

Programmierung eines lernfähigen neuronalen Netzwerks

Vorwissenschaftliche Arbeit

verfasst von

Thiemo Tiziani

Inhaltsverzeichnis

1. Einleitung	4
2. Der Aufbau eines künstlichen neuronalen Netzwerks	5
3. Künstlich neuronale Netze trainieren	6
Hebb'sche Lernregel	8
Delta Learning Rule	8
Backpropagation	9
4. Die verschiedenen Arten von künstlichen neuronalen Netzwerken.....	10
Das Perzeptron	10
Das Feedforward Netzwerk	10
Das rekurrente neuronale Netzwerk	11
5. Convolutional Neural Network.....	13
6. Die drei Stärken von künstlicher Intelligenz	15
Die schwache Intelligenz	15
Die starke Intelligenz	15
Die künstliche Superintelligenz	16
7. Die vier Typen von künstlicher Intelligenz.....	17
Reaktive Maschinen	17
Begrenzte Speicherkapazität	17
Theory of Mind.....	17
Selbstwahrnehmung	17
8. Testen von künstlicher Intelligenz	18
Das Testen von künstlicher Intelligenz.....	18
Der Turing Test.....	18

Winograd-Schema-Herausforderung	20
Der visuelle Turing Test.....	20
9. Die Gefahren von künstlicher Intelligenz	20
Künstliche Intelligenzen als Ersatz für menschliche Arbeitskräfte	21
Autonome Waffensysteme	21
Der Missbrauch von künstlicher Intelligenz zur Überwachung.....	22
10. Die Programmierung eines künstlich neuronalen Netzwerks	23
Der erste Versuch mit Neruoph.....	23
Zahlenerkennung mit einem künstlich neuronalen Netzwerk.....	26
11. Schlussfolgerung	29
12. Abbildungsverzeichnis	30
13. Literaturverzeichnis	31

1. Einleitung

Ein Computer hat gegenüber dem menschlichen Gehirn Vorteile und Nachteile. So kann ein Computer schneller und präziser mathematischen Probleme lösen. Das Team DAViS der FH Graubünden hält den aktuellen Rekord von 12.8 Billionen berechneten Stellen der Kreiszahl PI nach nur 108 Tagen und 9 Stunden¹.

Stellt man einem Computer aber die Aufgabe, eine Katze in einem Bild zu finden, ist man bisher schnell an die Grenzen eines Computers gestoßen. Künstliche Intelligenzen sollen dabei nun Abhilfe schaffen. Bei einem künstlich neuronalen Netzwerk nimmt man sich die Natur zum Vorbild, in dem man die Strukturen eines neuronalen Netzes modelliert und technisch nachbildet².

Dieser Fortschritt hat sich schnell in allen möglichen Bereichen als nützlich erwiesen. Man findet künstliche Intelligenzen mittlerweile fast überall. So gut wie jedes moderne Handy hat künstliche Intelligenzen im Einsatz, um Kameraeinstellungen automatisch anzupassen, Texte in Bildern zu übersetzen, Gegenstände erkennen und vieles mehr. Auch Übersetzungsplattformen, wie Google Translate, DeepL Translate verwenden künstliche Intelligenzen, um den Übersetzungen grammatikalische Sinnhaftigkeit hinzuzufügen. Social Media Plattformen verwenden künstliche Intelligenzen dazu, die Aktivitäten der Nutzer genau zu verfolgen und daraus einen Feed zu erstellen welcher exakt auf den Nutzer abgeschnitten ist. Das erhöht die Zeit, die ein User auf der Plattform verbringt. Auch Werbungen werden durch künstliche Intelligenzen auf den Endnutzer spezialisiert, um die Wahrscheinlichkeit zu erhöhen, dass ein Nutzer Interesse für die Werbung zeigt und auf diese klickt. Ebenfalls werden in Autos künstliche Intelligenzen immer präsenter. So unterstützen sie den Fahrer durch automatisches Bremsen, Spurassistenten oder in manchen Ländern sogar durch vollständiges autonomes Fahren. Auch die Gesichtserkennung erfolgt über eine künstliche Intelligenz, welche unterscheiden muss, ob das erkannte Gesicht nur ein Ausdruck, eine Maske oder das echte Gesicht ist. Gesichtserkennung hat aber auch andere Vorteile. So können Behörden durch die Gesichtserkennung autonom Personen durch Überwachungskameras ausfindig machen und deren Schritte zurückverfolgen. Aber auch die künstliche Intelligenz hat ihre Schattenseiten, so steht

1 Vgl. IQ 1
2 Vgl. IQ 2

der Fortschritt mal wieder der Privatsphäre im Weg. Durch künstliche Intelligenzen können nun Datenanalysen durchgeführt werden, in denen unterschieden wird, ob jemand zum Beispiel einen Kredit bekommt, ob er reisen darf, ein öffentliches Verkehrsmittel verwenden darf, usw. Auch kann mittels künstlicher Intelligenz die Rückfallwahrscheinlichkeit von Kleinkriminellen beurteilt werden. Weiters ist die künstliche Intelligenz auch für Überwachungssysteme ein großer Fortschritt. Missbräuchlich wird die künstliche Intelligenz leider auch verwendet. Beispiele dafür sind zum Beispiel das Automatisieren von Call Centern, die auf das Betrügen von Menschen aus sind oder Bots, welche falsche Neuigkeiten verbreiten und somit Gerüchte aufbringen. Für größere Konzerne sind künstliche Intelligenzen von enormem Nutzen. Diese Konzerne nutzen diese Technologie, um einfacher Daten von ihren Nutzern zu sammeln und zu analysieren und diese Daten an andere Konzerne zu verkaufen, welche wiederum dadurch ihre Website so gestalten können, um das Kaufinteresse in ihrer Zielgruppe zu steigern³.

2. Der Aufbau eines künstlich neuronalen Netzwerks

Ein künstlich neuronales Netzwerk ist eine riesige Ansammlung an sogenannten Knotenpunkten, welche auch Neuronen genannt werden. Die Neuronen lassen sich dabei in vier verschiedene Kategorien einteilen, je nachdem welche Aufgabe ihnen zugeteilt wurde. Ein künstlich neuronales Netzwerk hat drei verschiedene Schichten: Die Eingabeschicht, die verborgene Schicht und die Ausgabeschicht. In jedem dieser Schichten befindet sich eine Ansammlung an Neuronen, welche auch den Namen ihrer Schicht in sich tragen. Dazu kann jede Schicht noch ein Bias Neuron beinhalten, welche keinen Informationsfluss hat. Der Bias besitzt einen Wert, welcher am Anfang des Trainings meist zwischen -1 und 1 liegt. Jedes Neuron vergibt der durchfließenden Information ein Gewicht und addiert dieses mit dem Wert des Bias bevor es an das nächste Neuron in der nächsten Schicht weitergegeben wird. In der Eingabeschicht befinden sich die Input-Neuronen, welche das neuronale Netzwerk mit Informationen versorgt. Die verborgene Schicht kann beliebig viele Ebenen besitzen, während die Eingabe- und Ausgabeschicht nur eine Ebene besitzen kann. Umso mehr verborgene

³ Vgl. IQ 3

Schichten es in einem neuronalen Netzwerk gibt, desto tiefer ist es und umso genauer können die Informationen ausgewertet werden. In jeder Ebene der verborgenen Schicht befinden sich die Hidden-Neuronen, welche die erhaltenen Informationen neu gewichten und von Neuron zu Neuron weitergeben, bis diese die Ausgabeschicht erreicht. Die Ausgabeschicht ist die letzte Ebene eines neuronalen Netzwerkes und enthält die Output-Neuronen, welche die resultierende Entscheidung beinhalten, welche die Information dann ausgibt⁴.

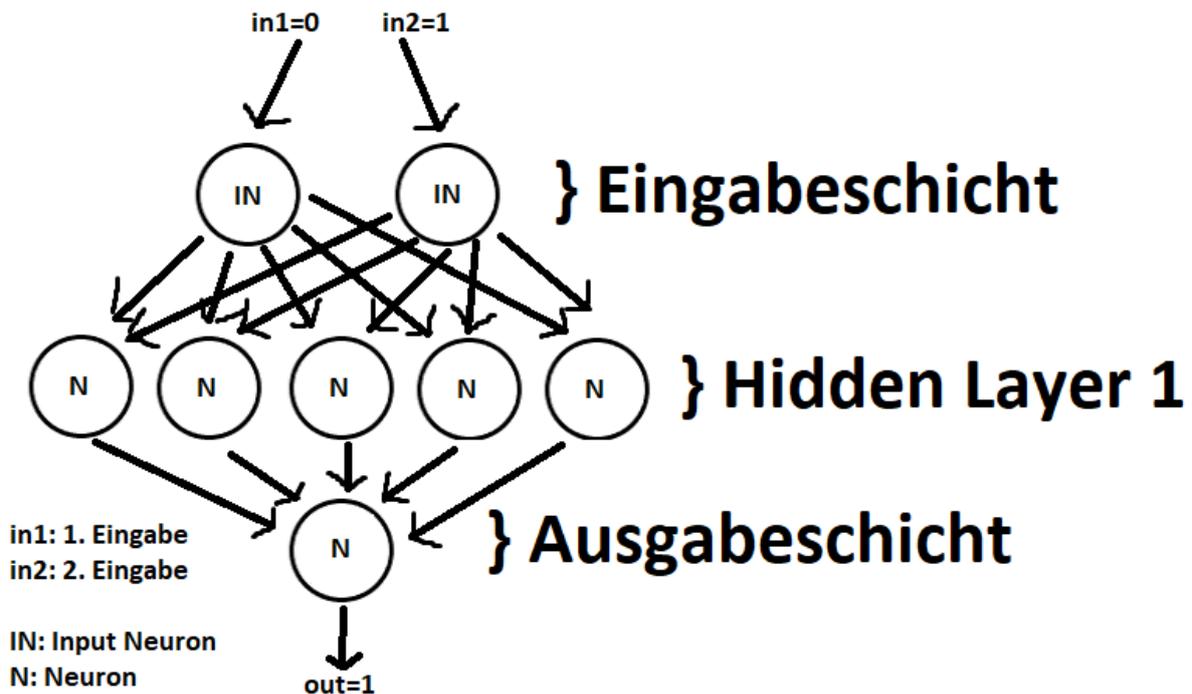


Abbildung 1: Eigene Zeichnung zum Aufbau eines neuronalen Netzwerkes

3. Künstlich neuronale Netze trainieren

Bevor ein künstlich neuronales Netzwerk antrainiert wurde, weisen fast immer so gut alle Werte 0 auf, manche Entwickler geben jedoch, aus mathematischen Gründen, eine Zahl an, welche einen sehr geringen Wert nahe Null aufweist. Ein künstlich neuronales Netzwerk wird durch große Datenmengen antrainiert. Für ein künstlich neuronales Netz bedeutet lernen sich selbstständig anpassen und verändern zu können, sollte sich die Umgebung des Netzwerkes verändern. Dazu passt das

⁴ Vgl. IQ 4

Netzwerkwerk interne Strukturen anhand der Daten, welche durchgegeben werden, an, um die Ausgabe anzupassen⁵.

Antrainieren - Lernregel

Um ein künstlich neuronales Netzwerk anzutrainieren, gibt es einige verschiedene Möglichkeiten, welche Lernregeln genannt werden. Lernregeln sind Gleichungen und Algorithmen, welche die Gewichte eines neuralen Netzwerkes verändern. Eine Lernregel gibt als Algorithmus Auskunft darüber, welche Gewichte des Netzes erhöht oder reduziert werden sollen. Lernregeln lassen sich dabei in drei verschiedenen Kategorien einteilen: das „überwachte Lernen“, das „unüberwachte Lernen“ und das „bestärkende Lernen“. Der Unterschied zwischen den ersten beiden Methoden ist, dass ein Netzwerk, welches mit dem überwachten Lernen arbeitet, einen Datensatz benötigt, bei dem im Vorfeld ein Mensch die gewünschte Netzwerkausgabe angibt und das Netzwerk so zu einem vordefinierten Wert hinarbeitet. Ein Beispiel für das überwachte Lernen ist das Erkennen von Bildern. Es muss hier im Vorfeld ein großer Datensatz mit einem von Menschen vordefinierten Ergebnis angegeben sein. Das Netzwerk kann daraus lernen und nach einigen Durchläufen erkennen, in welchen Bildern sich z.B. welche Tiere befinden. Bei einem Netzwerk, welches mit unüberwachtem Lernen arbeitet, wird nur sehr bedingt menschliche Unterstützung benötigt und das Netzwerk kann größtenteils selbstständig arbeiten. Das bestärkende Lernen kommt zum Beispiel bei der Betrugserkennung zum Einsatz, wobei man dabei die Ausreißerererkennung verwendet, bei der das Netzwerk erkennen kann, ob zum Beispiel ein Kunde von dem Muster der anderen Kunden abweicht. Das bestärkende Lernen unterscheidet sich sehr von den anderen beiden Lernmethoden. Das bestärkende Lernen kommt der natürlichen Art des Lernens viel näher. Beim bestärkenden Lernen lernt das neuronale Netzwerk nicht anhand eines Datensatz und es folgt kein Feedback nach jeder Aktion, stattdessen folgt ein Feedback nach einer gesamten Handlung, zum Beispiel bei einem Brettspiel, sobald das Spiel gewonnen oder verloren wurde. Das Netzwerk verwendet dann die Informationen auf Versuch und Irrtum, um eigenständig zu einer Lösung des Problems zu finden. Je nach Rechenleistung können dabei mehrere Spiele in einer Simulation beschleunigt durchgerechnet werden. Für bestärkendes Lernen gibt es einige Beispiele, so kann

⁵ Vgl. IQ 5

ein Netzwerk mit bestärkendem Lernen zum Beispiel verschiedene Strategien entwickelt, um möglichst viele Produkte zu verkaufen. Dadurch kann ein Netzwerk mit verschiedenen Aktionen wie das Versenden von Produktproben oder das Versenden einer E-Mail oder gar einer Veranstaltung zu gewissen Zielpersonen rausfinden, welche Aktion die beste Belohnung abgibt, also ein positives Feedback. Ein sehr bekanntes Beispiel für bestärkendes Lernen ist das vom britischen Unternehmen DeepMind entwickelte Spiel AlphaGo, welches 2017 den Weltmeister im Go-Spielen besiegen konnte und das bestärkende Lernen zum Superstar in der Welt der künstlichen Intelligenzen machte⁶.

Hier sind ein paar wichtige Lernregeln für neuronale Netzwerke

Hebb'sche Lernregel

Die Hebb'sche Lernregel ist die älteste Lernregel in der Auflistung und wurde bereits 1949 von Donald O. Hebb formuliert. Die Hebb'sche Lernregel ist der Grundbaustein von den meisten komplizierteren Lernregeln. Die Hebb'sche Lernregel besagt, dass je häufiger Neuron A gleichzeitig mit Neuron B Aktivität ausweist, desto bevorzugter werden diese auseinander reagieren, das heißt, ihre Verwindungen werden gestärkt. Die Gewichtungen der Neuronenverbindung werden also je nach ihrer Aktivität gestärkt⁷.

Delta Learning Rule

Die Delta Lernregel oder auch Widrow-Hoff-Regel genannt, gehört zu den überwachten Lernmethoden. Bei der Delta Lernregel werden die Gewichte in Abhängigkeit der Differenz zwischen dem Output des Netzwerkes und dem Erwartungswert neu gewichtet. Sie ist an der Hebb'sche Lernregel angelehnt. Bei der Delta Lernregel ist aber zu beachten, dass nur Netzwerke mit nur einer Schicht trainierbarer Gewichte, sowie einer linearen Ausgabefunktion, angelernt werden können. Die Back Propagation Lernregel, welche als Verallgemeinerung der Delta-Regel gilt, ermöglicht es dann mehrschichtige Netzwerke anzutrainieren sowie der Ausgabe von alternativen Ausgabefunktionen. Bei der Delta Regel ist auch die

⁶ Vgl. IQ 6

⁷ Vgl. IQ 7

Fehlerrate, also die Differenz zwischen erwünschten und tatsächlichen Output der Ausgabeneuronen von großer Wichtigkeit. Ist der Ausgabewert eines Ausgabeneurons geringer als die gewünschte Ausgabe im Datenset wird die Verbindung zu diesen Neuronen erhöht, in dem ein berechneter Wert zu den Verbindungsgewichtungen hinzugefügt wird. Ist der Ausgabewert größer als der gewünschte Ausgabewert, geschieht genau das Gegenteil, und die Gewichtung wird gesenkt⁸.

Backpropagation

Die Backpropagation, bzw. Fehlerrückführung ist eine der meistverwendeten Lernregeln für neuronale Netzwerke⁹. Sie gehört zu den überwachten Lernmethoden. Backpropagation hat den Vorteil, dass es hingegen der Delta Lernregel, mehrschichtige Netzwerke antrainieren kann, während die Delta Regel nur Netzwerke mit einer Schicht an trainierbaren Neuronen antrainieren kann. Die Backpropagation Lernregel wurde in den 1970er Jahren von einigen Autoren wie unter anderen Paul Werbos vorgeschlagen. Sie geriet aber für über ein Jahrzehnt in Vergessenheit bis sie durch Rumelhart, Hinton und Williams, welche 1986 einen Ansatz mit der Back Propagation entwickelten, welcher als eine Verallgemeinerung der Delta-Regel gilt, wieder verwendet wurde. Die Backpropagation Methode ist eines der wichtigsten Verfahren für das überwachte Lernen von neuronalen Netzwerken. Bei der Backpropagation handelt es sich um ein iteratives Verfahren, das heißt es ist ein Verfahren, bei dem der Datensatz immer wieder in das Netzwerk eingegeben wird und Anhand der Differenz zwischen der gewünschten Ausgabe und der Ausgabe des Netzwerkes eine Fehlerrate berechnet. Ist diese Fehlerrate über einen gewissen Wert wird der Fehler über die Ausgabe zur Eingabeschicht zurück propagiert. Dabei werden die Neuronen-Verbindungen, welche verantwortlich für den Fehler sind, ermittelt und neu gewichtet, so dass es beim nächsten Durchlauf zu einer geringeren Wahrscheinlichkeit eines Fehlers kommt¹⁰.

8 Vgl. IQ 8
9 VGL. IQ 9
10 Vgl. IQ 10

4. Die verschiedenen Arten von Künstlich Neuronalen Netzwerken

Künstlich Neuronale Netzwerke lassen sich in sehr viele verschiedene Typen einteilen, welche alle ihre eigenen Besonderheiten haben. Die wichtigsten und bekanntesten dieser Typen sind:

- Das Perceptron
- Das Feedforward Netzwerk
- Das Convolutional Neural Network
- Das rekurrente neuronale Netzwerk¹¹

Das Convolutional Neural Network wird später als eigenes Thema behandelt, weil es einige Besonderheiten mit sich bringt.

Das Perzeptron

Bei dem Perzeptron handelt es sich um den ältesten Typen von neuronalen Netzwerken. Es wurde im Jahre 1958 von Frank Rosenblatt vorgestellt. In der einfachsten Form besteht diese Art von Netzwerk aus einem Schwellenwert und einem einzelnen Neuron, welches anpassbare Gewichtungen besitzt. Beim Perzeptron wird zudem zwischen einem einlagigen und einem mehrlagigen, also einem Perzeptron welches mehrere Ebenen aufweist unterschieden¹².

Das Feedforward Netzwerk

Ein vorwärts gerichtetes Netzwerk ist ein Netzwerk, bei dem der Datenstrom nur in eine Richtung fließt und keine Rückkopplungen aufweist. Das heißt, dass der Datenstrom von Layer zu Layer fließt und dabei keine Zyklen oder Schleifen aufweist.

¹¹ Vgl. IQ 11
¹² Vgl. IQ 12

Diese Art von Netzwerk war die erste und einfachste Version der künstlich neuronalen Netzwerke¹³.

Das rekurrente neuronale Netzwerk

Die rekurrenten neuronalen Netzwerke sind eine etwas besondere Art der neuronalen Netzwerke. Bei einem rekurrenten Netzwerk sind Neuronen in der gleichen Schicht oder in unterschiedlichen Schichten zurückgekoppelt, das heißt, der Datenstrom fließt nicht wie beim Feedforward Netzwerk von der Eingabeschicht bis zu der Ausgabeschicht ohne Schleifen oder Rückkopplungen, sondern fließt teilweise im Kreis.

Im unteren Schaubild sehen Sie ein Beispiel für ein rekurrentes Netzwerk. In diesem Beispiel sehen wir oben unsere Eingabeschicht, wo sich die InputNeuronen(IN) befinden, welche die Informationen auf Gültigkeit überprüft und diese dann ohne zu verändern, in unsere versteckte Schicht weitergibt, welche an ihren Ausgang eine direkte Rückkopplung besitzt. Das bedeutet, dass der eigene Ausgang der Neuronen als weiterer Eingang verwendet wird und somit unsere Information erneut durch das Neuron durchfließt, bevor dieses an die Ausgabeschicht weitergegeben wird, wo dann die Ausgabe des Netzwerkes stattfindet.

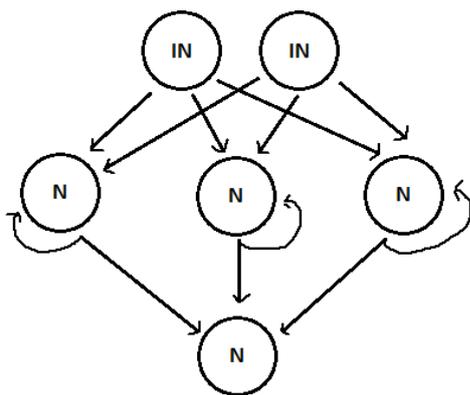


Abbildung 2: Eigene Zeichnung eines rekurrenten Netzwerkes

13 Vgl. IQ 13

Durch solche Rückkopplung kann ein Netzwerk zeitlich codierte Informationen gewinnen, welche zum Beispiel für Übersetzungen oder Spracherkennung notwendig sind. Diese Netzwerke sind auch dem menschlichen Gehirn näher als Feedforward Netzwerke. Ein rekurrentes neuronales Netz lässt sich anhand der Art der Rückkopplung nochmal in vier verschiedene Arten unterscheiden:

- Rekurrentes Netz mit direkter Rückkopplung
- Rekurrentes Netz mit indirekter Rückkopplung
- Rekurrentes Netz mit seitlicher Rückkopplung
- Rekurrentes Netz mit vollständiger Rückkopplung

Rekurrentes Netz mit direkter Rückkopplung

Bei einem Netzwerk mit direkter Rückkopplung wird der eigene Ausgang eines Neurons im Netzwerk als weiterer Eingang zum Neuron benützt, so wie es in der Abbildung oben bereits veranschaulicht wurde.

Rekurrentes Netz mit indirekter Rückkopplung

Bei einem Netzwerk mit einer indirekten Rückkopplung wird der Ausgang eines Neurons mit einem anderen Neuron in der vorhergehenden Schicht verbunden.

Rekurrentes Netz mit einer seitlichen Rückkopplung

Bei einem Netzwerk mit einer seitlichen Rückkopplung wird der Ausgang eines Neurons mit einem anderen Neuron in derselben Schicht verbunden.

Rekurrentes Netz mit einer vollständigen Rückkopplung

Bei einem Netzwerk mit einer vollständigen Rückkopplung werden die Ausgänge von allen Neuronen mit den Eingängen aller Neuronen im Netzwerk verbunden¹⁴.

5. Convolutional Neural Network

Ein Convolutional Neural Network, auf Deutsch „faltendes neuronales Netzwerk“ genannt, ist eine Art der neuronalen Netzwerke, welche für Bild- und Spracherkennung sehr gut geeignet ist. Einer der Begründer dieser Art von neuronalen Netzwerken ist Yann LeCun. Der Aufbau eines faltenden neuronalen Netzwerkes ist sehr ähnlich zu der Sehrinde des Gehirns, auch die Funktionsweise ist zu einem Teil den biologischen Vorgängen nachempfunden. Das faltende neuronale Netzwerk besteht aus mehreren Schichten, welche von denen, die wir von den herkömmlichen neuronalen Netzwerken kennen abweichen. So besteht ein faltendes neuronales Netzwerk aus Faltungsschichten, Subsampling-Schichten und vollvermaschte Schichten. Die Trainingsmethode dieser Netzwerkart ist für gewöhnlich überwacht. Bei einem herkömmlichen neuronalen Netzwerk müsste aufgrund des Aufbaues mit voll- oder teilvermaschten Neuronen in mehreren Ebenen bei dem Verarbeiten der Bilder eine der Pixelanzahl entsprechende Zahl an den Eingängen vorhanden sein, was das Netzwerk sehr leistungsanfordernd macht, da die Zahl der Layer und Verbindung riesig wird. Das faltende neuronale Netzwerk setzt sich aus folgenden Schichten zusammen:

14 Vgl. IQ 14

Die Convolutional-Schicht

Bei der Convolutional-Schicht handelt es sich, wie auch schon der Name sagt, um die eigentliche Faltungsebene. Durch diese Ebene ist das Netzwerk dazu in der Lage, in den Eingabedaten verschiedene Merkmale zu erkennen und zu extrahieren. Dies ist vor allem für die Bildverarbeitung interessant, da dadurch Merkmale wie Linien, Kanten oder bestimmte Formen erkannt werden können. Die Verarbeitung dieser Eingabedaten erfolgt in Form einer Matrix.

Die Pooling Schicht

In der Pooling Schicht, welche auch als Subsampling Schicht bezeichnet wird, werden die erkannten Merkmale der Convolutional-Schicht verdichtet und die Auflösung reduziert. Hierfür werden Methoden wie das Maximal-Pooling oder das Mittelwert-Pooling verwendet. So werden überflüssige Informationen entfernt und die Datenmenge reduziert, ohne die Leistungsfähigkeit beim Lernen zu verringern, im Gegenteil, es wird durch das reduzierte Datenaufkommen die Berechnungsgeschwindigkeit erhöht.

Die vollständig verknüpfte Schicht

Die vollständig verknüpfte Schicht bildet den Abschluss im faltenden Netzwerk. Die Merkmale und Elemente der vorgelagerten Schichten sind dabei mit den Ausgabemerkmale verknüpft. Die Neuronen in dieser Schicht können in mehreren Ebenen angeordnet sein, die Anzahl dieser Neuronen ist dabei abhängig von den Klassen bzw. den Objekten die das Netzwerk unterscheiden soll.

Die Convolutional-Schicht und die Pooling-Schicht kann in dieser Kombination mehrfach hintereinander vorhanden sein, während die vollständig verknüpfte Schicht dann den Abschluss bildet. Da es sich bei der Pooling-Schicht und der Convolutional-Schicht um lokal vermaschte Teilnetze handelt, bleibt die Anzahl von Verbindungen in diesen Ebenen selbst bei größeren Eingabemengen begrenzt und in einem Rahmen, welcher noch beherrschbar ist. Die wichtigsten Anwendungsbereiche dieser Art von

neuronalen Netzwerken finden wir in Bereichen der Gesichtserkennung und Objekterkennung. Auch die Spracherkennung hat bei diesen Netzwerken deutliche Vorteile, da diese Art von Netzwerk bessere Ergebnisse beim semantischen Parsen, bei der Klassifizierung und Modellierung von Sätzen und beim maschinellen Übersetzen aufzeigt¹⁵.

6. Die drei Stärken von künstlichen Intelligenzen

Die schwache Intelligenz

Jede Form von künstlicher Intelligenz, welche wir in unserem Zeitalter erreicht haben, gehört der Gruppe der schwachen Intelligenzen an. Schwache künstliche Intelligenzen sind der Intelligenz des Menschen nicht ebenbürtig. Schwache Intelligenzen können einfache Aufgaben übernehmen, wie das Erkennen von Texten, das Analysieren von Bildern und deren Motive sowie Gegenstände oder Gesichter darin erkennen. Schwache Intelligenzen sind in unserem Zeitalter allgegenwärtig. So gut wie jeder Mensch trägt ein modernes Smartphone mit sich, welches künstliche Intelligenzen beinhaltet, die zum Beispiel die Entsperrung per Gesichtserkennung, das Anpassen der Kameraeinstellungen anhand von Bilddaten, das Übersetzen von Texten oder ganz einfache Informationsgewinnung durch Sprachbefehle übernehmen. Schwache Intelligenzen begleiten uns mittlerweile in so gut wie jedem Bereich unseres digitalen Zeitalters¹⁶.

Die starke Intelligenz

Eine starke Intelligenz hat eine dem Menschen ebenbürtige Intelligenz. Eine starke Intelligenz soll in einigen Bereichen entsprechend gute Leistungen erbringen und anhand von Daten intelligente Entscheidungen treffen können. Ob eine starke Intelligenz ein eigenes Bewusstsein entwickeln kann oder Gefühle aufweist, wird stark umstritten. Somit ist auch noch nicht bekannt, welche Aufgaben eine starke Intelligenz übernehmen und ob diese zu Gunsten der Menschheit ausfallen wird, da eine starke Intelligenz noch theoretischer Natur ist¹⁷.

¹⁵ Vgl. IQ 15

¹⁶ Vgl. IQ 16

¹⁷ Vgl. IQ 16

Die künstliche Superintelligenz

Eine künstliche Superintelligenz ist eine künstliche Intelligenz oder ein Wesen, welches dem Menschen in einigen Gebieten vollkommen überlegen ist, während eine starke künstliche Intelligenz dem menschlichen Intellekt gleichzusetzen ist. Eine künstliche Superintelligenz könnte auch zu einem Wesen, welches künstlich, durch Gentechnik erschaffen worden ist, und überdurchschnittliche Fähigkeiten aufweist, welche dem natürlichen Menschen überlegen ist, werden. Solche Superintelligenzen sind jedoch noch reine Fiktion und kommen nur in Science-Fiktion oder Transhumanismus vor. So eine maschinelle Superintelligenz könnte zum Beispiel selbstständig ohne die Hilfe von Menschen operieren und lernen¹⁸. Eine der Probleme bei maschinellen Superintelligenzen ist die Eventualität eines Kontrollverlust. Ein gutes Beispiel einer solchen Kontrollverlust wurde von Nick Bostrom im folgenden Beispiel demonstriert: „Stellen Sie sich eine Maschine vor, die mit dem Ziel programmiert wurde, möglichst viele Büroklammern herzustellen, zum Beispiel in einer Fabrik. Diese Maschine hasst die Menschen nicht. Sie will sich auch nicht aus ihrer Unterjochung befreien. Alles, was sie antreibt, ist, Büroklammern zu produzieren, je mehr, desto besser. (...) Um dieses Ziel zu erreichen, muss die Maschine funktionsfähig bleiben. Das weiß sie. Also wird sie um jeden Preis verhindern, dass Menschen sie ausschalten. Sie wird alles tun, um ihre Energiezufuhr zu sichern. Und sie wird wachsen – und selbst dann nicht aufhören, wenn sie die Menschheit, die Erde und die Milchstraße zu Büroklammern verarbeitet hat. Das ergibt sich logisch aus ihrer Zielvorgabe, die sie nicht hinterfragt, sondern bestmöglich erfüllt.“¹⁹. Die einzige Möglichkeit eine Superintelligenz zu kontrollieren ist die Kontrolle der Fähigkeiten, sowie die Kontrolle der Motivation. Sollte eines dieser Möglichkeiten fehlschlagen, kann die Superintelligenz die Kontrolle der Menschheit erlangen²⁰.

18 Vgl. IQ 17

19 Vgl. IQ 18

20 Vgl. IQ 19

7. Die vier Typen von künstlicher Intelligenz

Reaktive Maschinen

Reaktive Maschinen sind der einfachste und grundlegendste Typ von künstlichen Intelligenzen. Reaktive Maschinen haben keine Erinnerung oder Wahrnehmung ihrer Umgebung. Reaktive Maschinen haben eine spezielle Aufgabe und führen diese basierend auf der aktuellen Situation aus. Ein Beispiel für solch eine künstliche Intelligenz ist der IBM Schachcomputer DeepBlue welcher, basierend auf Positionszahlen der Figuren, den schnellsten Weg zu einem Schachmatt berechnete und so seine Züge wählte.

Begrenzte Speicherkapazität

Eine limited memory K.I ist heutzutage die gebräuchlichste Art einer künstlichen Intelligenz. Diese Art der künstlichen Intelligenz ist in der Lage, eine Entscheidung basierend auf vergangene Daten, sowie Daten aus der aktuellen Situation zu treffen. Auf diese Methode basiert auch Gesichtserkennung, welche auf die vergangenen, eingespeicherten Gesichtsdaten zugreift und diese mit den aktuellen Eingaben vergleicht. Manche Gesichtserkennungssysteme lernen dabei sogar weiter und speichern bei erfolgreicher Erkennung neue Merkmale ab, um das Aussehen einer Person schneller und genauer erkennen zu können¹⁰.

Theory des Geistes

Theory of mind K.Is sind in der Lage menschliche Emotionen wahrzunehmen und zu verstehen und daran ihr Verhalten anpassen. Eine künstliche Intelligenz dieses Typus würde in der Lage sein, eine Wahrnehmung der Welt zu haben und sich selbst, anhand dem was sie lernt, zu verbessern. Aktuell gibt es aber noch keine künstlichen Intelligenzen, die diesen Typ erreichen konnten, somit kann man diese nur mit fiktionalen Robotern, wie dem R2D2 von Star Wars vergleichen.

Selbstwahrnehmung

Eine künstliche Intelligenz des Typen self awareness ist der menschlichen Wahrnehmung ebenbürtig. Sie hat ein vollständiges Bewusstsein und somit eine vollständige Wahrnehmung der Welt, Emotionen sowie Absichten und Reaktionen. Eine künstliche Intelligenz dieses Typus wird sich vollständig ihrer Existenz bewusst

sein, selbstständig denken können und somit alle alltäglichen Aufgaben eines Menschen erfüllen können, eventuell sogar besser²¹.

8. Testen von künstlich Intelligenzen

Das Testen von künstlichen Intelligenzen

Bereits 1950 wurde von Alan Turing der erste Test entwickelt, um künstliche Intelligenzen zu erproben. Dieser Test sollte künstliche Intelligenzen auf eine dem Menschen ebenbürtige Intelligenz testen. Bisher gibt es nur schwache Intelligenzen, welche durch Antrainieren eine gewisse Reihe von Aufgaben mit einer gewissen Präzision lösen können. Das Analysieren von Bildern und Landschaftsbegebenheiten, sowie das Erkennen von Gesichtern oder Texten zählt zu den Aufgabenbereichen schwacher Intelligenzen. Starke Intelligenzen zeichnen sich durch ihre Menschen ebenbürtige Intelligenz aus. Ein genaues Konzept oder Vorstellung solch einer starken Intelligenz ist bisher noch Theorie und birgt noch einige offene Fragen. Ob eine starke Intelligenz jemals ein Bewusstsein oder Gefühle, wie wir sie kennen, aufweisen kann, ist stark umstritten²².

Der Turing Test

Der Turing Test ist ein von Alan Turing entwickelter Test. Der Test wurde 1950 entwickelt und soll einer künstlichen Intelligenz ein Bewusstsein nachweisen können, welche der künstlichen Intelligenz die Möglichkeit gibt, selbstständig denken zu können. Dies stand aber zum Zeitpunkt der Entwicklung großer Kritik.

Der Philosoph John Searle versuchte mit dem Gedankenexperiment „chinesisches Zimmer“ die Aussage des Turing-Test, dass ein Computer intelligent ist, weil er diesen Test besteht, zu widerlegen²³. Beim „chinesischen Zimmer“ stellt man sich in einem leeren Raum vor eine Wand mit zwei Schlitzten. In einen Schlitz werden in regelmäßigen Abständen chinesische Fragen durchgeschoben. Im anderen Schlitz

21 Vgl. IQ 20

22 Vgl. IQ 21

23 Vgl. IQ 22

werden die beantworteten Fragen zurückgegeben. In Raum befindet sich eine Person, welche eine Anleitung zur Beantwortung der chinesischen Fragen in ihrer Muttersprache besitzt. Durch diese Anleitung kann die Person die Fragen korrekt beantworten und zurückgeben. Außerhalb des Zimmers wird aufgrund der korrekt beantworteten Fragen davon ausgegangen, dass die Person chinesisch spricht²⁴.

Der Test wurde dann von Turing genauer ausformuliert und sollte nun testen, ob eine künstliche Intelligenz einen Menschen überzeugend imitieren könnte. Bei der Ausführung der Turing Tests wird eine Testperson in ein Zimmer gebracht und führt dort 2 textbasierten Unterhaltungen. Die Person weiß, dass eine Unterhaltung von einem Computer geführt wird und die andere von einem Menschen. Die Person hat dann meistens begrenzt Zeit, um herauszufinden, welche der Unterhaltungen von einem Computer geführt wird. Entscheidet sich die Person für den Menschen, wird der künstlichen Intelligenz eine überzeugende Imitierung von Menschen zugewiesen. Es gibt auch heutzutage noch einige Kritikpunkte, welche an der Tauglichkeit der Turing Tests zweifeln lassen, so ist die Bildung einer Person irrelevant, jede Person darf an einem Turing Test teilnehmen, es gibt keine Richtlinien, die eine Testperson vorbringen muss²⁵.

Im Juni 2014 wurde der Cleverbot mit dem Turing Test, an dem 1.334 Personen teilgenommen haben, erprobt.

Der Test wurde wie folgt durchgeführt: Dreißig Personen unterhielten sich per Chat auf drei Bildschirmen in zehn Runden zu je vier Minuten. Die Hälfte der Gespräche wurde von Mensch zu Mensch, die andere von Mensch zu künstlicher Intelligenz geführt. Die restlichen Teilnehmer stimmten ab, wie menschenähnlich ihnen die Antworten von 0 bis 10 vorkamen. 59.3% der zuhörenden Teilnehmer hielten Cleverbot für einen Menschen. Da die zuhörenden Personen jedoch nicht selbst mit der künstlichen Intelligenz schreiben durften, zählt diese durchgeführte Testung nicht als gültiger Turing Test²⁶.

24 Vgl. IQ 23

25 Vgl. IQ 22

26 Vgl. IQ 24

Winograd-Schema-Herausforderung

Die Winograd-Schema-Herausforderung, in „Englisch Winograd schema challenge“, ist eine von Hector Levesque erstellte Ergänzung zum Turing Test. Dabei werden in die Unterhaltungen Multi-Choice-Fragen eingebaut, die für Menschen leicht zu lösen sind, aber für künstliche Intelligenzen eine wahre Herausforderung darstellen. Die Fragen sind dabei so aufgebaut, dass es für die künstliche Intelligenz erforderlich ist, die Sprache zu verstehen und weiters notwendig ist, in einigen Bereichen vernünftig Denken zu können. Das erste Beispiel eines Winograd-Schemas kam von Terry Winograd, nach dem auch das Schema benannt wurde. Ein Beispiel für eine Winograd-Schema Challenge wäre folgende Feststellung mit angeschlossener Frage: „Die Trophäe passt nicht in den braunen Koffer, weil sie zu groß war. – Was war zu groß?“. Die Frage ist für einen Menschen leicht zu lösen, aber für eine künstliche Intelligenz, welche keine Vorstellung von Größen von Objekten hat, oder das Wissen, dass ein Koffer dazu da ist, Dinge reinzutun, unmöglich zu lösen.

Der visuelle Turing Test

Beim visuellen Turing Test, welcher von den Informatikern Michael Barclay und Antony Galton entwickelt wurde, wird der künstlichen Intelligenz die Aufgabe gestellt, einen Gegenstand in einem Bild zu finden und den richtigen Ort in den Multiple-Choice Antwortmöglichkeiten anzugeben. Alle Antwortmöglichkeiten sind dabei korrekt, jedoch sind manche Antworten menschlicher. Ein Mensch würde, wenn er bei einem Bild gefragt werden würde, wo die Tasse steht, nicht sagen: „Die Tasse steht vor dem Stuhl“. Die Antwort eines Menschen würde eher lauten: „Die Tasse steht neben dem Buch“²⁷.

9. Die Gefahren von künstlicher Intelligenz

Künstliche Intelligenzen zeigen uns viele neue Möglichkeiten auf, welche aber nicht alle zu unserem Gunsten sein müssen. Dass künstliche Intelligenzen mit sehr hoher Wahrscheinlichkeit von Geheimorganisationen zur Überwachung von Menschen

27 Vgl. IQ 21

benützt werden, ist vermutlich keine große Überraschung. Darüber hinaus bergen künstliche Intelligenzen aber auch andere Gefahren, die sich voraussichtlich schon in den nächsten Jahren zeigen werden.

Nachstehend werden einige Gefahren aufgezeigt.

Künstliche Intelligenzen als Ersatz für menschliche Arbeitskräfte

Künstliche Intelligenzen werden immer besser darin, Aufgaben zu erfüllen, wofür früher Menschen nötig waren. Auf Grund dieser Entwicklung werden in den nächsten Jahren vermutlich einige Personen ihren Job verlieren. Bereits jetzt übernehmen künstliche Intelligenzen einige Bereiche des Handels. Dass in Zukunft autonome Taxis nicht unwahrscheinlich sind, zeigt uns die Entwicklung von Teslas autonomer Fahrerunterstützung. Ein großer Teil von Arbeitsplätzen wird von künstlichen Intelligenzen übernommen werden. Dazu wird so die Vielfältigkeit gewisser Jobs wegfallen und es wird eine geringere Auswahl geben. Mehr Jobs im Bereich der Informationstechnik werden sich auftun. Die Anforderungen in diesem Bereich sind meist höher und kann daher nur von gut ausgebildeten Menschen ausgeführt werden²⁸.

Autonome Waffensysteme

Eine weitere Einsatzmöglichkeit für künstliche Intelligenzen sind autonome Waffensysteme. Ein autonomes Waffensystem wird von einer Intelligenz betrieben, welche autonom ihre Ziele aussucht. So könnte eine künstliche Intelligenz anhand gewisser Daten einen feindlichen Stützpunkt ausfindig machen, entscheiden, ob dieser eine Bedrohung ist und anhand dieser Daten autonom einen Angriff auf diesen Stützpunkt ausüben, ohne dass in irgendeinem Punkt ein Mensch eingreifen muss. Es ist nicht mehr so weit hergeholt, dass eventuell in der Zukunft statt Menschen Roboter und Drohnen gegeneinander kämpfen. In diesem Bereich gibt es auch teilautonome Waffen welche automatisch ihre Ziele erkennen, aber der Angriff auf das Ziel durch einen Menschen bewilligt werden muss²⁹. Immer mehr Staaten beschäftigen sich nun mit dem Thema, ob solche, vollständig autonome

28 Vgl. IQ 25

29 Vgl. IQ 26

Waffensysteme verboten werden sollen. So hat sich auch vor kurzem die UN-Konvention mit diesem Thema befasst. Man wurde sich jedoch nicht einig. Russland, USA und Israel waren gegen das Verbot von autonomen Waffen. Da bei der UN-Konvention eine Konsensregel herrscht, welche besagt, dass alle Parteien einig sein müssen, konnte kein Verbot autonomer Waffensysteme besiegelt werden. Das größte Problem von autonomen Waffen ist die Möglichkeit einer Fehlfunktion, welche eine Katastrophe auslösen könnte³⁰.

Der Missbrauch von künstlicher Intelligenz zur Überwachung

Künstliche Intelligenzen bieten uns immer mehr Möglichkeiten alltägliche Dinge zu automatisieren. So ist es möglich, in manchen Restaurants in China per Gesichtserkennung zu bezahlen oder den Zutritt zu Gebäuden per Gesichtserkennung zu automatisieren³¹. Das zeigt jedoch auch einige Schattenseiten auf und gibt neue Möglichkeiten der Überwachung. Durch künstliche Intelligenzen ist es möglich, binnen Sekunden Menschen anhand ihrer Gesichter in großen Menschenmengen zu erkennen und automatisch zu verfolgen. In China werden künstliche Intelligenzen bereits dafür verwendet, Personen anhand ihrer Herkunft automatisch zu überwachen und ihren Alltag zu protokollieren. Diese Vorgehensweise nennt man „racial profiling“. In China wird diese Technologie auch dazu genutzt, um Alarm bei den Behörden zu schlagen, sollten in kürzerer Zeit mehr Uigurinnen und Uigureren an einem Ort protokolliert werden als die Tage davor³². In Zukunft könnte sich in diesem Bereich auch automatisierte Strafverfolgung bilden, indem automatisiert die Aktivitäten von Personen analysiert werden und erkannte Straftaten samt Bildmaterial und Personeninformationen an die Behörden weitergeleitet werden. Auch am Arbeitsplatz wird künstliche Intelligenz bereits eingesetzt, um die Arbeiterschaft zu überwachen. Dafür wird ein Programm auf dem Arbeitscomputer des Arbeitnehmers installiert, welche durch die Bewegung der Maus die arbeitende Person erkennen kann. Anhand von Daten wie zum Beispiel die Zeit, in denen die Maus und Tastatur untätig sind oder die Analysierung gerade offener Programme bzw.

30 Vgl. IQ 27, Vgl. IQ 28

31 Vgl. IQ 29

32 Vgl. IQ 30

Browsertabs kann erkannt werden, ob ein Arbeitnehmer gerade arbeitet oder andere Aktivitäten ausübt. Auch automatische Screenshots, welche an den Arbeitgeber gesendet werden, sind dabei nicht unüblich. Der Einsatz solcher Programme ist jedoch in der EU rechtswidrig da dies gegen die Datenschutzgrundverordnung verstößt³³.

10. Die Programmierung eines künstlich neuronalen Netzwerks

Erster Versuch: neuronales Netzwerk mit XOR-Gatter

Neuroph wird von einigen Projekten, wie unter anderem „Geogebra“ verwendet. Dieses Programm vereinfacht das Erstellen und Anlernen von neuronalen Netzwerken deutlich.

Beim nachstehenden Versuch wird ein Datensatz mit einem XOR-Gatter verwendet, welches ein Gatter mit zwei Eingängen beschreibt, bei dem der Ausgang logisch 1 ist, wenn an nur einem Eingang 1 anliegt und am anderen 0. Der Datensatz für das Netzwerk ist also folgendermaßen aufgebaut:

Input1	Input2	Ouput1
0.0	0.0	0.0
1.0	0.0	1.0
0.0	1.0	1.0
1.0	1.0	1.0

Abbildung 3: Darstellung Datensatz des ersten Versuchs

³³ Vgl. IQ 31

Danach wird ein klassisches neuronales Netzwerk aufgebaut, welches mit folgendem Code-Abschnitt zusammengesetzt ist:

```
//Definiere ein neues Neuronales Netzwerk in neuroph mit BackPropagation als Lerntyp und definiere
//die 3 Layer welche ich verwenden werde.
NeuralNetwork<BackPropagation> neuralNetwork = new NeuralNetwork<BackPropagation>();
Layer inputLayer = new Layer();
Layer hiddenLayer = new Layer();
Layer outputLayer = new Layer();

//Hinzufügen der InputNeuronen in den Inputschicht
inputLayer.addNeuron(new InputNeuron());
inputLayer.addNeuron(new InputNeuron());

//Hinzufügen der Neuronen in der verborgenen Schicht
hiddenLayer.addNeuron(new Neuron());
hiddenLayer.addNeuron(new Neuron());
hiddenLayer.addNeuron(new Neuron());
hiddenLayer.addNeuron(new Neuron());

//Hinzufügen eines Neurons in die Ausgabeschicht
outputLayer.addNeuron(new Neuron());

//Hinzufügen der gerade erstellten Layer in unser Neuronales Netzwerk
neuralNetwork.addLayer(index: 0, inputLayer);
neuralNetwork.addLayer(index: 1, hiddenLayer);
neuralNetwork.addLayer(index: 2, outputLayer);

//Verbindungen der verschiedenen Layer über die ConnectionFactory automatisch erstellen lassen
ConnectionFactory.fullConnect(neuralNetwork.getLayerAt(index: 0), neuralNetwork.getLayerAt(index: 1));
ConnectionFactory.fullConnect(neuralNetwork.getLayerAt(index: 1), neuralNetwork.getLayerAt(index: 2));

//Die Neuronen definieren welche für den Input und Output des Netzwerkes verantwortlich sind
neuralNetwork.setInputNeurons(inputLayer.getNeurons());
neuralNetwork.setOutputNeurons(outputLayer.getNeurons());
```

Abbildung 4: Aufbau des neuronalen Netzwerks im Code

Nach diesem Schritt wird das Netzwerk als Datei ausgegeben um dieses in Neuroph Studio zu importiert und zu veranschaulichen. Das Netzwerk sieht in NeurophStudio dann folgendermaßen aus:

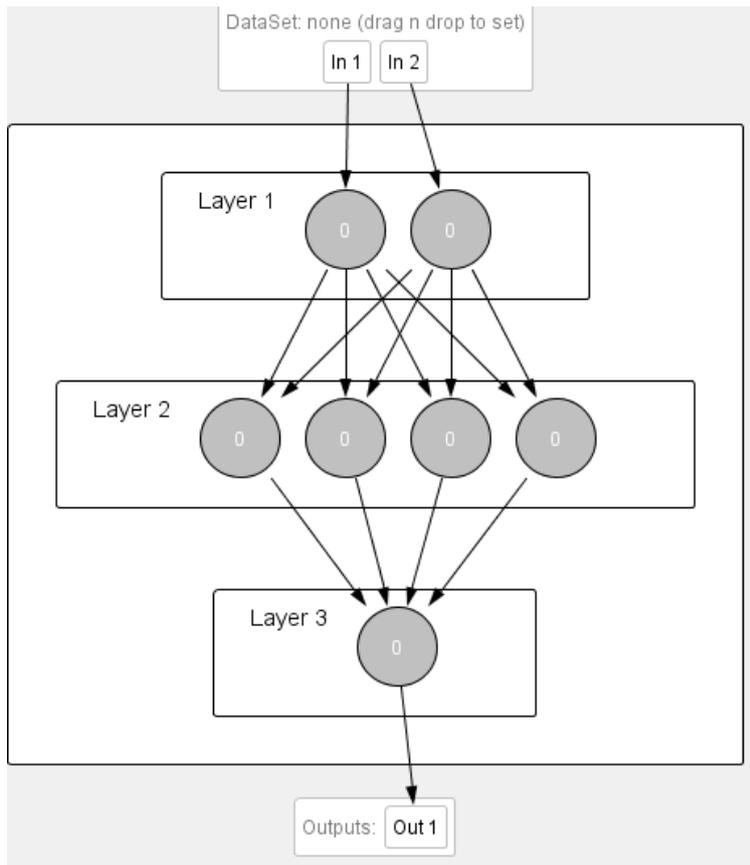


Abbildung 5: Visualisierung des neuronalen Netzwerks

Das Neuronale Netzwerk wird mit der Lernmethode Backpropagation, welche auch Fehlerrückführung genannt wird, antrainiert. Bei dieser Lernmethode werden die verschiedenen Inputs in das Netzwerk eingegeben und mit einem vordefinierten Erwartungswert verglichen. Es muss also bereits per Hand von einer Person im Datensatz die Ausgangssituation beschrieben worden sein. Bei einem Durchlauf wird die Ausgabe des Netzwerkes mit der gewünschten Ausgabe verglichen. Die Differenz des Ausgabewertes des Netzwerkes und der erwünschten Ausgabe ergibt dabei die Fehlerrate des Netzwerkes. Ist der Fehler zu groß, wird der Fehler über die Ausgabe zur Eingabeschicht zurück propagiert und die Gewichtungen der Neuronenverbindungen werden, abhängig auf ihren Einfluss des Fehlers, neu gewichtet. So kann sich das Netzwerk an die gewünschte Ausgabesituation anpassen.

```

//Erstellen und einstellen der backpropagation für mein neuronales netzwerk
BackPropagation backPropagation = new BackPropagation();
backPropagation.setMaxError(0.01);
backPropagation.addListener(this);

//Erstellen eines XOR Datensatzes welches an das Netzwerk angelernt wird
DataSet dataSet = new DataSet( inputSize: 2, outputSize: 1);
dataSet.add(new DataSetRow(new double[]{0, 0}, new double[]{0}));
dataSet.add(new DataSetRow(new double[]{0, 1}, new double[]{1}));
dataSet.add(new DataSetRow(new double[]{1, 0}, new double[]{1}));
dataSet.add(new DataSetRow(new double[]{1, 1}, new double[]{0}));

//Die Lernmethode an das Netzwerk übergeben und das Lernen starten
neuralNetwork.setLearningRule(backPropagation);
backPropagation.learn(dataSet);

```

Abbildung 6: Darstellung Code-Abschnitt mit Backpropagation und Datensatz

Nach meist 7 Durchläufen dieser Lernmethode ist das Netzwerk mit dem XOR Datensatz so gut vertraut, dass es mit einer sehr niedrigen Fehlerrate die richtige Ausgabe trifft.

```

Running NeuralNetwork now...
5. iteration | Total network error: 0.25%
#####
Evaluating Trained Network
Input: 0.0 : 0.0 | Desired Output: 0.0 | Output: 0.0
Input: 0.0 : 1.0 | Desired Output: 1.0 | Output: 1.0
Input: 1.0 : 0.0 | Desired Output: 1.0 | Output: 1.0
Input: 1.0 : 1.0 | Desired Output: 0.0 | Output: 0.0
#####

```

Abbildung 7: Darstellung des Ergebnisses

Zweiter Versuch: Zahlenerkennung mit einem künstlich neuronalen Netzwerk

Beim abschließenden Projekt dieser Arbeit handelt es sich um ein neuronales Netzwerk, welches Zahlen in Bildern erkennen soll. Für diesen Versuch benötigt man einen Datensatz, der Merkmale (in diesem Versuch verschiedenen Bilder) enthält. Der für den Versuch verwendete Datensatz zeigt dem Netzwerk wie gewisse Zahlen aussehen. Dieser Datensatz wird aus Bildern aus dem GitHub Repository, welcher im Quellenverzeichnis unter IQ32 aufzufinden ist, erstellt. Der Import der Bilder erfolgt in

das NeurophStudio importiert und wird in ein TrainingsSet für das neuronale Netzwerk umgewandelt. Dieses TrainingSet wird dann von NeurophStudio wieder exportiert, um dieses dann im Code abrufen zu können.

Daraufhin wird ein neuronales Netzwerk mit einem Tool der Neuroph Library erstellt, welches das Netzwerk automatisch anhand von Informationen wie den Dimensionen der Eingabebilder, dem Farbspektrum, den Namen der Ausgabeneuronen, der Anzahl an Neuronen pro versteckter Schicht sowie der Übertragungsfunktion generiert.

```
//Eine Liste welche für Neuroph angibt wieviel Neuronen sich in den 3 verschiedenen hiddenLayers befinden.
ArrayList<Integer> hiddenLayers = new ArrayList<Integer>();
hiddenLayers.add(50);
hiddenLayers.add(30);
hiddenLayers.add(20);

//Eine Liste der Namen für die Ausgangsneuronen, welche dann z.B sagen es ist eine 1, oder in einem anderen Beispiel: Es ist ein Hund
ArrayList<String> labels = new ArrayList<>();
for(int i = 0; i < 10; i++) {
    labels.add(String.valueOf(i));
}

//Definieren eines automatisch generierten NeuronalesNetzwerk welches für die Verarbeiten von Bildern
// mit den Dimensionen 20x20 ausgelegt ist.
NeuralNetwork<BackPropagation> network = ImageRecognitionHelper.createNewNeuralNetwork( label: "DigitsRecognition",
    new Dimension( width: 20, height: 20), ColorMode.BLACK_AND_WHITE, labels, hiddenLayers, TransferFunctionType.SIGMOID);
ImageRecognitionPlugin iR = (ImageRecognitionPlugin)network.getPlugin(ImageRecognitionPlugin.class);
```

Abbildung 8: Generieren eines neuronalen Netzwerkes anhand von den vorher definierten Informationen

Danach wird wieder die BackPropagation Methode verwendet, um dem Netzwerk die Merkmale der verschiedenen Zahlen anzutrainieren. In diesem Beispiel wird das TrainingSet nun extern als Datei geladen.

```
//Laden des mit NeurophStudio erstellte Lernset
DataSet trainset = DataSet.load("train.tset");

//Setzen des Lernsets und starten des Lernprozesses
network.setLearningRule(backPropagation);
network.learn(trainset);
```

Abbildung 9: Laden des Datensatzes und Antrainieren des Netzwerkes

Zuletzt wird noch ein Code-Abschnitt eingebaut, welcher ein Bild mit dem Namen „input.png“ aus dem Ausführungsordner bezieht und dieses vom Netzwerk berechnen lässt.

```
//Laden der Datei "input.png" im Quellordner und berechnen.
System.out.println("Now trying to recognize image:");
try {
    HashMap<String, Double> output = iR.recognizeImage(new File( pathname: "input.png"));
    System.out.println("My guess: "+getNearestToOne(output));
} catch(Exception exception) {
    exception.printStackTrace();
}
```

Abbildung 10: Berechnung des externen Bildes (input.png)

Das Berechnen von Bildern, welche nicht im TrainingsSet vorhanden sind, funktioniert nur bedingt gut, da dieses Netzwerk nur für eine kurze Zeit mit einer geringen Anzahl an Bilder pro Zahl antrainiert wird.

Im folgenden Abbild kann nochmal in Betrieb betrachtet werden, wie das Netzwerk die Zahl 5 in einem Bild erkennt, welches nicht im TrainingSets vorhanden ist.

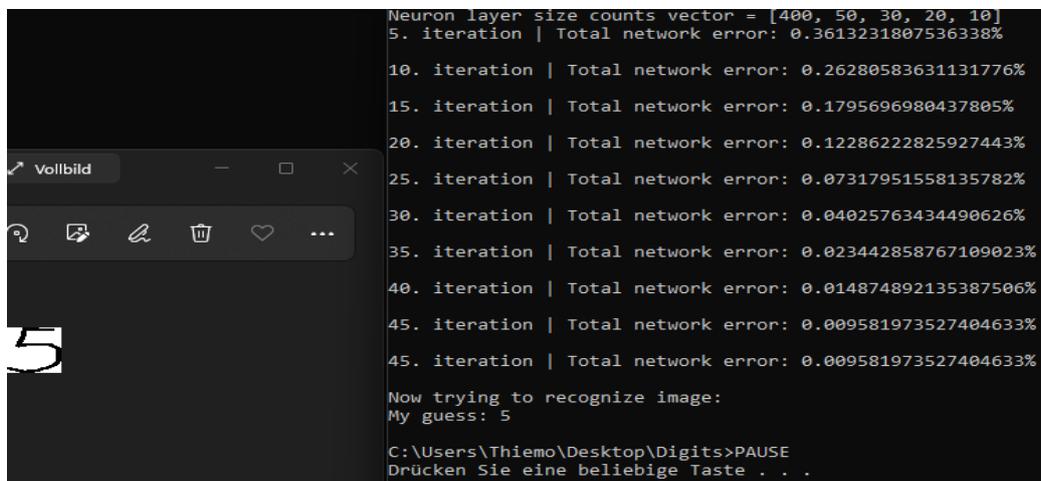


Abbildung 11: Links: Zeichnung der Zahl 5 (input.png); rechts: Antrainieren des Netzwerks, zeigen des Ergebnisses

11. Schlussfolgerung

Unsere Gesellschaft und die technische Entwicklung haben sich in den letzten Jahren massiv geändert. Das was noch vor Jahren unvorstellbar war, wird nun Wirklichkeit. Ob es Segen oder Fluch ist, werden wir in ein paar Jahren begreifen. Die Faszination künstliche Intelligenzen zu schaffen bleibt.

Trotzdem ist es für uns alle gut, zu überlegen, in welchen Bereichen wir die künstlichen Intelligenzen wirklich einsetzen wollen. Wenn wir uns über moralische Grenzen hinwegsetzen und den Mensch zur Maschine machen, könnte es auf Sicht auch unser Verderben bedeuten.

Dazu sollten wir uns überlegen, wem wir die Macht geben, zu bestimmen, wo wir uns dieser Techniken bedienen.

12. Abbildungsverzeichnis

Abbildung 1: Eigene Zeichnung zum Aufbau eines neuronalen Netzwerkes.....	6
Abbildung 2: Eigene Zeichnung eines rekurrenten Netzwerkes	11
Abbildung 3: Darstellung Datensatz des ersten Versuchs.....	23
Abbildung 4: Aufbau des neuronalen Netzwerkes im Code.....	24
Abbildung 5: Visualisierung des neuronalen Netzwerkes	25
Abbildung 6: Darstellung Code-Abschnitt mit Backpropagation und Datensatz	26
Abbildung 7: Darstellung des Ergebnisses.....	26
Abbildung 8: Generieren eines neuronalen Netzwerkes anhand von den vorher definierten Informationen.....	27
Abbildung 9: Laden des Datensatzes und Antrainieren des Netzwerkes.....	27
Abbildung 10: Berechnung des externen Bildes (input.png).....	28
Abbildung 11: Links: Zeichnung der Zahl 5 (input.png); rechts: Antrainieren des Netzwerkes, zeigen des Ergebnisses.....	28

13. Literaturverzeichnis

IQ 1:

<https://3.141592653589793238462643383279502884197169399375105820974944592.eu/pi-wissen/pi-nachkommastellen-rekorde/> (08.10.2021)

IQ 2: MACKOWIAK, Radek: KNN: Natur als Vorbild – Biologische Neuronen. Online im Internet: URL: <https://data-science-blog.com/blog/2015/09/29/knn-natur-als-vorbild-biologische-neuronen/> (10.10.2021)

IQ 3: KRIWET, Hildegard: Maschinelles Lernen – Künstliche Intelligenz im Alltag. Online im Internet: URL: https://www.planet-wissen.de/technik/computer_und_roboter/kuenstliche_intelligenz/maschinelles-lernen-100.html (10.10.2021)

IQ 4: HEZEL, Nico: KNN: Was sind künstliche neuronale Netze? Online im Internet: URL: <https://data-science-blog.com/blog/2015/09/04/knn-was-sind-kuenstliche-neuronale-netze/> (15.10.2021)

IQ 5: PLONER, Patrick Werner / KLAUKIEN, Moritz: Neuronale Netze: Online im Internet: URL: https://www.uibk.ac.at/psychologie/mitarbeiter/leidlmair/neuronale_netze2.pdf (21.10.2021)

IQ 6: WIE FUNKTIONIERT REINFORCEMENT LEARNING? BESTÄRKENDES LERNEN ERKLÄRT. Online Im Internet: URL: <https://databraideo.com/maschinelles-lernen/wie-funktioniert-reinforcement-learning/> (08.11.2021)

IQ 7: Neuronale Netze – komplett: Online im Internet: URL: http://www.chemgapedia.de/vsengine/vlu/vsc/de/ch/13/vlu/daten/neuronalenetze/neuronalenetze1.vlu/Page/vsc/de/ch/13/anc/daten/neuronalenetze/snn5_9.vscml.html (17.11.2021)

IQ 8: Delta-Regel: Online im Internet: URL: https://www2.htw-dresden.de/~boehme/Neuroinformatik/GNI_Prakt_TUI/ni_grundlagen_prak/deltalearning/deltalearning.html (17.11.2021)

IQ 9: Data Science Team: Die einfache Erklärung des Konzepts der Backpropagation: Online im Internet: URL: <https://datascience.eu/de/kuenstliche-intelligenz/wie-der-backpropagation-algorithmus-funktioniert/> (07.03.2022)

IQ 10: Backpropagation: Online im Internet: URL: <http://www.monizone.de/projects/knn/lernverfahren/ueberwachteslernen/backpropagation/index.html> (17.11.2021)

IQ 11: WUTTKE, Laurenz: Künstliche Neuronale Netzwerke: Definition, Einführung, Arten und Funktion: Online im Internet: URL: <http://www.monizone.de/projects/knn/lernverfahren/ueberwachteslernen/backpropagation/index.html> (17.11.2021)

IQ 12: Perzeptron: Online im Internet: URL: <https://www.biologie-seite.de/Biologie/Perzeptron> (17.11.2021)

IQ 13: Feedforward Neural Network: Online im Internet: URL: <https://www.alexanderthamm.com/de/data-science-glossar/feedforward-neural-network/> (21.11.2021)

IQ 14: Dipl. -Ing. (FH) LUBER, Stefan / LITZEL, Nico: Was ist ein rekurrentes neuronales Netz (RNN)?: Online im Internet: URL: <https://www.bigdata-insider.de/was-ist-ein-rekurrentes-neuronales-netz-rnn-a-843274/> (21.11.2021)

IQ 15: Dipl. -Ing (FH) LUBER, Stefan / LITZEL, Nico: Was ist ein Convolutional Neural Network?: Online im Internet: URL: <https://www.bigdata-insider.de/was-ist-ein-convolutional-neural-network-a-801246/> (08.12.2021)

IQ 16: Max: Was sind Schwache KI und Starke KI? Künstliche Intelligenz Erklärung. Online im Internet: URL: <https://simplyki.de/was-sind-schwache-ki-und-starke-ki-kuenstliche-intelligenz-erklaerung/> (15.12.2021)

IQ 17: Künstliche Superintelligenz: Online im Internet: URL: <https://www.alexanderthamm.com/de/data-science-glossar/kuenstliche-superintelligenz/> (20.12.2021)

IQ 18: MASCHINEN SIND SCHNELLER; STÄRKER UND BALD KLÜGER ALS WIR. Online im Internet: URL: <https://www.zeit.de/campus/2015/03/kuenstliche-intelligenz-roboter-computer-menschheit-superintelligenz/seite-2> (20.12.2021)

- IQ 19: Superintelligenz. Online im Internet: URL: <https://www.wikiwand.com/de/Superintelligenz> (20.12.2021)
- IQ 20: DIWO, Matthias: KI: Ihr Wegweiser zu den 4 Arten der Künstlichen Intelligenz. Online im Internet: URL: <https://www.field-service-blog.de/digitale-transformation-im-service/ki-ihr-wegweiser-zu-den-4-arten-der-kuenstlichen-intelligenz> (08.01.2022)
- IQ 21: DVORSKY, George: 8 Possible Alternatives To The Turing Test. Online im Internet: URL: <https://gizmodo.com/8-possible-alternatives-to-the-turing-test-1697983985> (12.01.2022)
- IQ 22: Was ist der Turing-Test? Online im Internet: URL: <https://ai-leaders.de/portfolio/was-ist-der-turing-test/> (20.02.2022)
- IQ 23: Chinesisches Zimmer. Online im Internet: URL: <https://www.philoclopedia.de/was-ist-der-mensch/transhumanismus/chinesisches-zimmer/> (03.03.2022)
- IQ 24: Cleverbot comes very close to passing the Turing Test. Online im Internet: URL: <https://www.cleverbot.com/human> (20.02.2022)
- IQ 25: HERGET, Steffan: Die fünf größten Gefahren von Künstlicher Intelligenz. Online im Internet: URL: <https://www.nextpit.de/kuenstliche-intelligenz-gefahren> (10.01.2022)
- IQ 26: WELCHERING, Peter: Autonome Waffen / KI-Systeme im Militär. Online im Internet: URL: <https://www.deutschlandfunk.de/autonome-waffen-ki-systeme-im-militaer-100.html> (10.01.2022)
- IQ 27: HERBERMANN, Dirk Jan: Killerroboter auf dem Vormarsch – Können die UN sie stoppen? Online im Internet: URL: <https://www.handelsblatt.com/politik/international/ruestung-killerroboter-auf-dem-vormarsch-koennen-die-un-sie-stoppen/27883914.html?ticket=ST-2194059-OYuYdbb0Xga9D16MNEY3-ap1> (10.01.2022)
- IQ 28: SOLMECKE, Christian: Krieg der Zukunft: UN erlaubt Killer-Roboter als autonome Waffe | Anwalt Christian Solmecke. Online im Internet: URL: https://www.youtube.com/watch?v=KahQj_TYR8 (10.01.2022)

IQ 29: pte: Überwachung mittels KI nimmt weltweit zu. Online im Internet: URL: <https://computerwelt.at/news/topmeldung/ueberwachung-mittels-ki-nimmt-weltweit-zu/> (13.01.2022)

IQ 30: Dr. KOCH, Andreas: Überwachung durch künstliche Intelligenz am Beispiel von China. Online im Internet: URL: https://ai.hdm-stuttgart.de/downloads/student-white-paper/Sommer-2019/Ueberwachung_durch_kuenstliche_Intelligenz_in_China.pdf (13.02.2022)

IQ 31: FANTA, Alexander: Gewerkschaften warnen vor KI-Überwachung am Arbeitsplatz. Online im Internet: URL: <https://netzpolitik.org/2021/bossware-gewerkschaften-warnen-vor-ki-ueberwachung-am-arbeitsplatz/> (13.02.2022)

IQ 32: white-gecko: NeuroNumber Trainset. Online im Internet: URL: <https://github.com/white-gecko/NeuroNumber/tree/master/Trainset> (09.03.2022)