spitzen Ohren eine Katze ist.



*Hallo, lieber Besserwisser! Für die KI-Experten unter Euch:

1. Dieses Whitepaper wurde nicht für Euch geschrieben!
2. Ja, mir ist schon klar, dass neuronale Netze nie ein eindeutiges „Ja" oder „Nein" als Antwort geben; vielmehr prognostizieren sie, dass es sich z.B. mit 99 % Wahrscheinlichkeit um eine Katze handelt – und mit 1 % eben nicht. Es handelt sich bei meiner Darstellung um ein vereinfachtes Modell. Ende der Durchsage!
3. Ja, ich weiß, dass sich die obenstehende Beschreibung besonders auf Convolutional Neural Networks (CNN) bezieht. Der Leser weiß aber bislang noch nicht einmal von der Existenz von CNNs - das ändern wir später noch. Okay?
4. Ja, ich weiß, dass die ersten Schichten von CNNs versuchen, Kanten zu erkennen und keine geschlossenen Formen wie Dreiecke, Kreise usw. Wir sind hier aber nicht im Forschungslabor - ich will die Dinge für fachfremde Leser so plakativ und verständlich wie möglich schildern. Verstanden?

So weit, so gut. Du bist soweit, Dein erstes Deep-Learning-System zu bauen (Wirklich! Trau Dich!). Fangen wir also an. Keine Angst, es tut nicht weh.

1 Eingabedaten

Als erstes brauchen wir Eingabedaten. Die Genauigkeit eines Deep-Learning-Systems hängt ganz wesentlich von den Eingabedaten ab. Du weißt bereits: Wenn man Blödsinn eingibt, kommt auch Blödsinn raus. Also musst Du Daten sammeln. Und zwar so viele wie möglich. Von externen und/oder internen Quellen.

Diese Bilder sind unsere Eingabedaten. Wir sind verrückt nach Katzen, also lass uns ein System bauen, das Katzen erkennt! Katzenerkennung – das ist doch wirklich mal ein kreatives Beispiel für KI und Deep Learning, oder?

2 Datenkategorisierung

Für die meisten Deep-Learning-Aufgaben müssen wir Daten kategorisieren. Und zwar von Hand. Ja, das ist verdammt harte Arbeit. Du kannst auch öffentlich verfügbare, kategorisierte Datenquellen nutzen (es gibt riesige Bild- oder Klangbibliotheken etc.). Allerdings kann darauf jeder zugreifen, sodass Du keinen Wettbewerbsvorteil daraus ziehen kannst, Faulpelz.

3 Lernkriterien

Und jetzt zum spaßigen Teil: Dem Aufbau des Systems. Fangen wir damit an: Was ist der Zweck unseres Deep-Learning-Systems, nach welchem grundlegenden Verfahren gehen wir vor?

A. SUPERVISED LEARNING

BESCHREIBUNG

Beim supervised Learning, also dem überwachten Lernen, haben wir es mit einer Menge von Daten zu tun, die bereits von Menschen kategorisiert bzw. verschlagwortet wurden. Es wird hierbei versucht, einen Algorithmus zu finden, der die Kategorisierungen bzw. Verschlagwortungen neuer Eingabedaten prognostiziert. Dabei geht es um das Erlernen einer Zuordnung von Eingabe- zu Ausgabedaten mithilfe eines einfachen, wunderschönen Algorithmus namens Backpropagation („Rückpropagierung“).

EINSATZBEREICH

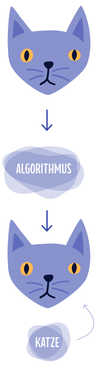
Etwas mittels Eingabedaten klassifizieren/prognostizieren.

TYPISCHE ANWENDUNGEN

- Erkennen einer Katze in einem Bild
- Erkennen von Krebs anhand von Röntgenbildern
- Voraussagen

BEISPIEL

Frage Ist das eine Katze?
Eingabe Bild
Ausgabe Ja/Nein



B. REINFORCEMENT LEARNING

BESCHREIBUNG

Bei einem Reinforcement-Problem, also einem „Bestärkungsproblem“, haben wir eine gegebene Ausgangssituation und können zwischen verschiedenen Aktionen wählen, deren zukünftiges Feedback unbekannt ist.

EINSATZBEREICH

Maximierung zukünftiger Belohnungen.

TYPISCHE ANWENDUNGEN

- Robotern beibringen, eine Kiste zu heben
- Hubschraubern beibringen, stabil zu fliegen
- Schach, Go oder Tic-Tac-Toe spielen

BEISPIEL

Frage Welche Maßnahme maximiert angesichts des aktuellen Status zukünftige Belohnungen für den Agenten?
Eingabe Status
Ausgabe Maßnahme

C. UNSUPERVISED LEARNING

BESCHREIBUNG

Bei unsupervised Learning, also unüberwachtem Lernen, sind lediglich Daten verfügbar, die nicht kategorisiert bzw. verschlagwortet wurden.

EINSATZBEREICH

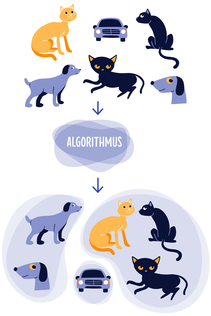
Ermitteln verborgener/inhärenter Strukturen eines Problems.

TYPISCHE ANWENDUNGEN

- Gemeinsames Gruppieren von Katzen- und Hundebildern
- Erkennung von Anomalien
- Ermitteln von Zusammenhängen zwischen Wörtern in einer Sprache

BEISPIEL

Frage Welche Gegenstände sind ähnlich?
Eingabe Bilder
Ausgabe Cluster ähnlicher Bilder



Ja, ja, Du Klugscheißer. Unsupervised Learning macht nur Sinn, wenn du Schritt 2 ausgelassen hast, d.h. deine Eingabedaten nicht manuell kategorisiert bzw. verschlagwortet hast.
Aber willst Du drei Millionen Katzen selbst kategorisieren?

Unsupervised Learning ist nach wie vor kommerziell irrelevant. Ja, Google findet damit Katzen, aber dafür haben sie 16.000 Computerprozessoren mit einer Milliarde Verbindungen gebraucht und mussten YouTube komplett durchforsten. Im Bereich des Deep Learnings funktioniert überwachtes Lernen kommerziell, unüberwachtes (bislang) noch nicht.

Exkurs: Die Zukunft des Deep Learning

Gegenwart: Heute nutzen wir in erster Linie überwachtes Lernen, was im Wesentlichen Imitationslernen ist. Menschen kategorisieren Bilder, Videos, Text oder Sprache und das neuronale Netz versucht nachzuvollziehen, was die Menschen tun.

Zukunft: Im Laufe der Zeit, wenn neuronale Netze „fähiger“ werden (also ziemlich gut im Supervised Learning und Imitieren von Verhalten), werden Unternehmen zusätzlich Reinforcement Learning einsetzen, um neuronalen Netzen tatsächliche Ziele zu setzen - also nicht nur den Menschen hinter den Kulissen nachzuahmen, sondern selbst Ziele zu erreichen wie:

- E-Commerce: Das Modell lernt das Kundenverhalten und richtet die Dienstleistung am Kundeninteresse aus.
- Finanzen: Das Modell lernt das Marktverhalten und erstellt Handelsstrategien.
- Roboter: Das Modell lernt (mittels Video), wie sich die physische Welt verhält, und navigiert dann in dieser Welt.

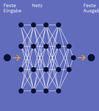
Um all das zu erreichen, bedarf es einer Menge Imitationslernen am Vorbild des Menschen.

4 Schichtarchitekturen

Unterschiedliche Architekturen von neuronalen Netzen funktionieren gut mit verschiedenen Datentypen. Für das Lösen von KI-Problemen mit optischen, Audio- oder Textdaten müssen wir den richtigen Schaltplan nutzen (das soll nur eine Metapher sein). Das ist die übergeordnete Struktur unseres Deep-Learning-Systems. Jetzt wird es ein bisschen mehr wie Raketenphysik. Bei neuronalen Netzen gibt es viele unterschiedliche Schichtarchitekturen, wobei die verbreitetsten die folgenden drei sind:

A. CONVOLUTIONAL NEURAL NETWORK (CNN OBER CONVNET)

SCHALTPLAN



BESCHREIBUNG

Das Netz lernt zunächst, Kanten zu erkennen und nutzt diese dann zum Erkennen einfacher Formen. Anschließend nutzt es diese Formen für das Erkennen weiterer übergeordneter Merkmale wie Gesichtsformen usw. Im Wesentlichen wird hierbei der komplizierte Prozess der Bildklassifizierung in eine Reihe einfacherer Schritte unterteilt, die jeweils von einer anderen Schicht des Modells beschrieben werden.

EINGAFTJBEREICH

Wenn es sich bei der Eingabe um eine Bild- oder Audiodatei handelt und die Aufgabe das Erkennen von Mustern darin ist, sind Convolutions („Faltungen") möglicherweise genau das Richtige.

TYPISCHE ANWENDUNG

- Katzenerkennung in Bildern
- Wahrnehmung der Umgebung beim autonomen Fahren
- Empfehlung eines Liedes, das Dir gefallen könnte

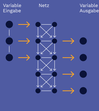
BEISPIEL

Aufgabe Erkennung
Eingabe Bild
Ausgabe Ja/Nein

Jetzt solltest Du es langsam verstanden haben, oder?
Na gut, also ... nehmen wir ein Bild mit, sagen wir, 360 Pixeln. Jedes Pixel hat einen Wert. Dieser Wert wird ins neuronale Netz eingegeben, weitergeleitet und schließlich gibt ein einziges Neuron Auskunft darüber, ob es sich bei der Eingabe um ein Kätzchen handelt oder nicht. Der Trick ist, dass alle Bilder genau gleich groß sein müssen.

B. SEQUENZMODELLE

SCHALTPLAN



BESCHREIBUNG

Eine Reihe von Daten werden in das Netz eingegeben und erzeugen eine Ausgabe für so viele Schritte wie gewünscht. Für die echten Pros: Bei den sogenannten „Neurcent Neurul Networks" (RNN), „Long short-term memory" (LSTM, etwa „Langes Kurzzeitgedächtnis") und „Recursive Networks" handelt es sich um Sequenzmodelle.

EINGAFTJBEREICH

Bei sequentiellen Daten ist die Zeit der Hauptunterscheidungsfaktor zwischen den Elementen einer Sequenz. Beispiele hierfür sind Texte, Sprache oder Videos.

TYPISCHE ANWENDUNG

- Verstehen natürlicher Sprache
- Übersetzung
- Verstehen von Videos

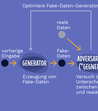
BEISPIEL

Aufgabe Übersetzung
Eingabe „Eine Katze frisst Kartoffeln"
Ausgabe Ich chat mange des pommes de terre.

Nehmen wir an, Du versuchst, Sätze zu übersetzen. Zunächst bestehen unterschiedliche Sätze aus unterschiedlich vielen Wörtern. Zweitens kann ein einzelnes Wort wie „Kartoffeln" im Französischen aus vier Wörtern bestehen (des pommes des terre – Äpfel aus der Erde). Oh Mann, die Franzosen – allerie, wenn mich jemand auf Französisch nach dem Weg zum Flughafen fragt, geht mir schon das Herz auf. Und genau hier sind die Sequenz-zu-Sequenz-Modelle die Rettung: Für die Übersetzung natürlich, nicht für mein Herz.

C. GENERATIVE ADVERSARIAL NETWORK (GAN)

SCHALTPLAN



BESCHREIBUNG

GANs sind eine Technik zur Erzeugung synthetischer Bilder/Videos, die für einen menschlichen Betrachter authentisch auseehen. Das Ziel ist es, den Betrachter dazu zu bringen, das künstliche Foto als ein echtes Bild zu akzeptieren. GANs sind aktuell der große Hype in der Deep-Learning-Community.

EINGAFTJBEREICH

Wenn Du selbst Daten, z.B. Bilder, erzeugen willst.

TYPISCHE ANWENDUNG

- Erstellen fotorealistischer Bilder aus Texten oder anderen Bildern
- Verbesserung der Auflösung von Fotos
- Voraussagen des nächsten Frames (Einzelbild) in einem Video

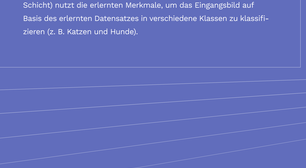
BEISPIEL

Aufgabe Generierung von Gesichtern
Eingabe keine Eingabe erforderlich
Ausgabe Bilder von Gesichten

Diese Gesichter wurden tatsächlich von einem Computer generiert.
Ja, verdammt. Ich kann es selbst kaum glauben. Mach Dich bereit für die Matrix, mein Freund.
Das geschieht, wenn ein neuronales Netz versucht, Fake-Gesichter zu erzeugen, um damit ein zweites neuronales Netz in die Irre zu führen.

Exkurs für Pros: CNN im Detail
Fühl Dich nicht wie ein Versager, wenn Du das hier überspringst.

Wie ich gehört habe, bist Du fasziniert von Convolutional Neural Networks (CNNs) und ihren Bilderkennungsfähigkeiten. Dann lass uns weiter ins Detail gehen:



Ein CNN ist grundsätzlich eine Eingabe-Ausgabe-Zuordnung. Es wandelt sensorische Rohdaten auf der Eingabeseite (z.B. ein Bild) in eine Klassifizierung bzw. Verschlagwortung um (z.B. eine Textbeschreibung des im Bild dargestellten Gegenstands). Die Eingabedaten werden in das Netz eingegeben, wo anschließend eine Reihe von Zwischenschichten im Netz zunehmend abstraktere „Merkmale" des Bildes extrahieren. Ein Merkmal ist ein Attribut, das das Netz aus den Daten „lernt".

Fühlst du dich stark genug für noch mehr technische Beschreibungen? Okay, Du hast es so gewollt. Bei einem klassischen CNN gibt es vier Hauptoperationen: Convolution-Operationen, ReLU-Operationen, Pooling-Operationen und, zu guter Letzt, Klassifizierungsoperationen:

1. Convolution-Operationen: Der Hauptzweck eines Convolution- oder Faltungsschritts ist es, Merkmale aus den Eingabedaten zu extrahieren (z. B. Erkennen einer Kante). Während dieses Vorgangs wird das Bild von unterschiedlichen Merkmal-Detektoren gescannt, wodurch eine „Merkmalskarte" (z. B. Zahlenmatrizen) entsteht.

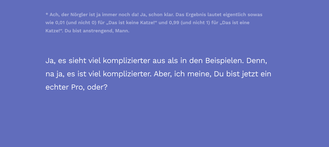
2. Rectified Linear Units (ReLU)-Operationen: Eine Convolution ist eine lineare Operation. Die meisten Daten in der realen Welt sind allerdings nicht linear. Der Zweck einer ReLU-Operation ist es, diese Nichtlinearität in das CNN einzubinden.

3. Pooling-Operationen: Die Pooling-Operation reduziert die Dimensionalität jeder einzelnen „Merkmalskarte", behält aber die wichtigsten Informationen bei. Dadurch reduziert sich auch die Größe der Eingabeinformationen, sodass sie besser handhabbar sind. Je nach Anzahl der Schichten im Netz finden mehrere Convolution- und Pooling-Schritte statt, bis die letzte Schicht im Netz erreicht ist.

4. Klassifizierungsoperation: Die Klassifizierungsoperation (letzte Schicht) nutzt die erlernten Merkmale, um das Eingangsbild auf Basis des erlernten Datensatzes in verschiedene Klassen zu klassifizieren (z. B. Katzen und Hunde).

Lernbereites System: **Eine Netzarchitektur für die Erkennung von Gegenständen in Bildern.**

Yeah, wir haben gerade ein Deep-Learning-System gebaut. Gehen wir der Einfachheit halber mal davon aus, dass es sich um ein **Supervised Learning System** mit **Convolution-Schichten** für die Identifizierung von Katzen in Bildern handelt.

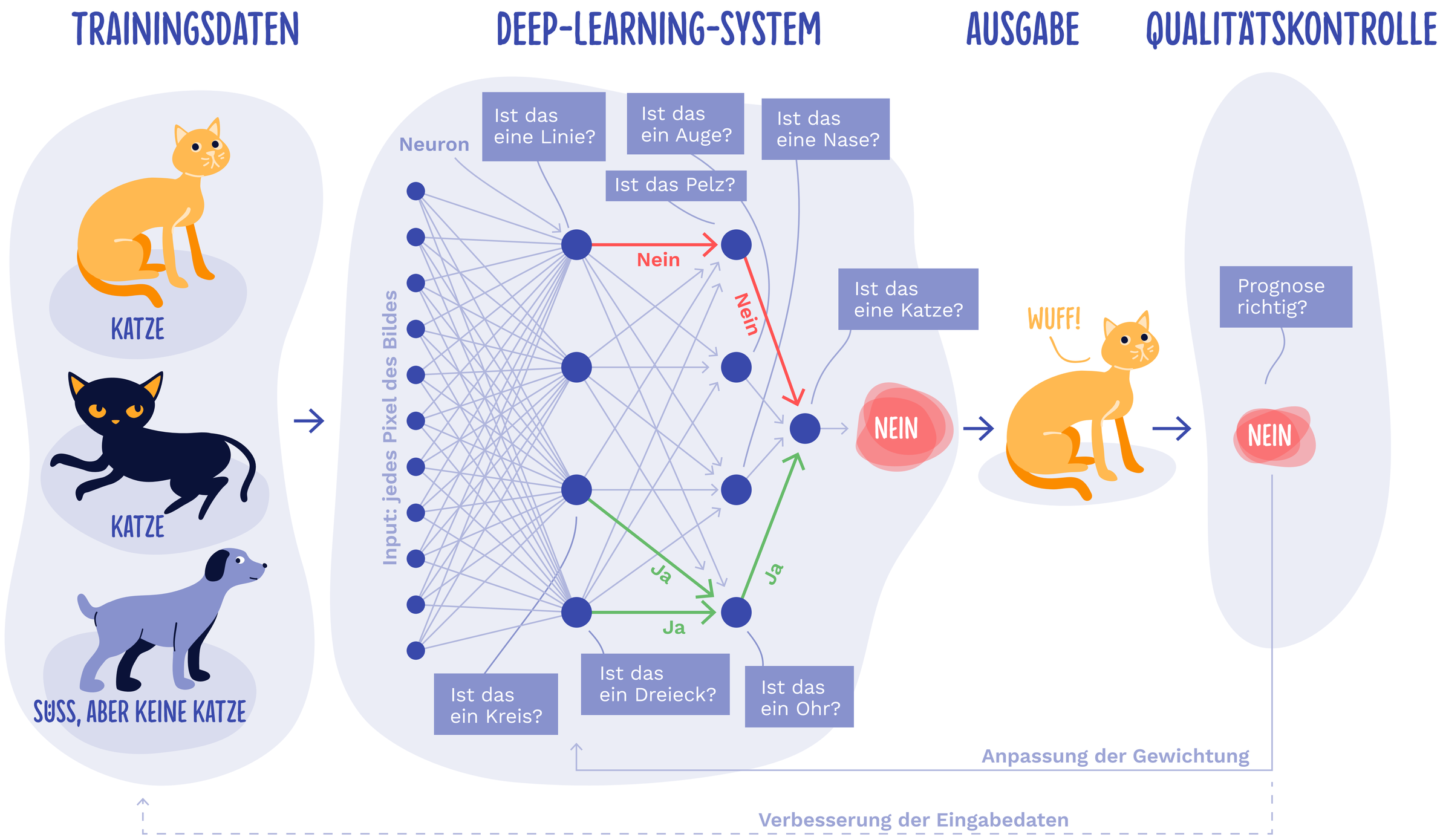


* Ach, der Nörgler ist ja immer noch da! Ja, schon klar. Das Ergebnis lautet eigentlich sowas wie 0,01 (und nicht 0) für „Das ist keine Katze!" und 0,99 (und nicht 1) für „Das ist eine Katze!". Du bist sooferngend, Mann.

Ja, es sieht viel komplizierter aus als in den Beispielen. Denn, na ja, es ist viel komplizierter. Aber, ich meine, Du bist jetzt ein echter Pro, oder?

5 Trainiere das neue System

Selbst der talentierteste Spieler ist nichts ohne Training. Also Vorhang auf: Nimm Deine gesammelten (kategorisierten) Daten und füttere damit das Monster, das Du erschaffen hast.

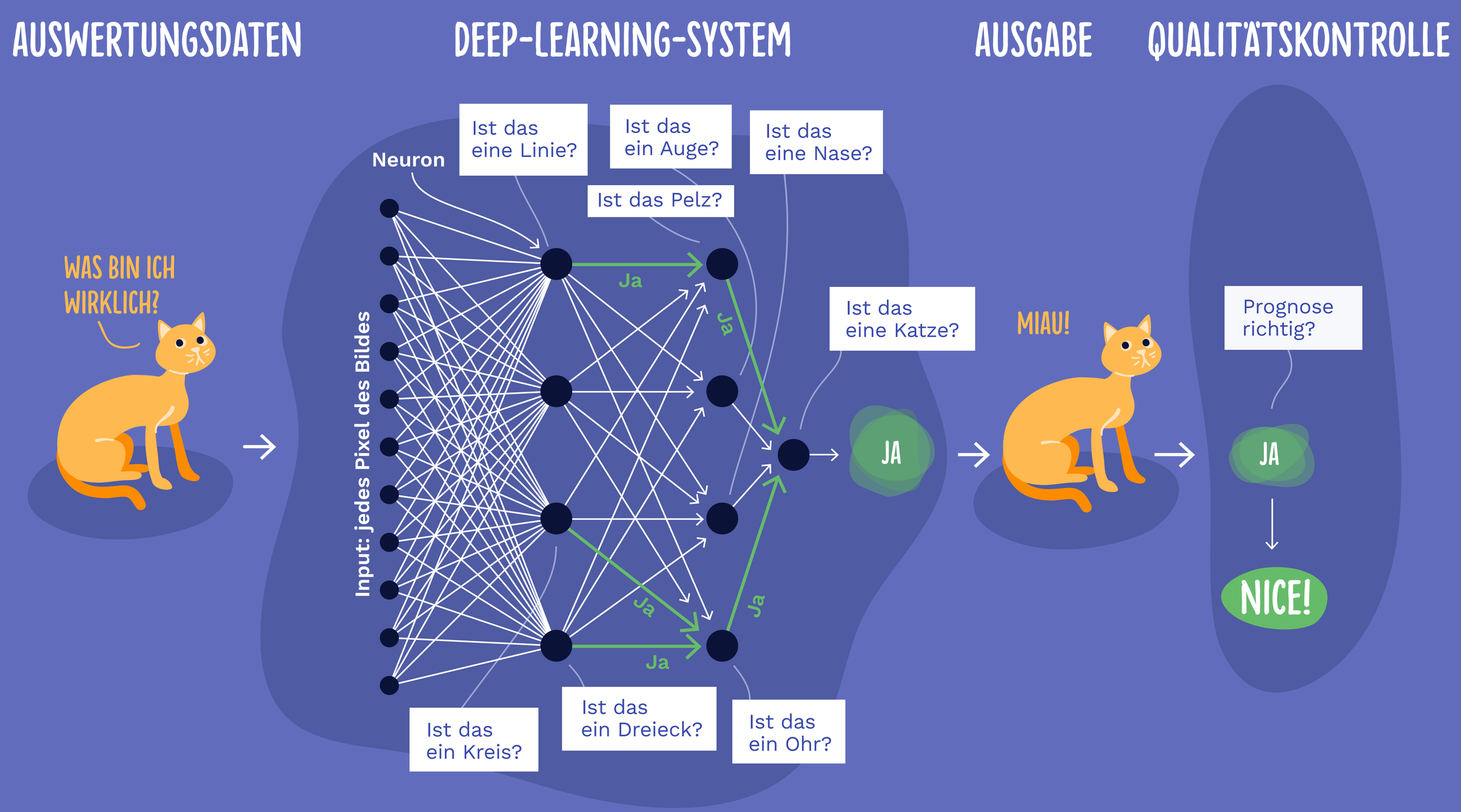


Das Training ist hart. Wir trainieren und optimieren unser Deep-Learning-System, bis die Ausgabe unsere extrem hohen Erwartungen erfüllt.

Optimierung bedeutet, die Gewichtung im Netz zu verändern, damit die Eingabe eines Katzenbildes auch tatsächlich zu einer als Katze kategorisierten Ausgabe führt. Das bedeutet aber auch, dass mehr bzw. bessere Daten gesammelt werden müssen.

6 Evaluierung des trainierten Systems

Wir können uns nicht einfach nur auf Ergebnisse aus unseren (kategorisierten) Auswertungsdaten verlassen. Das ist der Moment der Wahrheit: Lass uns neues, unbekanntes Datenmaterial hineinwerfen und sehen, wie unsere Maschine mit diesem neuen Input zurechtkommt.

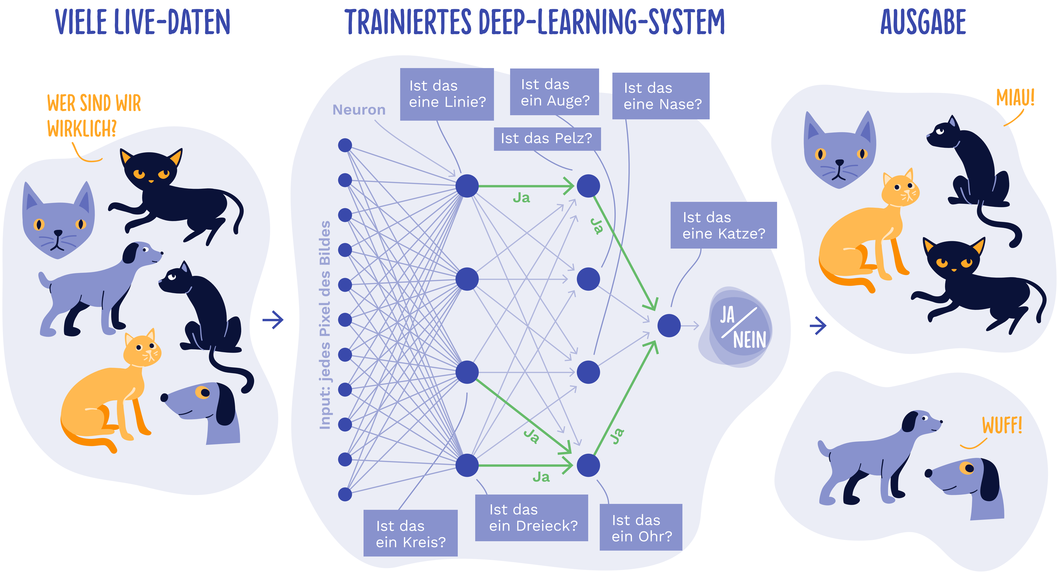


Wir optimieren unser System kontinuierlich, um auf Basis der Auswertungsdaten die besten Ausgabeergebnisse zu erzielen. Die Ergebnisse sind zufriedenstellend? Dann kann es losgehen.

7 Launch des Live-Systems

Für die Schlaumeier unter Euch: noch da? Okay, wir wissen: Jetzt müssen wir das System auf einem (Cloud-)Server* oder einem Client wie einem Smartphone einsetzen. Sobald das Training abgeschlossen ist, was Monate dauern kann**, benötigt das Live-System nur Millisekunden für die Evaluation. **BÄÄM!**

* Das wäre eigentlich die beste Gelegenheit in diesem Whitepaper, um Werbung für Azure zu machen, der großartigen Cloud-Lösung von Microsoft. Ich mein ja nur.
** Ohne Azure Batch AI-Dienste. Ich mein ja nur.



Man lernt nie aus, aber Du hast gerade Deinen Doktor in Deep Learning gemacht, Du alter Fuchs.

Autor

ISKENDER DIRIK

Managing Director / CEO
Microsoft ScaleUp Berlin

Zugabe: Warum ist Deep Learning so revolutionär?

Iskender Dirik: Moritz, Du bist Mitbegründer von Twenty Billion Neurons (TwentyBN) und arbeitest an fortschrittlichen Deep-Learning-Systemen, die Videos verstehen. Warum ist Deep Learning so revolutionär?

Moritz Müller-Freitag: Beim Deep Learning werden Computer auf eine völlig neue Art und Weise eingesetzt. Statt wie bisher detaillierte Schritt-für-Schritt-Anweisungen zu programmieren, denen der Computer folgen soll, setzen wir die tiefgehenden neuronalen Netze der realen Welt aus, damit sie von dieser lernen können. Wir „zeigen“ dem Computer, was er tun soll, statt es zu „programmieren“.

Für mich persönlich gibt es zwei Gründe, warum Deep Learning eine so transformative Technologie ist.
Zum einen können wir mit Deep Learning Probleme lösen, die zuvor unlösbar waren. In den vergangenen fünf Jahren gab es eine Reihe von Verbesserungen bei den Treppenfunktionen, wodurch die Grenzen der Fähigkeiten von Computern immer wieder komplett verschoben wurden. Dazu gehören Fortschritte bei der Bilderkennung, Maschinenübersetzung und Sprachsynthese sowie seit kurzem auch beim Verstehen von Videoaufnahmen, dem Schwerpunkt von TwentyBN.

Außerdem ist Deep Learning erstaunlich zugänglich für Studierende bzw. jeden, der sich für dieses aufstrebende Fachgebiet interessiert und sich fortbilden möchte. Wer Analysis und Algebra versteht, versteht auch, was beim neuronalen Netz unter der Haube abgeht. Studierende können es verstehen – und das kann man nicht über andere Fachgebiete wie Kernphysik oder Raketenwissenschaften sagen.

Alles in allem glaube ich, dass Deep Learning von Grund auf verändern wird, wie wir Software entwickeln. Das Spannende daran ist, dass wir noch ganz am Anfang dieser Veränderungen stehen. Es wird noch so viel mehr kommen!

Iskender Dirik: Vielen Dank, Moritz. Du könntest nicht zufällig meine Geheimwaffe gegen Schlaumeier sein, oder?