

Pädagogische Hochschule Schwyz

Master in Fachdidaktik

Medien und Informatik

Machine Learning for Teachers

Evaluation und Entwicklung von Lehr- und Lernmaterialien
zum Thema Künstliche Intelligenz für Lehrpersonen ab Sekundarstufe 1

Vorgelegt von:

Thomas Zurfluh

Gibraltarstrasse 32

6003 Luzern

E-Mail: thomas.zurfluh@stud.phsz.ch

Matrikelnummer: 05-642-558

Master in Fachdidaktik Medien und Informatik

Herbstsemester 2021 bis Frühlingsemester 2022

Betreuungsperson: Prof. Dr. Beat Döbeli Honegger

Datum: 24.07.2022

Inhaltsverzeichnis

Abbildungsverzeichnis	4
Abkürzungsverzeichnis	5
Einleitung	6
1 Theorie	9
1.1 Begriffsklärung	9
1.1.1 Künstliche Intelligenz	11
1.1.2 Maschinelles Lernen	14
1.1.3 Künstliche Neuronale Netze	16
1.1.4 Deep Learning	20
1.2 KI-Kompetenzen für Lernende ab der Sekundarstufe 1	24
1.2.1 Wahrnehmung	25
1.2.2 Repräsentation und Schlussfolgerung	26
1.2.3 Maschinelles Lernen	27
1.2.4 Interaktion	29
1.2.5 Gesellschaftliche Auswirkungen	30
1.3 Bestehende Methoden und Tools zur Förderung von KI-Kompetenzen	31
1.3.1 Wahrnehmung	31
1.3.2 Repräsentation und Schlussfolgerung	32
1.3.3 Maschinelles Lernen	33
1.3.4 Interaktion	35
1.3.5 Gesellschaftliche Auswirkungen	35

2	Methodik.....	37
2.1	Literaturrecherche.....	38
2.2	Entwicklung und Überarbeitung von Lehr- und Lernmaterialien	38
2.3	Experteninterviews.....	39
3	Ergebnisse.....	40
3.1	Ergebnisse aus der Literaturrecherche.....	40
3.1.1.	Systematische Übersichtsarbeiten	40
3.1.2.	Grundlagenliteratur KI, ML und KNN	41
3.1.3.	Unterrichtseinheiten zu ML mit KNN	43
3.2	Experteninterviews.....	45
3.3	Entwicklung und Überarbeitung von Lehr- und Lernmaterialien	47
3.3.1.	TRApp und TRApp unplugged.....	48
3.3.2.	Mehrlagiges Perzeptron für Filmempfehlungen	50
3.3.3.	Perzeptron unplugged	53
4	Diskussion.....	56
4.1	Zusammenfassung der entwickelten Lehr- und Lernmaterialien	56
4.1.1.	Empfehlungssysteme.....	56
4.1.2.	Be-greifbares Perzeptron.....	57
4.1.3.	Bilder erkennen	57
4.2	Zusammenfassung der Experteneinschätzung.....	57
4.3	Limitationen und weiterführende Forschung	59
5	Fazit.....	60
6	Literaturverzeichnis	62

Anhang	68
A Forschungs- und Entwicklungsprozess der vorliegenden Masterarbeit	68
B Interviewleitfaden.....	69
C Überlegungen zur Vereinfachung des lernenden Perzeptrons	70
C.1 Künstliches Neuron mit einem Eingabewert.....	70
C.2 Lernrate und Aktivierungsfunktion	72
C.3 Schwellenwert.....	75
C.4 Ein künstliches Neuron mit mehr als 2 Eingabewerten.....	78
C.5 Zusammenfassung.....	81
D Perzeptron unplugged Prototyp 1 und 2	84
E Perzeptron unplugged final 3 und 4.....	85

Abbildungsverzeichnis

Abbildung 1 Überschneidungen zwischen den Kernbegriffen KI, ML, KNN und DL	9
Abbildung 2 Symbolische Wissensrepräsentation mit Tau Prolog	13
Abbildung 3 Biologisches Neuron	16
Abbildung 4 Künstliches Neuron mit drei Eingängen	16
Abbildung 5 Lineares binäres Klassifikationsproblem mit zwei Eingaben	18
Abbildung 6 Architektur eines Deep-Learning-Modells.....	21
Abbildung 7 Bilderkennung mit einem künstlichen neuronalen Netz	22
Abbildung 8 Die fünf grossen Ideen der künstlichen Intelligenz nach AI4K12	25
Abbildung 9 Ausschnitt aus Labels des «Hexapawn Educable Robot».....	33
Abbildung 10 Iterativer Entwicklungsprozess neuer Lernaufgaben	38
Abbildung 11 Folie zur Erklärung von TRApp mit Screenshots.....	49
Abbildung 12 Simulation der Abweichung im TRApp-Algorithmus	50
Abbildung 13 Prototyp eines Movie-Recommendere auf Basis eines KNN	52
Abbildung 14 Schwellenwertfunktion.....	74
Abbildung 15 Sigmoidfunktion mit $a = 5$	74
Abbildung 16 ReLU-Aktivierungsfunktion.....	74
Abbildung 17 Inhibition von x_1 mit Bias	77
Abbildung 18 Inhibition von x_1 ohne Bias.....	77
Abbildung 19 Konjunktion (AND) mit Bias	77
Abbildung 20 Konjunktion (AND) ohne Bias	77
Abbildung 21 Antivalenz (XOR) mit Bias	78
Abbildung 22 Training mit 3 Inputs.....	78
Abbildung 23 Filmklassen als Punkte im kartesischen Koordinatensystem.....	81

Abkürzungsverzeichnis

AGI.....	<i>Artificial General Intelligence</i>
AI	<i>Artificial Intelligence</i>
AI4K12	<i>AI education for K-12</i>
DL	<i>Deep Learning</i>
K-12.....	<i>Kindergarten bis 12. Schuljahr</i>
KI	<i>Künstliche Intelligenz</i>
KNN	<i>Künstliches neuronales Netz</i>
ML.....	<i>Maschinelles Lernen</i>
ml4t.....	<i>Machine Learning for Teachers</i>
MLP	<i>Mehrlagiges Perzeptron</i>
NLP	<i>Natural Language Processing</i>

Einleitung

Die Vermittlung von Kompetenzen im Bereich der künstlichen Intelligenz (KI) ist ein weltweit schnell wachsender Forschungsbereich. In der Bildung vom Kindergarten bis zur Oberstufe gewinnt der Bereich daher zunehmend an Bedeutung (Zhou et al., 2020). Mit dem Begriff *KI* sind meist Systeme gemeint, die Daten mithilfe eines Modells interpretieren, das durch biologische neuronale Netze inspiriert wurde (vgl. Kapitel 1.1.1). Diese künstlichen neuronalen Netze (KNN) werden mit immer grösseren Datenmengen auf immer schnelleren Rechnern trainiert, um immer beeindruckendere Ergebnisse zu generieren. Ein aktuelles Beispiel ist ein Sprachmodell, dem es gelang, zwei KI-Forscher von seinem Bewusstsein zu überzeugen (Lemoine, 2022); ein weiteres ist das Open-Source-Projekt Craiyon¹, das es der Öffentlichkeit erlaubt, Bilder aus einer Textbeschreibung zu erschaffen.

Da solche Systeme in Form von Sprachassistenten, Empfehlungssystemen und selbstfahrenden Autos im Alltag immer präsenter werden, stellt sich die Frage, wie die Schule auf diese Entwicklungen reagieren soll. Das Schweizer Staatssekretariat für Bildung schreibt in seinem Bericht zur «künstlichen Intelligenz in der Bildung», die für die Nutzung von KI erforderlichen Fähigkeiten müssten sowohl in der Schulzeit als auch durch lebenslanges Lernen erworben werden, um eine Polarisierung der Gesellschaft zu vermeiden (SBFI, 2019). Trotz dieser Forderungen und der Tatsache, dass es international bereits mehrere Vorstösse gibt, die nötigen Kompetenzen fest in den entsprechenden Lehrplänen zu verankern (Kim et al., 2021), fehlen dahingehende Bestrebungen bisher in der Schweiz.² Gemäss den Kriterien von Schwill (1993) lässt sich jedoch leicht begründen, weshalb es sich bei KNN um eine fundamentale Idee der Informatik handelt, die ihren Platz im Unterricht verdient:

- **Horizontalkriterium:** KNN können in verschiedenen Bereichen vielfältig angewendet werden. Beispiele sind Spracherkennung, -verarbeitung und -erzeugung,

¹ <https://www.craiyon.com>.

² Ausnahme ist der Luzerner Lehrplan mit dem Wahlfach MINT. Dort wird der Begriff *KI* explizit zweimal erwähnt (MI.6.A.1.a und MI.6.A.1.B).

Bildererkennung, -verarbeitung und -erzeugung, autonome Systeme; Suchmaschinen, Empfehlungssysteme.

- **Vertikalkriterium:** Lehrmittel zu maschinellem Lernen (ML) und KNN sind bereits für den Kindergarten vorhanden (Liukas, 2019) und zeigen auf, dass das Thema auf jedem intellektuellen Niveau vermittelt werden kann.
- **Zeitkriterium:** Das erste mathematische Modell eines künstlichen Neurons stammt aus dem Jahr 1943 von McCulloch und Pitts (Loiseau, 2022). Seither wurden etliche künstliche Netzarchitekturen entwickelt, die in der Lage sind, Probleme zu lösen, von denen bis zu ihrer Entstehung davon ausgegangen wurde, dass es dafür menschliche Intelligenz benötigt (Schmidhuber, 2015). Das Tempo aktueller Errungenschaften zeigt, dass KNN auch längerfristig relevant bleiben werden (vgl. Kapitel 1.1.4).
- **Sinnkriterium:** KNN werden zunehmend in nutzerorientierte Technologien integriert. Die Algorithmen der gängigen Plattformen können jedoch für die Nutzenden undurchsichtig sein, die oft nicht erkennen, dass sie mit KI interagieren (Long & Magerko, 2020). Um KI-Systeme selbstbestimmt und effektiv zu nutzen, brauchen Lernende ein grundlegendes Verständnis für die Funktionsweise dieser Algorithmen.

Damit Lehrpersonen in der Lage sind, KI und KNN als Unterrichtsthema zu behandeln, müssen diese die zugrundeliegenden Konzepte verstehen. Lindner und Berges haben im Jahr 2020 die Präkonzepte deutscher Informatiklehrpersonen in der Ausbildung untersucht, um ein adäquates Weiterbildungsangebot zu produzieren. Dabei stellten sie fest, dass deren Konzepte insbesondere in Bezug auf technische Aspekte fehlerbehaftet und unvollständig sind (Lindner & Berges, 2020). Um eine Vermittlung inhaltlicher Fehlkonzepte zu vermeiden, braucht es somit nebst methodisch-didaktisch aufbereitetem Unterrichtsmaterial auch ein Vertiefungsangebot für Lehrpersonen. Diese Masterarbeit soll beide Aspekte auf der Webseite Machine Learning for Teachers (ml4t)³ vereinen. Darauf sollen Lernende die Konzepte hinter

³ <https://www.ml4t.ch/>.

KI, ML und KNN mit anregenden Lernaufgaben und interaktiven Übungen erfahren können. Ein ausführlicher didaktischer Kommentar soll Lehrpersonen dabei helfen, einen auf die individuellen Anforderungen ihrer Klasse passenden Lernweg aus den Materialien zusammenzustellen.

Die für diese Masterarbeit entwickelten Materialien wurden im Rahmen von Experteninterviews evaluiert und iterativ überarbeitet. In diesem Prozess wurde deutlich, welche Bedeutung der fachlichen Auseinandersetzung zukommt. Diese Erkenntnis kann für Studierende des Masters *Fachdidaktik Medien und Informatik* von Interesse sein, die ebenfalls eine Entwicklungsarbeit zu einem aktuellen Informatikthema wie Quantencomputer oder Blockchain-Technologie ins Auge fassen. Ohne eine umfassende fachliche Vertiefung und Anwendung wäre die fachdidaktische Verdichtung der entwickelten Lernmaterialien nicht möglich gewesen.

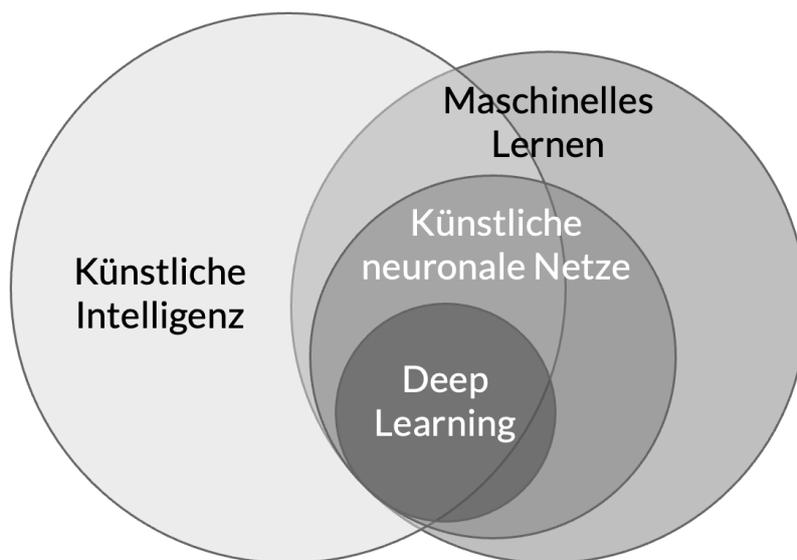
Das erste Kapitel (Theorie) der vorliegenden Masterarbeit richtet sich an Personen, die sich mit der Webseite ml4t auseinandergesetzt haben und ihr Hintergrundwissen in den Bereichen *KI, ML* und *KNN* vertiefen möchten. In den Kapiteln *Methodik, Ergebnisse* und *Diskussion* soll exemplarisch aufgezeigt werden, wie bei der Entwicklung von Lehr- und Lernmaterialien zu aktuellen oder zukünftigen Informatikthemen vorgegangen werden kann und welche Stolpersteine es zu beachten gilt. Das Fazit am Ende dieser Arbeit bietet einen Rück- und Ausblick auf die Beschäftigung des Autors mit dem Thema *KI* und soll Lesende dazu inspirieren, sich ebenfalls von aktuellen technologischen Entwicklungen begeistern zu lassen.

1 Theorie

Der theoretische Hintergrund der vorliegenden Arbeit wird in einen fachwissenschaftlichen Teil zur Begriffsklärung sowie in zwei fachdidaktische Teile aufgegliedert und richtet sich an interessierte Informatiklehrpersonen, die KNN im Unterricht behandeln möchten. Nachdem im ersten Teil definiert wird, wofür der Begriff *KI* in der vorliegenden Arbeit stehen soll, wird im folgenden Kapitel darauf eingegangen, welche Kompetenzen Lernende ab der Sekundarschule für den selbstbestimmten Umgang mit KI-Systemen benötigen. Im dritten Teil werden die Methoden aus einer Auswahl von Lehrwerken zur Entwicklung der genannten Kompetenzen beschrieben.

Abbildung 1

Überschneidungen zwischen den Kernbegriffen *KI*, *ML*, *KNN* und *DL*



1.1 Begriffsklärung

Das Ziel des folgenden Kapitels ist die Klärung und Trennung der Begriffe *künstliche Intelligenz (KI)*, *maschinelles Lernen (ML)*, *künstliche neuronale Netze (KNN)* und *Deep Learning (DL)*. Abbildung 1 stellt dar, in welcher Beziehung die Konzepte hinter den vier Begriffen zueinander stehen. Da KI sowohl der öffentlichkeitswirksamste als auch der

schwammigste und am weitesten gefasste Begriff ist (Krohn et al., 2020), finden sich in der Literatur häufig Mengendiagramme, welche das ML mit seinen beiden Unterkategorien als Teilmenge der KI darstellen. Gemäss Definition, die in der vorliegenden Arbeit für KI verwendet wird, passt diese Einteilung jedoch nicht mehr. Dass auch in der Fachwelt keine Einigkeit über die Begriffshierarchie herrscht, zeigt sich in der aktuellen Diskussion des englischen Wikipedia-Eintrags zum Maschinellen Lernen:

As of 2020, many sources continue to assert that ML remains a subfield of AI. Others have the view that not all ML is part of AI, but only an 'intelligent subset' of ML should be considered AI.⁴

Aus Sicht des Autors dieser Masterarbeit lässt sich der aktuelle Disput um den Begriff *KI* – wie bereits viele zuvor (vgl. Newell, 1982) – auf ihre Gründung als eigene Disziplin der Informatik zurückführen, als John McCarthy im Jahr 1956 das Ziel formulierte, Maschinen zu finden, welche Probleme lösen, die bis anhin dem Menschen vorbehalten waren. Eine weitere Erklärung findet sich bei Krohn et al.:

In der Praxis und vor allem in der Presse wird <KI> ... dazu verwendet, um jede herausragende Fähigkeit einer Maschine (eines Computers) zu beschreiben. ... Wenn eine KI-Fähigkeit zur Normalität wird (wie das Erkennen handgeschriebener Ziffern, was in den 1990ern bahnbrechend war ..., wird der Spitzname <KI> von der Presse üblicherweise fallengelassen. (Krohn et al., S. 62)

Da die herausragenden maschinellen Errungenschaften im Moment fast alle unter Einsatz maschineller Lernverfahren mit tiefen KNN erreicht werden, ist es nicht verwunderlich, dass KI und DL in der Presse häufig synonym verwendet werden (Krohn et al., 2020). Zum Überbegriff *KI* gehören jedoch auch Konzepte wie die *Artificial General Intelligence (AGI)*, starke KI oder Singularität, die Maschinen beschreiben, welche die Fähigkeit besitzen, jede intellektuelle Aufgabe zu lernen, die ein Mensch ausführen kann (Kurzweil, 2005). In der Fachwelt herrscht weder Einigkeit darüber, wann diese starke KI erreicht werden soll, noch, welche Architektur eine solche Maschine aufweisen müsste.⁵

⁴ https://en.wikipedia.org/wiki/Machine_learning#Artificial_intelligence. (22.07.2022)

⁵ https://en.wikipedia.org/wiki/Artificial_general_intelligence. (22.07.2022)

1.1.1. Künstliche Intelligenz

Der Begriff *KI* wurde im historischen Kontext von verschiedenen Personengruppen für unterschiedliche Konzepte verwendet (Konecny, 2020). Als Gründungsveranstaltung der Artificial Intelligence als akademisches Fachgebiet gilt das sechswöchige Sommerprojekt am Dartmouth College in New Hampshire im Jahr 1956, das von John McCarthy ins Leben gerufen wurde (McCarthy et al., 2006). Eines der Ziele der Tagung war herauszufinden, wie Maschinen dazu zu bringen sind, Probleme zu lösen, die bisher Menschen vorbehalten waren:

An attempt will be made to find how to make machines ... solve kinds of problems now reserved for humans. (McCarthy et al., 2006, S. 12)

Diese zeitabhängige Definition lässt sich rück- und vorausblickend immer noch auf den Begriff *KI* anwenden.⁶ Ein Beispiel ist der Erfolg des Schachcomputers Deep Blue von IBM, welcher im Jahr 1997 den russischen Schachweltmeister Garri Kasparow in mehreren Partien schlug. Der amerikanische Schach-Trainer Bruce Pandolfini betitelte das Ereignis als «The Historic Chess Match between Man and Machine», nach welchem die Welt Computer nie wieder auf dieselbe Weise betrachten würde (Pandolfini, 1997). Deep Blue gelang dieser Sieg allerdings mithilfe von Brute Force. Das heisst, das Programm überprüfte jeden möglichen Spielzug, bevor ein Ergebnis ausgegeben wurde. Bei einem Spiel wie Schach mit <nur> 10^{43} Stellungen brauchte es dazu jedoch keinen besonders intelligenten Algorithmus, sondern vor allem einen für die damalige Zeit besonders leistungsstarken Computer. Bis zum Jahr 2016 wurde daher davon ausgegangen, dass die besten Go-Spieler einem Computer überlegen sein müssten, da kein Supercomputer der Welt die 10^{170} möglichen Spielzüge des chinesischen Brettspiels in einer menschlichen Lebensspanne durchsuchen und beurteilen könnte. Als es dem Programm AlphaGo im März 2016 gelang, den amtierenden Go-Weltmeister Lee Sedol zu besiegen, wurde in den Medien abermals vom Erfolg der KI berichtet (Nüesch, 2016).

⁶ Eine chronologische Sammlung mit <intelligenten Maschinen> und technologischen Errungenschaften findet sich beispielsweise in Pickover (2021).

Wie die beiden Beispiele zeigen, liesse sich eine Geschichte der KI in Bezug auf die von den KI-Systemen erfolgreich durchgeführten Aufgaben schreiben. Im Gegensatz dazu steht die Dimension der intellektuellen Funktionen, die für ein intelligentes System notwendig sind: Repräsentation, Problemlösungsmethoden, Mustererkennung, Wissenserwerb usw., die als *Physiologie intelligenter Systeme* bezeichnet werden könnten (Newell, 1982). Der Artikel von Newell zeigt, dass sich die Forschung bereits vor dreissig Jahren nicht einig darüber war, welche intellektuellen Domänen zur KI gezählt werden sollen. Dass dieser Diskurs auch im Jahr 2022 noch nicht abgeschlossen ist, zeigt ein Tweet von Yann LeCun, der 2018 gemeinsam mit Yoshua Bengio und Geoffrey Hinton den Turing Award für seine Arbeit im Bereich *DL* erhielt (Ormond, 2019):

I never called what I was working on AI (AI was supposed to designate ‘symbolic’ methods). Then around 2013, the public and the media became interested in deep learning & *they* called it AI. We could not explain that AI people didn't view DL as AI. Because it made no sense.⁷

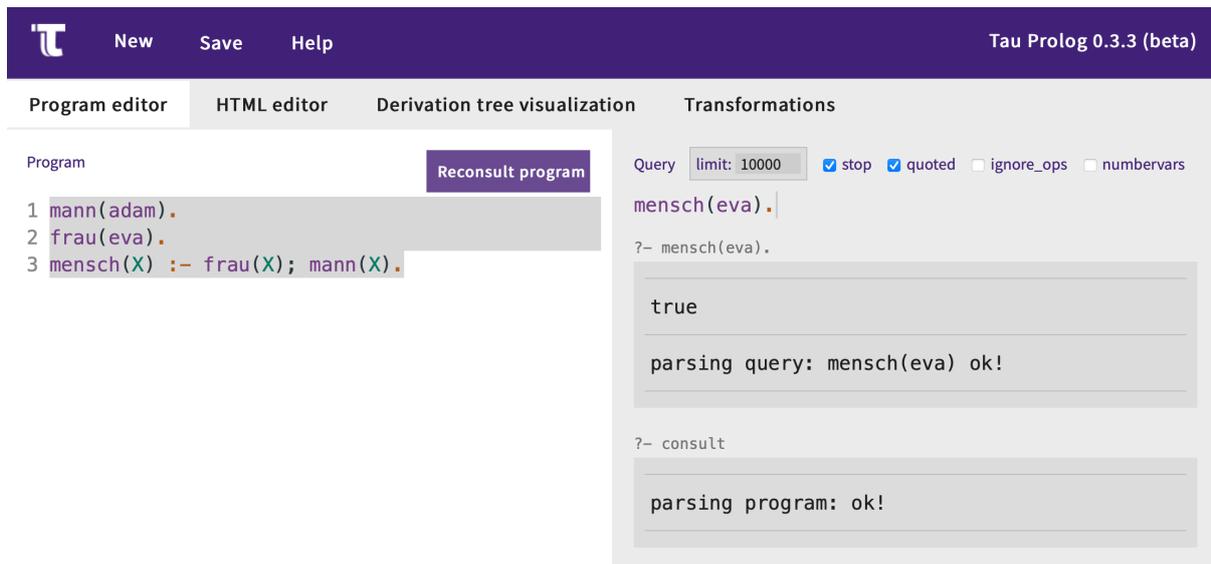
Mit symbolischen Methoden sind jene Verfahren gemeint, die auf Logik und Suche beruhen und aufgrund ihrer Darstellung für den Menschen lesbar sind. Ein Beispiel ist das logische Programmierparadigma, dessen bekannteste Sprache Prolog⁸ in den 1970er-Jahren entwickelt und später für die Erstellung von Expertensystemen verwendet wurde (Newell, 1982). Ein Expertensystem besteht aus einer Wissensbasis und der Möglichkeit, dieses Wissen nutzbar zu machen (Haffner, 2017). In Prolog kann beispielsweise als Wissensbasis definiert werden, dass *Adam* und *Eva* den beiden Kategorien *Mann* und *Frau*, und dass beiden Geschlechterkategorien der Oberkategorie *Mensch* zugeordnet werden sollen. Das System ist nun in der Lage, aus der Wissensbasis zu schliessen, dass Eva ein Mensch sein muss (Abbildung 2).

⁷ <https://twitter.com/ylecun/status/1503719133506158596>.

⁸ Für das Ausprobieren von ProLog im Browser eignet sich <http://tau-prolog.org/sandbox/>. Tau Prolog ist ein vollständig in JavaScript implementierter Prolog-Interpreter.

Abbildung 2

Symbolische Wissensrepräsentation mit Tau Prolog



Anmerkung. Screenshot aus der Webanwendung Tau Prolog (<http://tau-prolog.org/sandbox/>). Links befindet sich das Fenster des Programmcodes (Wissensbasis). Durch einen Klick auf *Consult Program* wird das Programm gestartet und die Wissensbasis analysiert (*parsing program*). Anschliessend kann in der rechten Spalte eine Anfrage (*prolog goal*) gestellt werden und das Expertensystem antwortet, ob die Aussage korrekt ist. Im vorliegenden Beispiel wird das Expertensystem gefragt, ob Eva der Klasse *Mensch* angehört. Da in der Wissensbasis in Zeile 2 definiert wird, dass Eva ein Element aus der Klasse *Frau* ist und in Zeile 3 steht, dass Elemente der Klasse *Frau* oder *Mann* auch der Klasse *Mensch* angehören, folgert das Programm korrekt, dass Eva ein Element aus der Klasse *Mensch* sein muss.

Im Gegensatz zu den symbolischen Verfahren stehen die numerischen Verfahren zur Mustererkennung. In diesem Bereich wurden in den letzten zwanzig Jahren vor allem Fortschritte durch die Verwendung tiefer KNN in Verbindung mit schnellerer Hardware und besser verfügbaren grossen Datenmengen erzielt (Schmidhuber, 2015).

Die Definition von McCarthy (1956) hat somit bis in die Gegenwart nicht an Gültigkeit verloren, da sie beschreibt, dass KI in der Lage sein soll, jene Probleme zu lösen, bei denen *bis jetzt* davon ausgegangen wurde, dass menschliche Intelligenz benötigt wird. Zu diesen gehört beispielsweise die Computerlinguistik (Engl. Natural Language Processing, kurz NLP), die sich mit der Erkennung, Verarbeitung und Erzeugung geschriebener und gesprochener Sprache durch Maschinen auseinandersetzt. Die Fortschritte im Bereich *NLP* mithilfe umfangreicher Sprachmodelle werden in den Medien beispielsweise mit *A. I. is Mastering Language* betitelt

(Johnson & Iziev, 2022). Für die Fortschritte, die zu solchen medialen Superlativen führen, sind in den letzten Jahren ausschliesslich tiefe KNN verantwortlich (Ekman, 2021), ein Teilbereich des ML.

1.1.2. Maschinelles Lernen

Gemäss Pickover (2021) lässt sich der Begriff *Machine Learning (ML)* auf den amerikanischen Elektroingenieur und Informatiker Arthur Lee Samuel zurückführen. In seinem Aufsatz «Some Studies in Machine Learning Using the Game Checkers» aus dem Jahr 1959 erläuterte er:

Das «Programmieren von Computern, die aus Erfahrung lernen» ... [könnte] einen grossen Teil der erforderlichen ausdrücklichen und aufgabenspezifischen Programmierung und Anweisung überflüssig machen. (Pickover, 2021, S. 99)

Diesem Ansatz folgend geht es bei ML nicht darum, die Muster zur Lösung eines Problems selbst zu erkennen und in einem Programm zu implementieren, sondern darum, die Rohdaten geschickt vorzubereiten, um Features zu extrahieren, die zu den datenmodellierenden Algorithmen passen (Krohn et al., 2020). Dabei werden grundsätzlich je nach Ziel und Ausgangslage drei Ansätze des ML unterschieden:

- Überwachtes Lernen (Supervised Learning)
- Unüberwachtes Lernen (Unsupervised Learning)
- Verstärkendes Lernen (Reinforcement Learning)

Beim überwachten Lernen besteht der Datensatz aus bekannten Ein- und Ausgabepaaren. Typischerweise wird es für Regressions- oder Klassifikationsprobleme eingesetzt. Bei der Regression ist die Ausgabe eine kontinuierliche Variable, z. B. die Preisentwicklung eines Produkts oder einer Aktie. Bei der Klassifikation bilden die Ausgabewerte eindeutige Kategorien. Das Ziel beim überwachten Lernen besteht darin, das Modell eine Funktion finden zu lassen, welche die Eingabewerte verwendet, um sich dem gewünschten Ausgabewert zu nähern (Krohn et al., 2020). Die gefundene Funktion wird nach der Trainingsphase mithilfe eines Testdatensatzes evaluiert. Die Qualität des Modells wird nach einer endlichen Anzahl an

Trainingsiterationen durch die Grösse des Fehlers eruiert. Durch das geschickte Vorbereiten der Rohdaten und Auswählen der passenden Modelle kann diese Qualität beeinflusst werden. Probleme des unüberwachten Lernens unterscheiden sich von jenen des überwachten Lernens durch das Fehlen eines zuvor bekannten Ausgabewerts im Trainingsdatensatz. Das Ziel des unüberwachten Lernens besteht somit darin, ein Muster in den Eingabedaten zu entdecken. Beispielsweise wird das unüberwachte Lernen bei kollaborativen Empfehlungsdiensten (Recommender Systems) eingesetzt, um Nutzer in Clustern mit ähnlichen Verhaltensmustern zu gruppieren (Alpaydm, 2021).

Ein Beispiel für verstärkendes Lernen ist das autonome Fahren oder Erlernen eines Computerspiels⁹. Bei Problemen führt ein Programm (genannt Agent) in einer Umgebung eine Reihe von Aktionen aus. Die relevanteste Unterscheidung gegenüber dem überwachten und unüberwachten Lernen besteht darin, dass die Aktionen, die der Agent ausführt, die Informationen beeinflussen, welche die Umgebung dem Agenten anbietet – das heisst, «der Agent erhält eine direkte Rückmeldung über die Aktion, die er unternimmt» (Krohn et al., 2020, S. 68). Beim autonomen Fahren kann beispielsweise ein Belohnungs- und Bestrafungssystem implementiert werden. Das Fahrzeug ist in diesem Beispiel der Agent, der in seiner einfachsten Form zwei Aktionen ausführen kann: Beschleunigen und Steuern. Mit jeder Einheit, die er näher an das definierte Ziel rückt, wird eine kleine Belohnung ausgeschüttet. Kommt das Fahrzeug vom vorgegebenen Weg ab, erhält es eine kleine und bei der Berührung eines anderen Verkehrsteilnehmers eine grosse Bestrafung. Den Bedingungen entsprechend verhält sich der trainierte Agent so, dass er die Belohnungen maximiert und die Bestrafungen minimiert. Die Schwierigkeit besteht im Beispiel des autonomen Fahrens darin, Belohnung und Bestrafung so zu wählen, dass der Agent zwar versucht, bestrafendes Verhalten

⁹ Auf YouTube finden sich zahlreiche Beispiele, die anschaulich zeigen, wie ein anfänglich «dummes» Computerprogramm dank einem einfachen Belohnungssystem lernt, Computerspiele zu meistern. Ein bekanntes Beispiel ist MarI/O, das trainiert wird, das Nintendo-Spiel Super Mario World zu spielen: <https://youtu.be/RcSjInPiXPL>.

zu vermeiden, aber die Belohnung gross genug ist, dass er den Verkehrsfluss nicht durch zögerndes Fahrverhalten unterbricht.

1.1.3. Künstliche Neuronale Netze

Eine entscheidende Inspiration für die Architektur intelligenter Systeme war bereits in den 1940er-Jahren die biologische Nervenzelle, auch *Neuron* genannt. Ein biologisches Neuron (Abbildung 3) ist auf die Aufnahme und Verarbeitung von Signalen spezialisiert. Die eingehenden elektrischen Erregungen werden von den Dendriten an den Zellkörper weitergeleitet. Übersteigt die Erregung einen gewissen Schwellenwert, entlädt sich die Spannung über das Axon. An den Axonterminalen werden die Signale über Synapsen chemisch oder elektrisch an Dendriten der nächsten Nervenzelle weitergereicht (Zitzler, 2017).

Abbildung 3

Biologisches Neuron

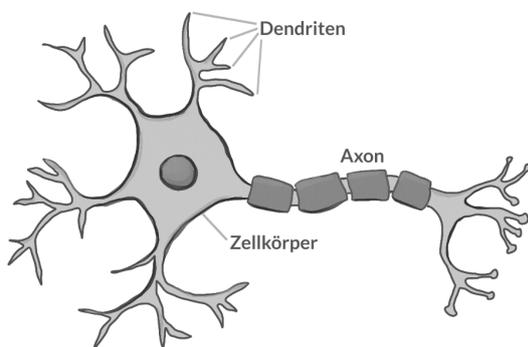
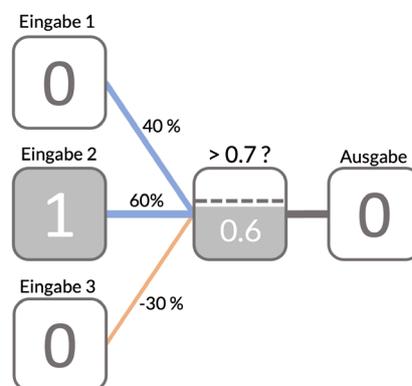


Abbildung 4

Künstliches Neuron mit drei Eingängen



Anmerkung. Eigene Darstellungen zur visuellen Gegenüberstellung biologischer und künstlicher Neuronen. Angelehnt an Zitzler, 2017.

Das erste mathematische Modell eines künstlichen Neurons war die Threshold Logic Unit (TLU) von Warren S. McCulloch und Walter H. Pitts Jr im Jahr 1943 (Loiseau, 2022). Die TLU wird durch folgende Eigenschaften definiert (vgl. Loiseau, 2022). Eine TLU hat ...

- eine binäre Ausgabe (1 = feuern, 0 = nicht feuern).
- eine Reihe von erregbaren binären Eingangssignalen.

- einen einzigen hemmenden Eingang. Wenn dieser eingeschaltet ist, kann das Neuron nicht feuern.
- einen Schwellenwert θ (theta). Wenn die Summe der Eingangssignale grösser als der Schwellenwert ist, feuert das Neuron (Ausgabe = 1). Andernfalls bleibt die Ausgabe 0.

Abbildung 4 zeigt ein künstliches Neuron mit drei Eingängen, das durch die TLU inspiriert ist. Es besitzt ebenfalls einen Schwellenwert. Im Beispiel beträgt der Schwellenwert $\theta = 0.7$ und sowohl Ein- als auch Ausgänge können im gezeigten Beispiel nur binäre Werte einnehmen. Im Unterschied zur TLU sind in Abbildung 4 die Eingänge gewichtet. Das heisst, dass jeder Eingabewert mit einem dazugehörenden Gewicht multipliziert wird. Diese veränderbaren Gewichte wurden im Neuronen-Modell von Frank Rosenblatt erstmals im Jahr 1957 vorgestellt (Loiseau, 2022). Dies hat zur Folge, dass einige Eingaben einen grösseren Einfluss auf die Ausgabe des Neurons haben können als andere. Das künstliche Neuron aus Abbildung 4 besitzt zwei positiv gewichtete Eingänge, wovon der zweite Eingabewert stärker gewichtet (60 %) wird als der erste (40 %). Ausserdem können durch die Ergänzung von Rosenblatt die Eingabewerte durch eine negative Gewichtung auch einen hemmenden Einfluss haben (Eingabe 3 wird mit – 30 % gewichtet).

Frank Rosenblatt nannte sein Modell des künstlichen Neurons *Perzeptron*, eine Wortschöpfung aus den Worten *Perception* (Engl. *Wahrnehmung*) und *Neuron*. Rosenblatts Ziel war es, eine Maschine zu bauen, welche wie das Gehirn Eingabedaten mithilfe der Neuronenverbindungen ohne explizite Instruktion korrekt klassifiziert. Eine frühe Version des Perzeptrons wurde beispielsweise trainiert, um mithilfe von Fotozellen Bilder von Männern und Frauen den beiden Klassen zuzuordnen.¹⁰ Trainieren heisst im Falle dieses Beispiels Folgendes:

- Ein Bild wird von 20 x 20 Photozellen aufgenommen (= 400 Eingabedaten).

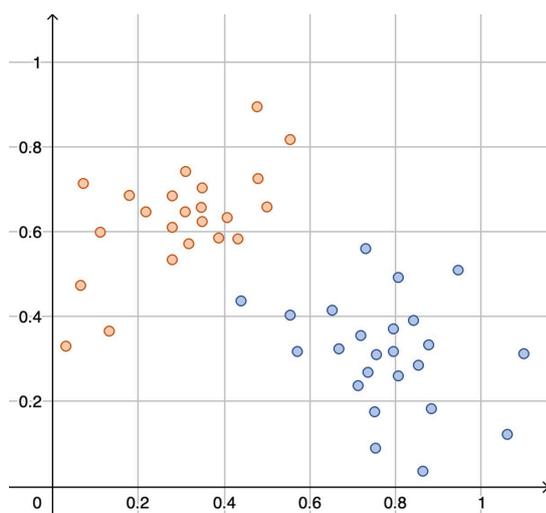
¹⁰ https://youtu.be/cNxadbrN_aI (PseudoIntellectual, 2008).

- Die Eingabedaten werden unter Berechnung einer Ausgabe durch gewichtete Verbindungen im Netzwerk weitergereicht, z. B. *Frau*.
- Falls die Ausgabe korrekt ist, werden die Verbindungen (Gewichte) nicht verändert. Bei einer falschen Ausgabe werden die Gewichte leicht angepasst.
- Dieser Prozess wird mit unterschiedlichen Bildern so lange wiederholt, bis die Fehlerrate einen akzeptablen Wert erreicht.

Die relevanteste Errungenschaft von Rosenblatt war die Entwicklung des überwachten Lernalgorithmus, der das künstliche Neuron in die Lage versetzte, die korrekten Gewichte direkt aus den Trainingsdaten selbst herauszufinden (Loiseau, 2022). Durch die Anpassung der Gewichte findet der Lernalgorithmus die Parameter einer Hyperebene¹¹, welche die Klassen im n-dimensionalen Raum voneinander trennen. Im oben beschriebenen Beispiel mit 400 Eingabedaten wäre dies ein 400-dimensionaler Raum. Etwas einfacher lässt sich eine Klassifizierung mit zwei Eingabedaten darstellen (Abbildung 5). Um dieses Klassifikationsproblem zu lösen, muss der Lernalgorithmus die Parameter der geraden Linie zur Separation der beiden Klassen finden.

Abbildung 5

Lineares binäres Klassifikationsproblem mit zwei Eingaben



¹¹ Ein einfacher Punkt in 1D, eine gerade Linie in 2D, eine regelmässige Ebene in 3D usw.

Die Einfachheit und Effizienz dieses Lernalgorithmus für linear trennbare Probleme sind zwei der Hauptgründe, warum das Perzeptron in den späten 1950er- und frühen 1960er-Jahren so populär wurde. Diese öffentliche Begeisterung veranlasste Rosenblatt jedoch dazu, die Lernfähigkeit des Perzeptrons zu übertreiben, was zu unrealistischen Erwartungen in der wissenschaftlichen Gemeinschaft und auch in den Medien führte (Loiseau, 2022). Eine vorübergehende Dämpfung der Popularität künstlicher neuronaler Netze wurde durch Marvin Minsky und Seymour Papert eingeleitet, die im Jahr 1969 das Buch «Perceptrons: an introduction to computational geometry» veröffentlichten (Loiseau, 2022). Darin zeigten sie, wie begrenzt die Problemstellungen sind, die mit dem Perzeptron tatsächlich gelöst werden können und dass es die einfache logische XOR-Funktion nicht lernen kann (siehe Kapitel 4.1.2). Vielerorts wird argumentiert, dass die Veröffentlichung dieses Werks und die Demonstration der Grenzen des Perzeptrons den *KI-Winter* der 1980er-Jahre ausgelöst haben (Krohn et al., 2020; Loiseau, 2022; Pickover, 2021).

Newell argumentierte 1982, dass die symbolischen Verfahren den Disput um das vorherrschende KI-Paradigma gegen die Mustererkennung gewannen, weil sich der Ansatz Rosenblatts zu stark auf die Bild- und Spracherkennung fokussierte, während die symbolischen Verfahren Maschinen zur Lösung von Problemen entwickelten, die Theoreme beweisen, Spiele gegen Menschen gewinnen und Rätsel lösen konnten.

To the AI community the intellectual depth of the tasks performed by the pattern-recognition systems seemed relatively trivial compared with the problem-solving tasks done by the programming systems. (Newell, 1982, S. 12)

Zusammen mit der Kritik von Minsky und Papert führte dies dazu, dass die Forschungsgelder im Bereich der KNN gekürzt wurden und das Thema lange Zeit aus der öffentlichen und akademischen Aufmerksamkeit verschwand (Schmidhuber, 2015). Der entscheidende Wendepunkt kam erst im Jahr 2012, als ein KNN mit vielen versteckten Schichten (DL-

Modell) bei der ILSVRC¹² alle bisherigen Benchmarks der herkömmlichen ML-Algorithmen signifikant übertreffen konnte (Kahn et al., 2020).

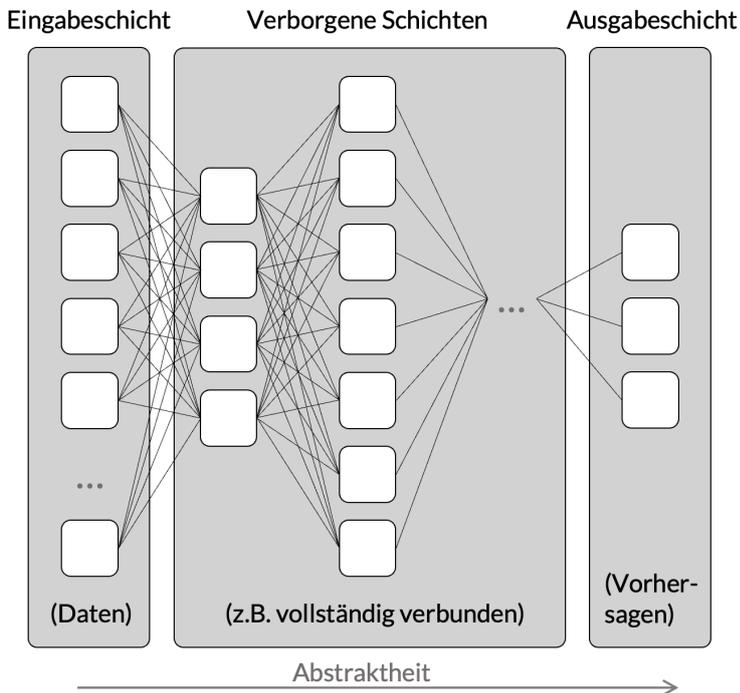
1.1.4. Deep Learning

Ein KNN ist eine Sammlung von künstlichen Neuronen (Abbildung 4) die so angeordnet sind, dass sie einander Informationen senden und voneinander empfangen (Abbildung 6). Ein solches Netz mit mehreren verborgenen Schichten wird als *Deep Learning-Netz* bezeichnet (Krohn et al., 2020). Deep Learning (DL) ist somit eine Klasse von Algorithmen für ML, die mehrere Schichten von Recheneinheiten (künstlichen Neuronen) verwenden, wobei jede Schicht ihre eigene Darstellung der Eingabedaten erlernt. Diese Repräsentationen werden von späteren Schichten auf hierarchische Weise kombiniert und in der Regel mit jeder Schicht weiter abstrahiert (Ekman, 2021). In Abbildung 6 sind jeweils alle künstlichen Neuronen jeder Schicht mit jenen der folgenden Schicht verbunden und die Daten werden von links nach rechts weitergegeben (vollständig verbundenes Feedforward-NN). In der Praxis haben sich jedoch viele unterschiedliche Netzarchitekturen für spezifische Anwendungsfälle bewährt (Schmidhuber, 2015).

¹² Die ILSVRC (ImageNet Large Scale Visual Recognition Challenge) ist ein seit 2010 jährlich stattfindender Wettbewerb, bei dem es um die computergestützte Identifizierung von Objekten in digitalen Bildern geht. Die 14 Millionen Bilder der ImageNet-Datenbank sind in 22 000 Kategorien unterteilt. Bei der Challenge treten hochkarätige Teams gegeneinander an, die mit ihren Algorithmen versuchen, die niedrigste Fehlerfrequenz bei der Kategorisierung zu erzielen. Im Jahr 2012 gelang es dem Gewinnerteam dank des Einsatzes von DL die Fehlerquote von bis dato üblichen 30 % zu halbieren. Daraufhin übernahmen sämtliche Teams diesen Ansatz, welcher kurze Zeit später sogar die durchschnittliche Fehlerquote einer menschlichen Vergleichsgruppe zu unterschreiten vermochte (Daum, 2019).

Abbildung 6

Architektur eines Deep-Learning-Modells

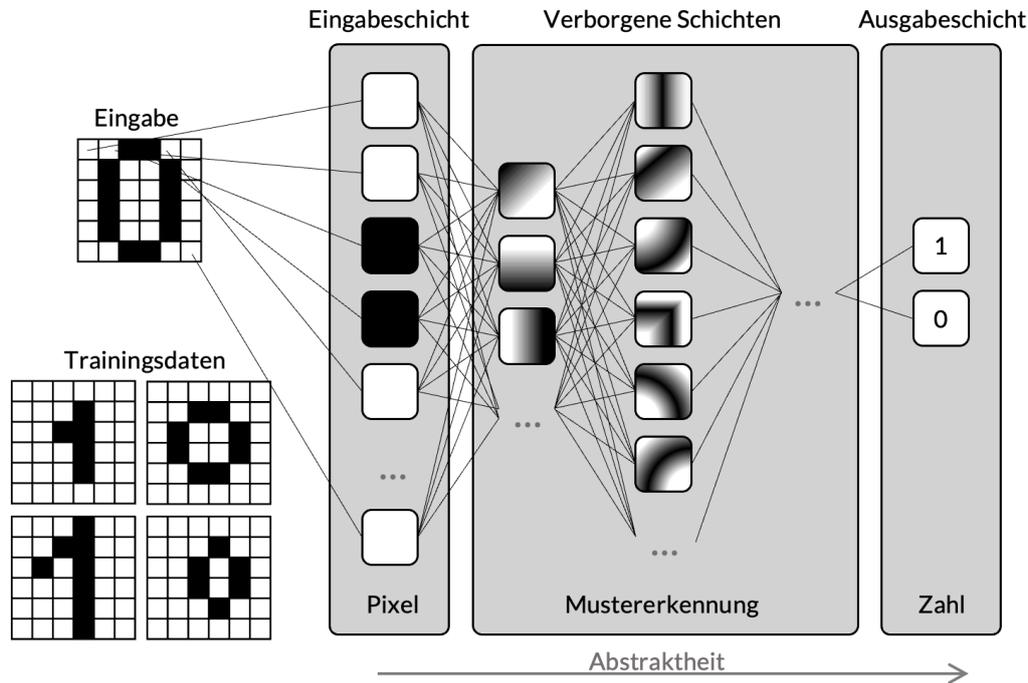


Anmerkung. Eigene Darstellung, angelehnt an Krohn et al. (2020)

Ein anschauliches Beispiel für die Abstraktion, die von einer Schicht zur nächsten stattfindet, ist die Bilderkennung. Abbildung 7 zeigt ein KNN, das als Eingabe ein 6 x 6-Pixelbild erhält und erkennen soll, ob das Bild eine 1 oder eine 0 darstellt. Die Eingabeschicht besteht aus 36 Pixeln, die in diesem Beispiel nur schwarz oder weiss aufnehmen. Die 36 Einzelwerte kombiniert das KNN in diesem Beispiel in der ersten Schicht eher unten oder rechts im Bild, um zu erkennen, ob es oben links mehr dunkle Werte hat. Diese Muster werden anschliessend verbunden und damit weiter abstrahiert. So können in einer tieferen Schicht beispielsweise runde Formen oder Ecken erkannt werden. Da die abstrakten Repräsentationen vollständig autonom anhand der Trainingsdaten gebildet werden, ist es bei tiefen KNN in der Regel nicht mehr nachvollziehbar, welche Merkmale das Modell priorisiert, solange die Vorhersage mit dem erwarteten Wert übereinstimmt.

Abbildung 7

Bilderkennung mit einem künstlichen neuronalen Netz



Anmerkung. Eigene Darstellung, angelehnt an Krohn et al. (2020)

Dies ist auch einer der Gründe, weshalb DL-Algorithmen immer wieder für negative Schlagzeilen sorgen (Ouchchy et al., 2020). Da im Vergleich zu den symbolischen Ansätzen der KI bei DL teilweise nicht nachvollziehbar ist, aufgrund welcher Merkmale ein Modell Entscheidungen trifft, ist ihre Anwendung beispielsweise in der Vorsortierung von Bewerbungen höchst umstritten (Dastin, 2018). Dabei wird ein DL-Modell, das den Bewerbungsprozess beschleunigen soll, trainiert, indem es Muster in Bewerbungsunterlagen sucht, die häufiger zu einer Anstellung führen. Dem Modell wird dabei nicht explizit vorgeschrieben, nach welchen Attributen es die Unterlagen sortieren soll; Merkmale wie Geschlecht oder ethnische Herkunft werden sogar häufig manuell aus den Daten herausgefiltert. Trotzdem ist es möglich, dass das Modell andere Merkmale so kombiniert, dass diese in einer tieferen Schicht aufgrund des voreingenommenen Datensatzes wieder entdeckt werden und so Frauen oder Minderheiten benachteiligt werden (Dastin, 2018). Solche Vorurteile in automatisierten Verfahren führten dazu, dass sich der Forschungszweig

Explainable AI entwickelte, der DL-Modelle mit symbolischen Methoden kombiniert, um deren Entscheidungen nachvollziehbarer zu gestalten (Holzinger, 2018).

Trotz kritischer Stimmen konnte KI dank DL in den letzten zehn Jahren einschneidende Erfolge verzeichnen (Chatterjee, 2022). Diese lassen sich grösstenteils auf zwei Entwicklungen zurückführen: die Datenanhäufung im Internet und das Aufkommen neuer Rechnerarchitekturen, die zahlreiche parallele Rechnungen ermöglichen. Ein aktuelles Beispiel ist Googles neues Sprachmodell¹³ mit 540 Milliarden Parametern. Das Modell ist in der Lage, sprachliche Probleme zu lösen, die bis vor kurzem als dem Menschen vorbehalten galten.¹⁴ Es wurde mit einer Kombination aus englischen und mehrsprachigen Datensätzen trainiert, die Webdokumente, Bücher, Wikipedia-Einträge, Konversationen und GitHub-Code umfassen (Narang & Chowdhery, 2022). Verarbeitet wurden diese Daten mithilfe von Googles Pathways-System mit 6144 TPU-Chips, die für parallele Rechenanwendungen optimiert sind. Obwohl es Forschende gibt, die argumentieren, der DL-Ansatz stosse langsam an seine Grenzen (Marcus, 2022), scheinen diese Grenzen anhand der beinahe wöchentlich erscheinenden Errungenschaften immer weiter ausgelotet zu werden. Brett Winton, Forschungsdirektor der Investmentgesellschaft Ark Invest¹⁵, fasste diese Beobachtung in einem Twitter-Beitrag folgendermassen zusammen:

People confidently state that neural nets will not be able to do X.

And yet at certain dataset-size/compute/model-depth thresholds models prove suddenly and unexpectedly capable (of things where they hadn't shown any previous trend improvement.)¹⁶

¹³ <https://ai.googleblog.com/2022/04/pathways-language-model-palm-scaling-to.html> (Narang & Chowdhery, 2022).

¹⁴ Zum Beispiel das Erklären von Witzen: <https://youtu.be/kea2ATUEHH8>.

¹⁵ https://de.wikipedia.org/wiki/Ark_Invest.

¹⁶ <https://twitter.com/wintonARK/status/1537487274350243840>.

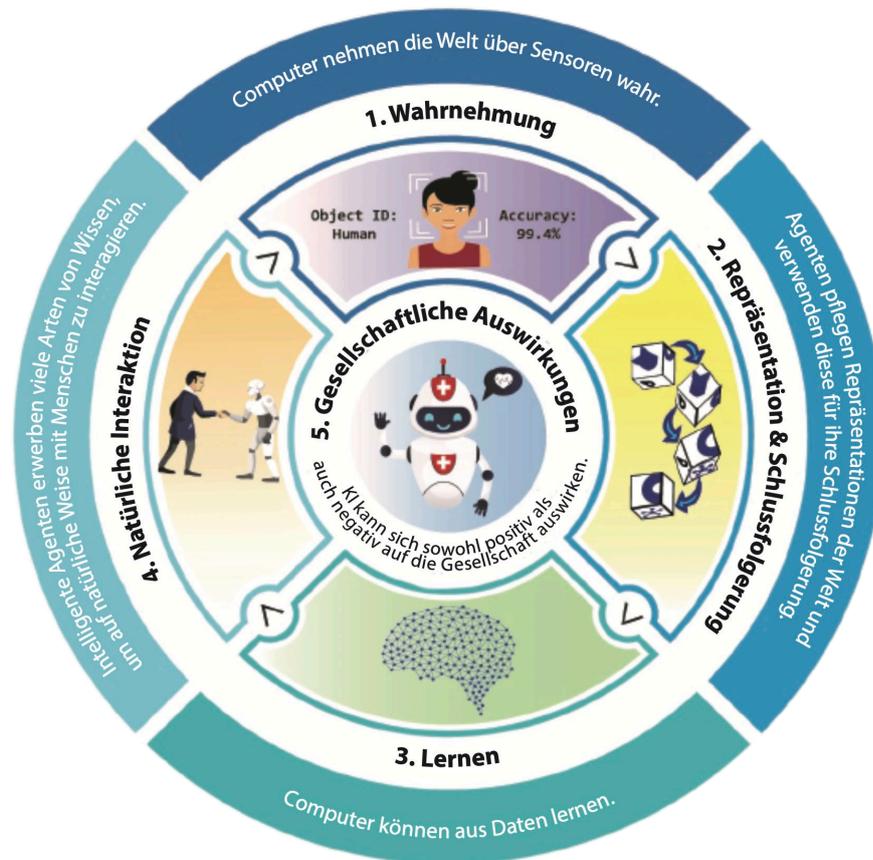
1.2 KI-Kompetenzen für Lernende ab der Sekundarstufe 1

Das folgende Kapitel soll die Frage beantworten, welche Kompetenzen Lernende benötigen, um selbstbestimmt mit KI zu interagieren und sie kritisch zu bewerten. Hiermit sind in der Regel jene Systeme gemeint, die gemäss der Begriffsklärung in Kapitel 1.1.1 in der Lage sind, Probleme zu lösen, von denen bis vor kurzem davon ausgegangen wurde, dass es dafür menschliche Intelligenz benötigt. Diese Definition trifft zum Zeitpunkt der Entstehung der vorliegenden Masterarbeit fast ausschliesslich auf Systeme mit KNN zu. Unter den Begriff *KI-Kompetenzen* fallen somit jene Fähigkeiten und Fertigkeiten, die benötigt werden, um mit diesen KNN-Systemen selbstbestimmt zu interagieren und sie kritisch zu bewerten.

Im Mai 2018 bildeten die Association for the Advancement of Artificial Intelligence (AAAI) und die Computer Science Teachers Association (CSTA) eine gemeinsame Arbeitsgruppe, um nationale Richtlinien für die Informatikausbildung vom Kindergarten bis zur Oberstufe (K-12) in den vereinigten Staaten zu entwickeln (Touretzky et al., 2019). Die Arbeitsgruppe formulierte in einem ersten Entwurf die fünf *Big Ideas of AI*, die als Richtlinien für künftige Lehrpläne dienen sollen: Wahrnehmung, Repräsentation und Schlussfolgerung, Maschinelles Lernen, Interaktion und Gesellschaftliche Auswirkungen (Abbildung 8). Die Formulierung von Schlüsselkonzepten anstelle der Unterteilung in Domänenwissen (z. B. Bild- und Sprachverarbeitung oder Robotik) lässt sich leichter mit der Kompetenzorientierung des Lehrplans vereinbaren. Die Strukturierung des folgenden Kapitels orientiert sich daher an diesen fünf Konzepten. Jedes Unterkapitel enthält eine Beschreibung der Kompetenz und mindestens ein technologisches Beispiel mit möglichen Fragestellungen, um dieses im Unterricht zu thematisieren.

Abbildung 8

Die fünf grossen Ideen der künstlichen Intelligenz nach AI4K12



Anmerkung. Ausschnitt aus deutscher Übersetzung des Big Ideas Posters (AI4K12, 2020).

1.2.1. Wahrnehmung

Wahrnehmung ist gemäss Definition von Touretzky et al. (2019) die Fähigkeit, Bedeutung aus sensorischen Informationen zu extrahieren. Die entsprechende Kompetenz zeigt sich durch ein Verständnis für die Umwandlung der sensorischen Signale in bedeutungsvolle Repräsentationen. Dazu gehört beispielsweise bei der visuellen Wahrnehmung die Kodierung eines Bildes in eine zweidimensionale Pixelmatrix, wobei jedes Pixel eine Zahl ist, welche die Helligkeit eines Bildteils angibt. Die Umwandlung dieser Rohdaten in eine Bedeutung erfolgt in mehreren Stufen, wobei die Merkmale aus den Rohdaten in jeder Stufe zunehmend abstrahiert werden (vgl. Abbildung 7). Damit z. B. die Mimikererkennung für dynamische Gesichtsfiler in Social-Media-Applikationen wie Snapchat funktioniert, durchläuft der

Algorithmus in der Regel drei Phasen: Vorverarbeitung, Merkmalerfassung und -klassifikation (Li & Deng, 2020):

- In der Vorverarbeitung werden die vom Gesicht übermittelten visuellen semantischen Informationen ausgerichtet und normalisiert, damit das tiefe neuronale Netz beim Erlernen der aussagekräftigen Merkmale nicht durch Störvariablen wie unterschiedliche Hintergründe, Beleuchtungen oder Kopfhaltungen beeinflusst wird.
- Das tiefe neuronale Netzwerk versucht anschliessend, Abstraktionen auf hoher Ebene durch hierarchische Architekturen mit mehreren Transformationen und Darstellungen zu erfassen.
- Nach dem Erlernen der Merkmale besteht der letzte Schritt der Mimikererkennung darin, das gegebene Gesicht in einer der vorgegebenen Emotionskategorien zu klassifizieren. Die Zuordnung erlernt das Modell dank tausender durch Menschen beschrifteter Bilder. Dieser Prozess ist rechenintensiv und geschieht vor der Anwendungsphase.

Das Beispiel zeigt, dass sich die 5 Big Ideas nicht isoliert betrachten lassen. Da Wahrnehmung nicht nur die Umwandlung sensorischer Signale in eine digitale Repräsentation beschreibt, muss auch die anschliessende Abstraktion der Sensordaten behandelt werden. Moderne KI-Systeme nutzen für diese Abstraktion häufig tiefe neuronale Netze, die durch ML trainiert wurden. Um somit zu verstehen, wie eine KI ein für den Menschen einfaches Problem wie die Mimikererkennung löst, benötigen Lernende ein Verständnis für eine Reihe maschineller Prozesse.

1.2.2. Repräsentation und Schlussfolgerung

Wissensrepräsentationen modellieren die Welt in einer Weise, die für einen Computer verständlich ist (Long & Magerko, 2020). Die relevantesten beiden Wissensrepräsentationen

sind nach den Autoren von AI4K12 die symbolische und die numerische Repräsentation.¹⁷ Bei symbolischen Repräsentationen werden logische Schlussfolgerungsregeln verwendet (vgl. Abbildung 2), während bei numerischen Repräsentationen komplexe mathematische Funktionen wie KNN zum Einsatz kommen. Gemäss den AI-Literacy-Empfehlungen von Long und Magerko (2020) benötigen Nutzende von KI-Systemen kein tiefgreifendes Verständnis für die Implementierung von Wissensrepräsentationen. Ein konzeptuelles Verständnis von Repräsentationen könnte Lernenden jedoch bei der Erkenntnis helfen, dass in einer Repräsentation der Welt immer etwas Wissen verloren geht (Long & Magerko, 2020). Um die Aktionen oder Schlussfolgerungen eines KI-Systems kritisch hinterfragen zu können, sollten Lernende somit auch den Unterschied zwischen symbolischer und numerischer Repräsentation verstehen, da sie unterschiedliche Fragen aufwerfen:

- «Versteht mich der Sprachassistent nicht, weil ich eine Frau bin und das Sprachmodell mit Männerstimmen trainiert wurde?»
- «Versteht der Sprachassistent den Inhalt meiner Frage nicht?»

Im ersten Fall handelt es sich um eine fehlerhafte numerische Repräsentation, weil das Modell durch einseitige Daten trainiert wurde und einen unerwünschten Bias entwickelte. Das zweite Problem lässt sich möglicherweise mit einer Lücke in der Wissensbasis des Expertensystems erklären, auf das der Assistent zurückgreift.

1.2.3. Maschinelles Lernen

Gemäss den AI4K12-Empfehlungen von Touretzky et al. (2019) sollen Lernende vier ML-Schlüsselkonzepte erklären können: (1) Definition von ML, (2) Funktionsweise von ML, (3) Rolle von Trainingsdaten in ML sowie (4) Anwendung eines gelernten Modells auf neue Daten.

¹⁷ Die Initiative AI4K12 stellt auf ihrer offiziellen Webseite www.ai4k12.org Leitlinien um ihre fünf grossen Ideen der KI zur Verfügung. Diese bieten gemäss den Seitenautoren einen Rahmen, um Autoren von Standards sowie Lehrplanentwickler bei der Festlegung von KI-Konzepten und beim Erwerb grundlegender Kenntnisse und Fähigkeiten für die Klassenstufen zu unterstützen.

- (1) ML ermöglicht es einem künstlichen System, sich Verhaltensweisen anzueignen, ohne dass ein Mensch diese ausdrücklich programmiert. Solche können Vorhersagen eines Wettermodells, Empfehlungen eines Online-Shops oder Entscheidungen eines autonomen Fahrzeugs sein.
- (2) Das Erlernen neuer Verhaltensweisen ergibt sich aus den Änderungen, die der Lernalgorithmus an den internen Darstellungen eines Denkmodells vornimmt, z. B. an den Gewichten eines neuronalen Netzes (vgl. Kapitel 1.1.3). Gemäss Long und Magerko (2020) sind Lernende oft überrascht, dass ML menschliche Entscheidungen erfordert und nicht vollständig automatisiert ist. Um dieses Missverständnis zu überwinden, empfehlen die Autoren eine körperliche Interaktion, die es ihnen erlaubt, die Rolle eines lernenden Agenten einzunehmen.
- (3) Je grösser die Vielfalt der zu erwartenden Verhaltensweisen eines Modells, desto relevanter werden Umfang, Vielfalt und Qualität der Trainingsdaten für die Entwicklung des künstlichen Denkmodells. Ein Beispiel sind Sprachmodelle, die natürliche Sprache erzeugen. Während frühere regelbasierte Modelle der Spracherzeugung in der Regel «Wortsalate» produzierten, sind heutige auf DL basierte Modelle mit hunderten Milliarden Parametern wie GPT-3 in der Lage, täuschend echte Texte im Schreibstil bekannter Autoren nachzuahmen (Elkins & Chun, 2020). Eine mögliche Aufgabe für Lernende könnte darin bestehen zu erläutern, welche Auswirkungen es haben könnte, wenn als Trainingsdatensatz eines Sprachmodells lediglich Bibeltexte verwendet würden und wie eine optimale Ausgewogenheit im Modell erzeugt werden kann.
- (4) Das vierte Schlüsselkonzept betrifft den Übergang von der Lern- zur Anwendungsphase. ML verwendet gekennzeichnete Trainingsdaten, um Schlussfolgerungsmodelle zu erstellen, die eine Klassifizierung oder Vorhersage vornehmen. Während des Trainings wird das Modell auf die Trainingsdaten angewendet und der Lernalgorithmus passt das Modell so an, dass seine Ausgaben

besser mit den Bezeichnungen übereinstimmen. Nach Abschluss des Trainings kann das Modell auf neue Daten angewendet werden, um Probleme zu lösen oder Entscheidungen zu treffen.

1.2.4. Interaktion

In der Literatur finden sich die Begriffe *schwache*, *starke* oder *allgemeine KI*. Derzeitige KI-Systeme sind gut darin, eng gefasste Aufgaben zu lösen, z. B. ein Bild aus einer Beschreibung mit Worten herzustellen.¹⁸ Die Modelle, die für diese Aufgabe trainiert wurden, sind jedoch hoch spezialisiert und nicht in der Lage, eine triviale arithmetische Aufgabe zu lösen. Das neueste Sprachmodell von Google (Narang & Chowdhery, 2022) zeigt zwar, dass grössere Modelle fähig sind, natürlichsprachliche Aufgaben zu lösen, von denen bis vor kurzem davon ausgegangen wurde, dass sie menschliche Intelligenz benötigen (vgl. Kapitel 1.1.4), jedoch wäre das Modell nicht in der Lage, eine Melodie zu erfinden. Eine starke oder allgemeine KI (Engl. Artificial General Intelligence; AGI) verspricht eine allgemeine Denkfähigkeit, die mit jener eines Menschen vergleichbar ist. Ob die gegenwärtige Generation deren Entwicklung noch erleben wird oder kurz bevor steht, wird zwischen Futurist*innen und Wissenschaftler*innen seit den 1960er-Jahren debattiert (Pickover, 2021).

Um selbstbestimmt mit KI-Systemen zu interagieren, sollen Lernende in der Lage sein, den Unterschied zwischen starker und schwacher KI zu beschreiben. Auch sollen sie Problemtypen, bei denen KI ihre Stärken ausspielt, von solchen unterscheiden können, die für KI eine grössere Herausforderung darstellen. Diese Information sollen Lernende nutzen, um zu entscheiden, wann der Einsatz von KI sinnvoll ist und wann menschliche Fähigkeiten genutzt werden sollten (Long & Magerko, 2020). Auch sollen sie in der Lage sein zu erklären, dass Computer Emotionen zwar erkennen, aber nicht erleben können (Touretzky et al., 2019).

¹⁸ Zwei aktuelle Beispiele sind Dall-E 2 von OpenAI (<https://openai.com/dall-e-2/>) und Googles Imagen (<https://imagen.research.google>).

1.2.5. Gesellschaftliche Auswirkungen

Damit Lernende in der Lage sind, Schlagzeilen über neuartige oder zukünftige KI-Systeme einzuordnen, müssen sie sich mit positiven wie negativen gesellschaftlichen Auswirkungen auseinandersetzen (Touretzky et al., 2019). In den Medien wird häufig entweder kritisch oder enthusiastisch über KI berichtet (Ouchchy et al., 2020). Lernende begegnen in den sozialen Medien oder Online-Zeitungen beispielsweise dystopischen Berichten über Killerroboter¹⁹, der Auslöschung von Arbeitsplätzen durch KI²⁰ oder dem bedingungslosen Grundeinkommen dank Robotersteuer²¹. In einem Alter, in dem sich die Lernenden mit ihrer Zukunft auseinandersetzen sollen, können solche Schlagzeilen zu Ängsten oder gar Ohnmacht führen, wenn diese nicht bewusst reflektiert und sachgemäss eingeordnet werden. Die Auswirkungen auf den Arbeitsmarkt sind nur eines der Ethikthemen aus dem Bereich der KI, die Lernende gemäss den Vorschlägen von Long und Magerko (2020) identifizieren und aus verschiedenen Perspektiven beschreiben können sollen:

- Die Auswirkungen der Automatisierung auf den Arbeitsmarkt werden durch KI beschleunigt. Welche Kompetenzen braucht es in einer automatisierten Welt?
- Die Analyse persönlicher Datenspuren wird durch KI verbessert. Welche Folgen hat dies auf das Individuum und die Gesellschaft?
- Die Produktion und Verbreitung von Fehlinformationen werden durch KI vereinfacht.²² Wie lassen sich Filterblasen verhindern und der Wahrheitsgehalt einer Meldung überprüfen?
- Die Besorgnis über die Entwicklung der *Singularität* (vgl. Kapitel 1.1.1) wächst mit jeder Hürde, die KI überwindet. Wie real ist diese Sorge?

¹⁹ <https://www.20min.ch/video/nationalrat-warnt-vor-killerrobotern-kommission-nehme-es-nicht-ernst-625263310404>.

²⁰ <https://www.blick.ch/wirtschaft/wer-um-seinen-job-fuerchtet-kann-aufatmen-roboter-ersetzen-menschen-doch-nicht-ganz-id8401295.html>.

²¹ <https://www.20min.ch/story/schweizer-fordern-eine-strafsteuer-fuer-roboter-453637896314>.

²² Politische Instrumentalisierung der Deep-Fake-Technologie: <https://youtu.be/enr78tJkTLE>.

- Wie werden ethische Entscheidungsstrategien in technische Systeme eingebettet?²³
- Algorithmische Voreingenommenheit (Bias) steht oft in direktem Zusammenhang mit Verzerrungen in Trainingsdatensätzen. Welche Massnahmen zur Vorbeugung können angewandt werden?
- Auch wenn die grundsätzliche Funktionsweise von DL-Algorithmen erklärbar ist, sind viele Modelle zu gross und zu komplex, um deren Entscheidungen im Detail nachzuvollziehen. In welchen Bereichen ist es unbedenklich, solche Blackbox-Algorithmen zu verwenden und wo ist Transparenz unverzichtbar?

1.3 Bestehende Methoden und Tools zur Förderung von KI-Kompetenzen

Wie am Beispiel *Wahrnehmung* in Kapitel 1.2.1 erklärt, lassen sich die Bereiche der fünf grossen KI-Ideen nur bedingt isoliert betrachten. Das folgende Kapitel soll einen Überblick über Tools und Methoden zur Kompetenzförderung in den fünf Bereichen aus Kapitel 1.2 bieten. Weiter wird aufgezeigt, welche Überlegungen zum Ein- oder Ausschluss bestimmter Ansätze für die Unterrichtseinheit geführt haben. Aufgrund der umfangreichen Menge an Neuentwicklungen erhebt die folgende Sammlung keinen Anspruch auf Vollständigkeit.

1.3.1. Wahrnehmung

Als handlungsorientierter Einstieg in die Funktionsweise wahrnehmender Systeme eignet sich Google Teachable Machine²⁴. Mit dem Tool lassen sich sowohl akustische als auch visuelle Signale anschaulich darstellen und mithilfe vortrainierter Modelle für überwachtes Lernen verwenden. Teachable Machine bietet drei Modi: Bilderkennung, Tonerkennung und Körperposenerkennung. In der Trainingsphase werden beispielsweise für ein Projekt zur

²³ Die Webseite Moral Machine erfasst, wie Menschen zu moralischen Entscheidungen stehen, die von Maschinen getroffen werden: <https://www.moralmachine.net/hl/de>.

²⁴ <https://teachablemachine.withgoogle.com>.

Tonerkennung das Miauen einer Katze und das Bellen eines Hundes aufgenommen. Die Audioschnipsel werden den beiden Klassen *Miauen* und *Bellen* zugeordnet. Teachable Machine erlaubt es, das Modell schon mit wenigen Beispielen zu trainieren und anschliessend zu testen. So verstehen Lernende, dass ein Computer den Unterschied zwischen den beiden Tiergeräuschen erst erkennt, wenn er eine Vielzahl beschrifteter Daten erhält. Fehlerhafte Zuordnungen des Modells können ausserdem dazu verwendet werden, um über Limitationen trainierter Modelle und über den Vergleich mit menschlicher Wahrnehmung zu sprechen.

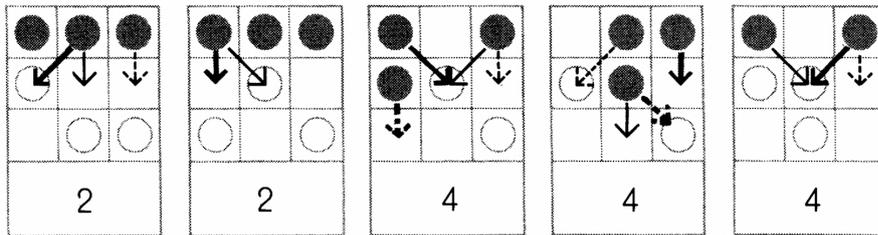
1.3.2. Repräsentation und Schlussfolgerung

Symbolische Wissensrepräsentationen lassen sich für einfache Probleme mit analogen Mitteln darstellen und enaktiv erfahren. Ein bekanntes Beispiel in KI-Unterrichtsmaterialien ist das Bauernschach, Engl. *Hexapawn*²⁵ (Gallenbacher, 2021; Gardner, 1991; Seegerer et al., 2019; Wirth, 1993). Dabei spielt ein menschlicher Spieler gegen einen Agenten, der ausschliesslich durch Regeln gesteuert wird, auf einem 3 x 3 Felder grossen Spielbrett mit Bauernfiguren. Die Wissensbasis für das Verhalten des Agenten besteht aus allen möglichen Spielzuständen mit allen jeweils möglichen Spielzügen des Agenten. Das Wissen zu jedem Zustand kann mithilfe von verschiedenfarbigen Plättchen oder Hölzchen repräsentiert werden. Mit der einfachen Lernregel, dass bei Spielverlust das entsprechende Hölzchen entfernt wird, das den zum Verlust führenden Spielzug repräsentierte, wird der regelbasierte Agent bereits nach wenigen Spielen unschlagbar. Das Spiel eignet sich auch, um die Grundprinzipien des verstärkenden Lernens zu entdecken.

²⁵ Eine spielbare Online-Version findet sich auf <https://www.stefanseegerer.de/schlag-das-krokodil/>.

Abbildung 9

Ausschnitt aus *Labels des «Hexapawn Educable Robot» aus Gardner (1991)*



Da der Fokus der vorliegenden Masterarbeit auf der Vermittlung der Kompetenzen für die selbstbestimmte Interaktion mit KI-Systemen liegt, die ihr Wissen numerisch ²⁶ repräsentieren, wird in den Unterrichtsmaterialien auf eine Vertiefung der symbolischen Repräsentation verzichtet.

Für die Veranschaulichung der numerischen Wissensrepräsentation in KNN gibt es unterschiedliche Ansätze. Shamir und Levin (2021) haben eine programmierbare Lernumgebung auf Scratch-Basis erstellt, die es bereits Lernenden auf Primarstufe erlaubt, eigene Artefakte zu erstellen. Die Wissensrepräsentation lässt sich dank den einblendbaren Variablen nachvollziehen (Shamir & Levin, 2021). Viele didaktische Ansätze sind weniger handlungsorientiert und beschränken sich auf eine vereinfachende Darstellung von KNN (Bieri, 2020; Gallenbacher, 2021; Tadeusiewicz, 2020).

1.3.3. Maschinelles Lernen

Damit Lernende nachvollziehen können, was in einem System passiert, wenn von ML gesprochen wird, reicht es nicht, ein lernendes System wie Google Teachable Machine zu benutzen. Wie im vorherigen Unterkapitel beschrieben, liegt der Fokus der vorliegenden Arbeit auf dem ML mit KNN. Dazu gibt es z. B. die Webapplikation TensorFlow Playground²⁷, die den maschinellen Lernprozess eines KNN visualisiert. Durch die abstrakte Problemstellung

²⁶ vgl. Kapitel 1.2.2.

²⁷ <https://playground.tensorflow.org>.

der Datensätze eignet sich das Tool jedoch eher für die vertiefte Auseinandersetzung ab Sek2-Stufe.

Einen konstruktivistischen Ansatz, um eigene ML-Artefakte herzustellen, wählten die Autoren von SnAIp²⁸ (Michaeli et al., 2020). Mithilfe vorgefertigter Blöcke können Lernende auf der blockbasierten Programmierumgebung Snap!²⁹ einfache Spiele erstellen und die verändernden Parameter beim verstärkenden Lernen mittels Q-Learning beobachten. Der Ansatz eignet sich gemäss den Autoren, um in die Blackbox zu blicken und Lernenden aufzuzeigen, dass ML kein *Hexenwerk* ist (Michaeli et al., 2020). Da vom Q-Learning-Algorithmus jedoch nicht direkt auf die Funktionsweise von KNN schliessen lässt, eignet sich der Einsatz dieser Materialien nicht für die Zielsetzungen der vorliegenden Arbeit.

Weitere vorgefertigte Blöcke des ML für die Snap!-Umgebung finden sich im Projekt eCraft2Learn³⁰ (Kahn et al., 2022). Zwar sind unter den begleiteten Projekten auch vorgefertigte Blöcke für KNN, jedoch werden diese in der blockbasierten Umgebung derart unübersichtlich, dass eine strukturierte textbasierte Umgebung für diese Komplexitätsstufe zu bevorzugen ist. Aus diesem Grund wird in der vorliegenden Arbeit von der Integration beliebiger KNN-Bibliotheken wie TensorFlow³¹ in eine blockbasierte Umgebung abgeraten.

Zuletzt sind Lernumgebungen zu erwähnen, welche die Funktionsweise von KNN auf einzelne künstliche Neuronen herunterbrechen und mit entsprechenden Übungen den Lernprozess erfahrbar machen. Im deutschsprachigen Raum wurden mehrere Unterrichtseinheiten gefunden, die dieses Ziel verfolgten (Janssen, 2020; Lardelli, 2021). Beide Ansätze enthalten sowohl Aufgaben zum analogen Berechnen der Gewichtsveränderungen eines KNN beim überwachten Lernen als auch Übungen mit digitalen Tools zur Veranschaulichung des Lernprozesses bei grösseren Datenmengen. Während bei Janssen (2020) der

²⁸ <https://computingeducation.de/proj-snaip-A/>.

²⁹ <https://snap.berkeley.edu>.

³⁰ <https://ecraft2learn.github.io/ai>.

³¹ <https://www.tensorflow.org>.

Abstraktionsgrad der Aufgaben eher hoch ist und sich daher für Lernende der Sekundarstufe 2 eignet, wird bei Lardelli (2021) Bezug auf den Alltag der Lernenden genommen.

1.3.4. Interaktion

Über *starke KI* lässt sich mit Lernenden anhand von Beispielen aus Science-Fiction diskutieren. Dazu finden sich verschiedene Ansätze, um ins Thema einzusteigen. Für jüngere Lernende eignet sich beispielsweise die Reise von Globi ins Land der Roboter (Bieri, 2020), das auch interessante Fragen für Erwachsene aufwirft. Etwas mehr Tiefe bietet das Buch «Ava im Land der Zukunft» (Tadeusiewicz, 2020). Das Bilderbuch begleitet die Protagonistin *Ava*, die KI überall in ihrem Alltag antrifft und von ihrem Onkel Hintergrundinformationen erhält. Inhaltlich sind die Kapitel dicht und daher eher für Lernende ab Sekundarstufe geeignet, obwohl das Buch für Kinder ab neun Jahren geschrieben wurde.

Um aktiv mit KI-Systemen zu experimentieren, empfehlen Kahn und Winters (2021) programmierbare Lernumgebungen. Dazu gehören Machine-Learning for Kids³², die AI Snap! Bibliothek von eCraft2Learn³³ (Kahn et al., 2020), Cognimates³⁴ (Druga, 2018) sowie ein Projekt mit dem MIT App Inventor (Zhu, 2019), die den gemeinsamen Ansatz verfolgen, eigene intelligente Artefakte in einer blockbasierten Programmierumgebung zu erstellen. Die Lernenden können mit diesen Tools Projekte erstellen, die sehen, hören, lernen, klassifizieren oder vorhersagen und sich so auf praktische Weise mit den Stärken, Schwächen, Konzepten und Fähigkeiten lernender Systeme auseinandersetzen (Kahn et al., 2020).

1.3.5. Gesellschaftliche Auswirkungen

Die Ansätze zur Thematisierung gesellschaftlicher Auswirkungen und ethischer Überlegungen können in zwei Kategorien eingeteilt werden. Auf der einen Seite gibt es eine Fülle an

³² <https://machinelearningforkids.co.uk>.

³³ <https://project.ecraft2learn.eu/>.

³⁴ <http://cognimates.me/home/>.

Unterrichtsmaterialien und frei zugänglichen Webseiten, welche die Wirkung von KI-Systemen auf die Gesellschaft anhand von aktuellen KI-Beispielen thematisieren und passende Fragen zur Diskussionsanregung in der Klasse zur Verfügung stellen.³⁵ Auf der anderen Seite stehen Aktivitäten, die Lernende durch die Entwicklung eigener KI-Artefakte zur Diskussion über ethische Implikationen (z. B. algorithmische Voreingenommenheit) anregen sollen (Payne & Breazeal, 2019).³⁶

³⁵ <https://code.org/oceans>;
<https://www.ki-konkret.de/was-darf-ki.html>;
<https://www.moralmachine.net>.

³⁶ <https://www.media.mit.edu/projects/ai-ethics-for-middle-school/> und die deutsche Übersetzung:
<https://thingminds.ch/de/kikids/>.

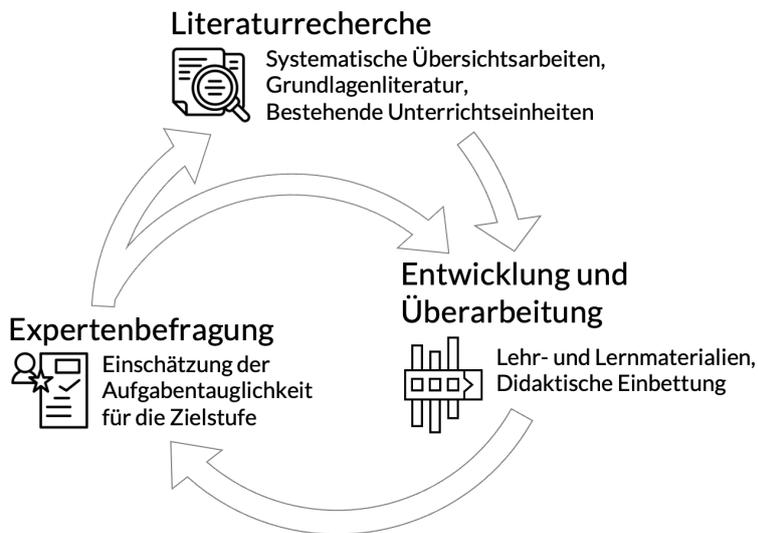
2 Methodik

Die folgenden Kapitel richten sich an Forschende, die eine fachdidaktische Entwicklungsarbeit zu neuartigen Technologien (z. B. Quanteninformatik oder Blockchain-Technologie) planen. Im Kapitel *Methodik* wird der iterative Entwicklungsprozess der Lehr- und Lernmaterialien beschrieben, der mit Experteneinschätzungen und Literaturrecherchen mehrere Durchgänge durchlief. Im Kapitel *Ergebnisse* werden die gesichtete Literatur und Erkenntnisse aus den Experteninterviews zusammengefasst und beschrieben, wie die Unterrichtsmaterialien und didaktische Einbettung entstanden und überarbeitet wurden. In der Diskussion werden der Entwicklungsprozess zusammengefasst und Limitationen aufgezeigt sowie Vorschläge für weiterführende Forschung dargelegt.

Abbildung 10 zeigt das methodische Vorgehen bei der Entwicklung und Überarbeitung der Lehr- und Lernmaterialien. Der Prozess wurde durch eine Literaturrecherche initiiert, aus welcher Ideen für Lernaufgaben der Zielstufe hervorgingen. Nach der ersten Entwicklungsphase wurden die entstandenen Materialien mit Experten auf ihre Praxistauglichkeit überprüft. Dieser Austausch führte wiederum zur Überarbeitung oder erneuten Literaturrecherche. Eine weitere Darstellung dieses Prozesses findet sich in Anhang A. Die Grafik entstand im letzten Drittel der Masterarbeit für die Präsentation im Kolloquium, um die Oszillation des Forschungsschwerpunkts zwischen der fachlichen und der pädagogisch-didaktischen Perspektive aufzuzeigen.

Abbildung 10

Iterativer Entwicklungsprozess neuer Lernaufgaben



2.1 Literaturrecherche

Da in den letzten Jahren im Bereich der KI-Kompetenzen auf der primären und sekundären Bildungsstufe ausgiebig geforscht wurde, fand in einem ersten Schritt eine Literaturrecherche im Snowballing-Verfahren (Wohlin, 2014) zu fachdidaktischen Papers und Lehrmitteln statt. In diesem Prozess wurden mehrere Forschungsarbeiten gefunden, die einen aktuellen Überblick bieten. Viele der in diesen Papieren referenzierten Quellen, Unterrichtsmaterialien und Tools wurden wiederum begutachtet und deren Literaturverzeichnisse studiert. Weitere, insbesondere deutschsprachige Literatur wurde durch Hinweise der befragten Experten und des Betreuers der vorliegenden Arbeit ermittelt.

2.2 Entwicklung und Überarbeitung von Lehr- und Lernmaterialien

Die Webseite Machine Learning for Teachers ³⁷ entstand im Jahr 2021 im Rahmen des Wissenschaftstransferprojekts im Masterstudiengang Fachdidaktik Medien & Informatik an der Pädagogischen Hochschule Schwyz. Am Projekt beteiligt waren Tobias Rikenmann und

³⁷ <https://www.ml4t.ch/>.

der Autor der vorliegenden Arbeit. Das Ziel bestand darin, eine Webseite zu entwickeln, die Lehrpersonen einen niederschweligen Einstieg in die Themen *KI* und *ML* ermöglichen. Da *KI* für Lehrpersonen ein ebenso neues Thema ist wie für die Lernenden, wurde die Zielgruppe *Lehrpersonen* gewählt. Damit soll verhindert werden, dass Lehrpersonen bei der Vorbereitung der Unterrichtseinheit Fehlkonzepte entwickeln, die sie bei der Durchführung im Unterricht an die Lernenden weitergeben (Lindner & Berges, 2020). In der ersten Phase der Überarbeitung und Entwicklung neuer Lehr- und Lernmaterialien wurden die bestehenden Materialien komplett ausgeblendet, um bei den Experteninterviews den Hauptfokus auf die Neuentwicklungen zu setzen.

2.3 Experteninterviews

Die entwickelten Materialien wurden mit drei Experten der Informatikdidaktik und einem Data Scientist mit Lehrerfahrung besprochen. Das Ziel der Interviews war es, eine professionelle Einschätzung zur Tauglichkeit der Materialien für die Zielstufe in Erfahrung zu bringen (siehe Anhang B Interviewleitfaden). In den ersten beiden Fragen wurden die Experten gebeten einzuschätzen, wie relevant das Thema *ML mit KNN* für Lernende der Sekundarschule ist und wie vertieft die Auseinandersetzung aussehen sollte. Die restlichen Fragen bezogen sich jeweils auf die zum Zeitpunkt des Interviews bestehenden Materialien. Nach jedem Interview wurden die Materialien auf der Webseite angepasst.

3 Ergebnisse

Im vorliegenden Kapitel werden die Ergebnisse aus der Literaturrecherche und den Experteninterviews dargestellt. Anschliessend wird beschrieben, wie die Lehr- und Lernmaterialien aufgrund der gewonnenen Erkenntnisse entstanden und überarbeitet wurden.

3.1 Ergebnisse aus der Literaturrecherche

Die Literatur wird in drei Unterkapitel aufgeteilt. Zuerst werden systematische Übersichtsarbeiten analysiert, um einen Überblick über fachdidaktische Literatur zum Thema *KI im Unterricht vom Kindergarten bis zum zwölften Schuljahr* zu gewinnen. Im Anschluss wird aktuelle Grundlagenliteratur für verschiedene Zielgruppen zum Thema *KI, ML* und *KNN* betrachtet. Abgeschlossen wird das Kapitel mit einer Analyse bestehender Unterrichtseinheiten zu *ML* mit *KNN*. Viele Erkenntnisse aus der Literaturrecherche finden sich bereits im Theorieteil dieser Arbeit. Im Gegensatz zum Kapitel *Theorie*, das sich explizit an Lehrpersonen richtet, die sich mit den Materialien von *ml4t* auseinandergesetzt haben, wird im vorliegenden Kapitel jene Literatur analysiert, die massgeblich zum Entwicklungsprozess der Lehr- und Lernmaterialien beigetragen hat.

3.1.1. Systematische Übersichtsarbeiten

Kandlhofer und Steinbauer fassen in ihrer Arbeit wissenschaftliche Fachzeitschriften, Projekte und Ressourcen im *KI*-Unterricht für Lernende vom Kindergarten bis zum zwölften Schuljahr zusammen. Das Paper bietet einen ersten Überblick und die Autoren erheben keinen Anspruch auf Vollständigkeit, was sie mit der raschen Entwicklung des Feldes begründen (Kandlhofer & Steinbauer, 2021).

Einen Einblick in die internationale Perspektive bieten die drei Interviews von Fredrik Heintz (2021) mit zwei Praktikern und einer politischen Entscheidungsträgerin aus dem Bereich *KI in der Bildung* in den USA, Singapur und Europa. Die drei Perspektiven zeigen auf, dass in

allen drei Regionen im privaten Sektor umfassendes Unterrichtsmaterial produziert wird, das teilweise eine hohe Qualität aufweist, besonders die grossen Tech-Unternehmen aus den USA bieten vielfältige Kurse und Tools für Lernende auf verschiedenen Niveaus an (Heintz, 2021). Einig sind sich die Fachpersonen aus den drei Regionen auch bezüglich der relevantesten offenen Frage im Zusammenhang mit KI in der Bildung:

The main issue is with the number of trained educators who can deliver our AI4K curriculum effectively. (Heintz, 2021, S. 235)

Einen aktuellen systematischen Überblick über visuelle Tools für den KI-Unterricht in der K-12-Bildung bieten Gresse von Wangenheim et al. (2021). In der Arbeit werden 16 Tools hinsichtlich verschiedener Kriterien evaluiert: didaktische Merkmale; Unterstützung der Lernenden bei der Entwicklung von ML-Modellen; Anwendung von ML-Modellen sowie Art und Weise, wie die Werkzeuge entwickelt und evaluiert wurden (Gresse von Wangenheim et al., 2021). Während die Autoren den Einsatz der Tools für Einsteiger grundsätzlich befürworten, kritisieren sie vor allem, dass die Black Box der ML-Konzepte durch den reinen Anwendungscharakter zu wenig aufgebrochen wird (Gresse von Wangenheim et al., 2021).

Einen weitergefassten Ansatz wählten Marques et al. (2020), die eine systematische Bestandsaufnahme mit dem Schwerpunkt auf ML-Unterrichtseinheiten in Grund- und Oberstufenschulen durchführten. Die Forschenden präsentierten die ML-Kompetenzen, die derzeit in den bestehenden Unterrichtseinheiten gelehrt werden, die abgedeckten Anwendungsbereiche, die verwendeten Datentypen und die Arten der verwendeten Lehrmethoden und -materialien. Ausserdem betonten sie, dass ein Kompromiss zwischen der Schaffung transparenter Systeme, die umfassendere Informationen liefern, und der Vereinfachung oder dem *Blackboxing* der zugrundeliegenden ML-Konzepte besteht, um eine kognitive Überlastung zu vermeiden (Marques et al., 2020).

3.1.2. Grundlagenliteratur KI, ML und KNN

Mit der Grundlagenrecherche für die vorliegende Arbeit werden zwei Ziele verfolgt: Einerseits soll das Verständnis des Autors für KI, ML und KNN vertieft und andererseits ein Überblick

über die Erklärungsansätze für verschiedene Zielgruppen gewonnen werden. Tabelle 1 bietet eine Übersicht über die gefundenen Ansätze aus folgenden drei Perspektiven:

1. Welche Zielgruppe soll erreicht werden, wie gross muss das Vorwissen sein, um sich mit dem Text auseinandersetzen zu können und wie stark lässt sich das Wissen damit vertiefen? Die Dimensionen *Vorwissen* und *Tiefe* werden jeweils auf einer Skala von 1 bis 3 mit Sternen dargestellt.
2. Welche Erklärungsansätze (Konzepte, Mathematik, Anwendungsbeispiele zum Nachmachen, gesellschaftliche Perspektive und ethische Überlegungen) bietet der Text und wie breit wird dieser Ansatz abgedeckt?
3. Welches sind die Schwerpunkte des Textes?

Tabelle 1
Grundlagenliteratur

	Referenz	Zielgruppe (Vorwissen, Tiefe)	Erklärungsansatz (K/M/A/G, Breite)	Schwerpunkt
	(Alpaydin, 2021)	Wissenschaft (***)	K (***), M (***), G (***)	ML-Grundlagen
	(Bieri, 2020)	Kinder (*, **)	K (*), G (*)	KI-Errungenschaften
	(Dickmann, 2020)	Kinder (*, *)	G (*)	KI-Errungenschaften
	(Ekman, 2021)	Wissenschaft (**, ***)	K (**), M (**), A (***), G (**)	Aktuelle DL Verfahren mit Anleitungen
	(Fan, 2020)	Allgemeinheit (*, **)	G (**)	KI-Errungenschaften
	(Flessner, 2020)	Kinder (*, **)	K (**), G (*)	KI-Errungenschaften, ML-Grundlagen
	(Gallenbacher, 2021)	Kinder/Allgemeinheit (*, ***)	K (**), M (*), A (*), G (**)	KI-Errungenschaften, ML-Grundlagen, unplugged Übungen
	(Haffner, 2017)	Allgemeinheit (*, ***)	K (**), M (**), A (*)	ML-Grundlagen
	(Kersting et al., 2019)	Allgemeinheit (*, **)	K (***), M (*), G (**)	ML-Grundlagen
	(Konecny, 2020)	Allgemeinheit (*, *)	G (**)	KI-Errungenschaften
	(Krohn et al., 2020)	Wissenschaft (*, **)	K (**), M (*), A (***)	DL-Anleitungen
	(Liukas, 2019)	Kinder ab KG-Stufe (*, **)	K (***), G (**)	ML-Grundlagen, unplugged Übungen
	(Pickover, 2021)	Allgemeinheit (*, *)	G (*)	KI-Errungenschaften
	(Tadeusiewicz, 2020)	Kinder (**, ***)	K (**), G (***)	KI-Errungenschaften, ML-Grundlagen
	(Trask, 2020)	Allgemeinheit (**, **)	K (*), M (*), A (**)	ML-Grundlagen, DL-Anleitungen
	(Zitzler, 2017)	Allgemeinheit (*, **)	K (**), M (*)	ML-Grundlagen
	(Zitzler, 2019)	Allgemeinheit (*, *)	K (**)	ML-Grundlagen

Anmerkung: K = Konzepte, M = Mathematik, A = Anwendungsbeispiele zum Nachmachen, G = gesellschaftliche Perspektive und ethische Überlegungen

Als Zielgruppen wurden drei Kategorien definiert: Kinder, Allgemeinheit und Wissenschaft, wobei sich beispielsweise der Text «Computer auf der Schulbank» gemäss Untertitel des Buches an «alle von 9 bis 99» richtet (Gallenbacher, 2021) und damit zwei Kategorien zugeordnet wurde. Der Text bietet in Bezug auf diese Zielgruppe zahlreiche Erklärungen und Übungen zur Vertiefung (gekennzeichnet durch drei Sterne: ***) mit wenig Vorwissen (*). Während die meisten analysierten Texte wenig Vorwissen voraussetzen, lässt sich ein massgeblicher Unterschied in der gebotenen Tiefe feststellen. Die gesichteten Texte mit dem Schwerpunkt *KI-Errungenschaften* boten meist wenig inhaltliche Tiefe.

3.1.3. Unterrichtseinheiten zu ML mit KNN

Viele Unterrichtseinheiten, die ML mit KNN behandeln, befassen sich mit dem Lernprozess, allerdings auf unterschiedlichem Niveau. Dies zeigen z. B. Marques et al. (2020), die dreissig Unterrichtseinheiten aus dem Bereich *ML* untersuchten:

Most of these Instructional Units present several ML concepts only on an abstract level black-boxing some of the underlying ML processes. In these cases, the model learning process may be approached by only executing a pre-defined model learning process without any need for further interaction ... Very few Instructional Units systematically introduce ML performance measures, such as a correctness table, confidence graph, presenting accuracy often in a more superficial way. (Marques et al., 2020, S. 299)

Zu diesen eher oberflächlichen Ansätzen gehören beispielsweise die Übungen von «Machine Learning for Kids»³⁸ (Lane, 2021). Darin werden zwar Modelle mit eigenen Daten trainiert und die trainierten Modelle können mithilfe von Scratch auf einer blockbasierten Oberfläche implementiert werden, jedoch fehlt bei diesem Ansatz die Veranschaulichung des Lernprozesses. Unterrichtseinheiten, welche die Materialien von *ML for Kids* integrieren, füllen diese Lücke z. B., indem sie ein Erklärvideo einbauen.³⁹

³⁸ <http://www.machinelearningforkids.co.uk/>.

³⁹ <https://www.appsforgeood.org/courses/machine-learning-std/>.

Eine Lücke in den bestehenden Unterrichtsmaterialien konnte ausserdem im Bereich der unplugged Aktivitäten gefunden werden. Long und Magerko (2020) beschreiben in ihren Gestaltungsüberlegungen für die Entwicklung von Lehr- und Lernmaterialien die Relevanz, sich in die Rolle des lernenden Agenten zu versetzen, um dessen Argumentationsprozess nachzuvollziehen:

Consider designing interventions in which individuals can put themselves ‘in the agent’s shoes’ as a way of making sense of the agent’s reasoning process. This may involve embodied simulations of algorithms and/or hands-on physical experimentation with AI technology. (Long & Magerko, 2020, S. 6)

Mit dieser Aufgabe haben sich im deutschsprachigen Raum bereits die Autoren von «AI unplugged» beschäftigt (Seegerer et al., 2019). Die von ihnen beschriebenen Unterrichtseinheiten beinhalten Aktivitäten mit symbolischer Wissensrepräsentation (vgl. 1.3.2) sowie eine Aktivität zu KNN. Letztere zeigt, wie bei der Bilderkennung die Rohdaten mit jeder Schicht im KNN weiter abstrahiert werden. In der Aktivität schlüpfen die Lernenden allerdings nicht in die Rolle des lernenden Agenten, sondern spielen ein bereits trainiertes Netzwerk nach.

Dasselbe gilt für die Lernenden in der unplugged Übung *Brain in a Bag*⁴⁰. Darin versetzen sich vier Teilnehmende in die Rolle der Neuronen eines KNN, das in der Lage ist, eine einfache Variante des Kartenspiels «Snap» zu spielen. Bei diesem Spiel geht es darum, *Snap* zu rufen, wenn zwei Karten der gleichen Farbe aufliegen. Die sensorischen Neuronen der ersten Schicht werden aktiviert, wenn die gezeigte Karte der vorgegebenen Eigenschaft entspricht (z. B. rote Karte in Position 1). Darauf wird eine Rolle auf einem Seil zum Schaltneuron in der versteckten Schicht weitergeleitet. Erhält das blinde Schaltneuron zwei Eingangssignale, leitet dieses eine Rolle zum Ausgangsneuron weiter, welches in diesem Fall *Snap* ruft. Mit der Übung lässt sich zwar erleben, wie Signale im Netz weitergereicht werden, aber es wird nicht klar, woher das

⁴⁰ <https://teachinglondoncomputing.org/resources/inspiring-unplugged-classroom-activities/the-brain-in-a-bag-activity/>.

KNN weiss, weshalb genau diese Konstellation der Signalweiterleitung zum gewünschten Verhalten führt.

3.2 Experteninterviews

Den umfangreichsten Beitrag zur Überarbeitung leistete Michael Hielscher ⁴¹, wissenschaftlicher Mitarbeiter der PH Schwyz und Experte für computergestützte Lehr- und Lernumgebungen. Mit ihm wurde das erste Interview am 28.01.2022 durchgeführt, in welchem fast zwei Stunden darüber diskutiert wurde, wie detailliert sich Lernende der Sekundarschule mit KNN auseinandersetzen sollen. Am Ende konnte die Frage zwar nicht abschliessend beantwortet werden, aber dafür stand fest, wie die weitere Entwicklung aussehen würde. Michael Hielscher schlug vor, sich auf Filmempfehlungen anhand von Filmeigenschaften zu fokussieren und eine entsprechende Webapplikation zu erstellen. Dafür musste sich der Autor der vorliegenden Arbeit jedoch wesentlich vertiefter mit der Funktionsweise von KNN auseinandersetzen. Dies wurde auch bei der Besprechung der ersten Unterrichtsidee ⁴² deutlich, welche unter Fehlvorstellungen der Lernkonzepte eines KNN entwickelt wurde. Aus diesem Grund wurde die Aufgabe komplett gestrichen und durch die neue Aufgabe *Movie Recommender*⁴³ ersetzt.

Nebst Michael Hielscher wurde für die zweite Expertenrunde vom 29.03.2022 Werner Hartmann⁴⁴ eingeladen. Dieser ist Co-Autor des informatikdidaktischen Grundlagenwerks «Informatik planen und durchführen» (Hartmann et al., 2006) und beschäftigte sich unter anderem mit der Frage, welche fundamentalen Ideen im Informatikunterricht behandelt werden sollen. Aus seiner Sicht gehört das Thema *KI* gegenwärtig zur Allgemeinbildung und aufgrund der wirtschaftlichen Entwicklungen sei auch erkennbar, dass es sich nicht bloss um einen Hype handelt. Besonders gefallen hat ihm, dass der Fokus der Übung in Bezug auf das

⁴¹ <https://www.phsz.ch/forschung/medien-und-schule/michael-hielscher/>.

⁴² <https://www.ml4t.ch/unterrichtseinheit/trapp-unplugged> (ausgeblendete Übung).

⁴³ ml4t.ch/uebungen/movie-recommender.

⁴⁴ <http://infosense.ch/hartmann/index.html>.

Dagstuhl-Dreieck (GI, 2016) auf der technologischen Perspektive liegt, während die meisten Ansätze, denen er bisher begegnet ist, Fragen zur gesellschaftlich-kulturellen Perspektive ins Zentrum stellen.

Die einschlägige Erkenntnis aus dieser Expertenrunde war, dass die unplugged Aufgabe mit dem Perzeptron grundsätzlich funktioniert, jedoch beim didaktischen Kommentar darauf aufmerksam gemacht werden sollte, dass theoretisch ein Overfitting stattfindet. Das heisst, das System ist nach der Trainingsphase nicht in der Lage, Daten ausserhalb des Trainingsatzes korrekt zu klassifizieren.

Die weiteren Kommentare der Experten trugen massgeblich dazu bei, dass die Bedienung und Benutzerführung auf der Webseite verbessert wurden, beispielsweise durch eine klarere Hervorhebung der Aufgaben, eine Verlinkung relevanter Fachbegriffe in einem Glossar sowie die Einbindung interaktiver Elemente.

Das dritte Experteninterview wurde am 06.04.2022 mit Eckart Zitzler⁴⁵ geführt. Er ist Vizedirektor und Leiter der Forschung im Departement Informatik an der Hochschule Luzern. Im Buch «Dem Computer ins Hirn geschaut» (Zitzler, 2017) zeigt er mit anschaulichen Darstellungen und bildhafter Sprache auf, wie die Natur als Inspiration für lernende Systeme dient. Die Frage, ob KNN im Unterricht behandelt werden sollen, beantwortet Eckart Zitzler nuanciert:

Maschinelles Lernen muss man thematisieren. Ob das dann schon Richtung Perzeptron sein soll, bin ich mir nicht sicher. Aber auf der Gimy-Stufe auf jeden Fall.

Auf der Sekundarstufe ist es aus seiner Sicht noch nicht relevant nachzuvollziehen, wie der Algorithmus genau funktioniert, sondern das Bewusstsein zu schaffen, dass ein KNN ein dem Gehirn nachempfunderer Algorithmus ist.

Eine weitere Interviewfrage lautete, ob der Bezug zum Alltag der Jugendlichen durch die gewählten Beispiele gewährleistet ist. Dies ist laut Einschätzung von Eckart Zitzler mit dem

⁴⁵ <https://eckartzitzler.ch/docs/about/>.

Thema *Film* gelungen. Ob aber die gewählten Filme den Jugendlichen auch bekannt sind, sei eine andere Frage, die er nicht beantworten könne. Da zwischen der zweiten und dritten Interviewrunde zu wenig Zeit für eine umfassende Überarbeitung der Materialien lag, wurden einige Anregungen zur Benutzerführung von Werner Hartmann eine Woche später auch von Eckart Zitzler angebracht. Die zentrale Veränderung war die Ergänzung eines interaktiven Perzeptrons beim Einstieg in die Übung *Movie Recommender*.

Das letzte Experteninterview fand am 18.05.2022 mit Marc Bravin statt. Da die ersten drei Befragungen mit Personen aus der Fachdidaktik durchgeführt wurden, lag der Fokus des vierten Interviews auf der fachwissenschaftlichen Perspektive. Auch Marc Bravin schätzt das Thema *KNN* als relevant ein, da er aus Forschungssicht bestätigen kann, dass sich hinter den Errungenschaften, die in den Medien unter dem Titel *KI* erscheinen, jeweils *KNN* verstecken. Aus seiner Sicht ist der Einstieg in neuronale Netze über das einschichtige Perzeptron sinnvoll gewählt, da das Mastermodul an der HSLU zu DL ebenfalls mit diesem Beispiel startet. Auch gefällt ihm die Idee, die Gewichte des Perzeptrons durch Schieberegler darzustellen und er findet es sinnvoll, dass die Anpassung der Gewichte nicht willkürlich geschieht, sondern nach einer vorgegebenen Regel. Problematisch ist aus Sicht von Marc Bravin, dass das Perzeptron auf die Trainingsdaten übertrainiert und nicht mit einem Holdout-Set getestet wird. Lernende sollten zwischen der Lern- und der Anwendungsphase eines *KNN* unterscheiden können.

Den Lernprozess kannst du so zwar gut visualisieren, aber schlussendlich lernst du sonst etwas, das nichts bringt.

Die letzte Kritik, die Marc Bravin anbrachte, betrifft die Darstellung der Begriffe von *KI*, *ML*, *KNN* und *DL* (Abbildung 1). Aus seiner Sicht ist *KI* der Überbegriff, der die anderen umschliesst. Eine Erklärung für die alternative Darstellung, die im Rahmen dieser Masterarbeit gewählt wurde, findet sich in Kapitel 1.1.

3.3 Entwicklung und Überarbeitung von Lehr- und Lernmaterialien

Wie im Kapitel *Methodik* aufgezeigt, verlief der Entwicklungsprozess der Lehr- und Lernmaterialien iterativ. Dieses Kapitel teilt diesen Prozess in drei Phasen mit jeweils einer Eigenentwicklung im Zentrum. Da zwischen und während den Entwicklungen ein Austausch mit den oben erwähnten Experten stattfand, enthalten die folgenden Unterkapitel sowohl Beschreibungen und Gestaltungsüberlegungen als auch Expertenfeedback zu den entwickelten Materialien.

3.3.1. TRApp und TRApp unplugged

Die erste entwickelte Unterrichtseinheit trug den Namen *Transparent Rating Application* (kurz TRApp)⁴⁶. In der Übung wird eine *Userin* oder ein *User* bestimmt, der oder dem Bewertungen von Items aus einer Sammlung (z. B. Musikstücke) vorgeschlagen werden. Der Rest der Klasse schlüpft in die Rolle der *Kritiker*innen*, welche für die vorgeschlagene Bewertung zuständig sind. Jede Kritikerin und jeder Kritiker erhält ein Gewicht, das bestimmt, wie stark ihre Bewertung zur Gesamtbewertung beiträgt. Nach jedem Item werden die Vorschläge der Kritiker*innen mit der tatsächlichen Bewertung der Userin bzw. des Users verglichen und entsprechend der Übereinstimmung angepasst. So tragen die Bewertungen der signifikant abweichenden Kritiker*innen nach einigen Runden weniger zur Gesamtbewertung bei und die vorgeschlagene Bewertung verbessert sich. Mit dieser Übung sollen Lernende erfahren, wie ein KNN seine Gewichte im Laufe des Trainings anpasst, um bessere Voraussagen zu treffen.

Da das analoge Anpassen der Gewichte geführt werden muss, damit sich bei der Berechnung keine Fehler einschleichen, wurde nebst der unplugged Übung auch eine Web-Variante mit dem Code.org App Lab⁴⁷ programmiert. Darin konnten Lernende eine eigene Liste mit Items erstellen, die nach dem gleichen Prinzip durch User*innen und Kritiker*innen bewertet

⁴⁶ <https://www.ml4t.ch/unterrichtseinheit/trapp-unplugged> (ausgeblendete Übung).

⁴⁷ <https://code.org/educate/applab>.

werden und somit Empfehlungen generieren. Die Applikation ⁴⁸ wurde mit Familienmitgliedern und Freunden des Autors getestet, bevor sie im Oktober 2021 mit einer dritten Oberstufe aus der Gemeinde Ruswil im Wahlfach MINT erprobt wurde (Abbildung 11).

Abbildung 11

Folie zur Erklärung von TRApp mit Screenshots



Gruppe 1



<https://bit.ly/3iQ35BZ>

- 1) Diskutiert die Stärken und Schwächen des Algorithmus
- 2) Wie könnten die Empfehlungen noch verbessert werden?



Gruppe 2



<https://bit.ly/30cwqQK>

Im ersten Experteninterview stellte Michael Hielscher die Frage, ob der Algorithmus funktioniert, da dies aus der Beschreibung und dem Applikationscode nicht erkennbar sei. Weiter wollte er wissen, ob es sich bei diesem Algorithmus wirklich um ein KNN bzw. ein Perzeptron handelte. Um die erste Frage zu beantworten, wurde ein kurzes Programm in Python geschrieben, mit der die Übung in einer Klasse mit 21 Lernenden simuliert werden sollte.⁴⁹ Darin werden 25 Items randomisiert auf einer Skala von 1 bis 5 beurteilt, gefolgt von

⁴⁸ Projektseite der plugged-Variante von TRApp:

https://studio.code.org/projects/applab/TVQEKNBoNBTL8QmWn2mC_k2hbut6M9-aUHMbx-Q_s.

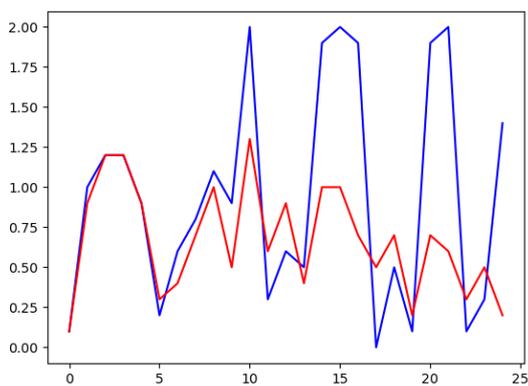
⁴⁹ Programmcode der Klassensimulation TRApp:

<https://github.com/ml4teachers/mlp/blob/585f157fb82f6c25f98f077262dd1cac9f20633a/Perceptron.py>.

zwei erstellten Gruppen, um die Kritiker*innen zu simulieren. Gruppe A bewertet die Items ähnlich (Standardabweichung 1) und Gruppe B immer genau umgekehrt, jedoch ebenfalls mit einer Standardabweichung von 1. Da das Programm die Gruppenzuordnung nicht kennt, gewichtet es zu Beginn alle Bewertungen gleich. Nach jedem Item wird die Gewichtung zu den verschiedenen Bewertungen angepasst. Ähnliche Ergebnisse werden stärker gewichtet, abweichende schwächer. Abbildung 12 zeigt, wie sich die Abweichung der Empfehlung bei anpassenden Gewichten (rot) im Vergleich zu den gleichbleibenden Gewichten (blau) verbessert.

Abbildung 12

Simulation der Abweichung im TRApp-Algorithmus



Noch nicht beantwortet war die Frage, ob sich der in den beiden Übungen verwendete Algorithmus tatsächlich auf die Funktionsweise von KNN zurückführen lässt. Hierfür war eine weitere Auseinandersetzung mit der Fachliteratur nötig, was die zweite Entwicklungsphase einleitete.

3.3.2. Mehrlagiges Perzeptron für Filmempfehlungen

Auf Anregung von Michael Hielscher wurde in der zweiten Phase probiert, ein System zu entwickeln, das Filmempfehlungen aufgrund von bestehenden Bewertungen mithilfe eines KNN erstellt. Dieses soll eine einfache Nutzeroberfläche und die Möglichkeit bieten, in die

Entscheidungsfindung des Algorithmus zu blicken, der wiederum so aufbereitet sein sollte, dass er für Lernende der Sekundarschule leicht nachvollziehbar ist.

Mit diesem Ziel wurde ein Prototyp⁵⁰ auf der Plattform Google Colaboratory⁵¹ erarbeitet (siehe Abbildung 13). Das Programm greift auf bereinigte Filmdaten⁵² der viertausend beliebtesten Filme von The Movie DataBase zu. Diese können auf der Plattform mithilfe des Pandas Dataframes⁵³ in Tabellenform angezeigt und begutachtet werden. Anschliessend werden die Filmdaten mit dem One-Hot-Encoding-Verfahren in Dummy-Variablen umgerechnet. So erhält das mehrlagige Perzeptron⁵⁴ (MLP) nur Nullen und Einsen, um den Film zu klassifizieren.

⁵⁰ Movie Recommender Prototyp auf Google Colab: <https://colab.research.google.com/drive/1jNkTGieZf21PZOZBQFT6ESKVOnyuEJaP?usp=sharing>.

⁵¹ Google Colaboratory (<https://colab.research.google.com/>) ist eine Kollaborationsplattform, die Jupyter-Notebooks auf Google Drive speichert und auf einer virtuellen Maschine ausführen lässt. Dies hat den Vorteil, dass keinerlei Installationen auf dem eigenen Rechner nötig sind und mit minimalen Entwicklerkenntnissen gleichzeitig an Dateien gearbeitet werden kann.

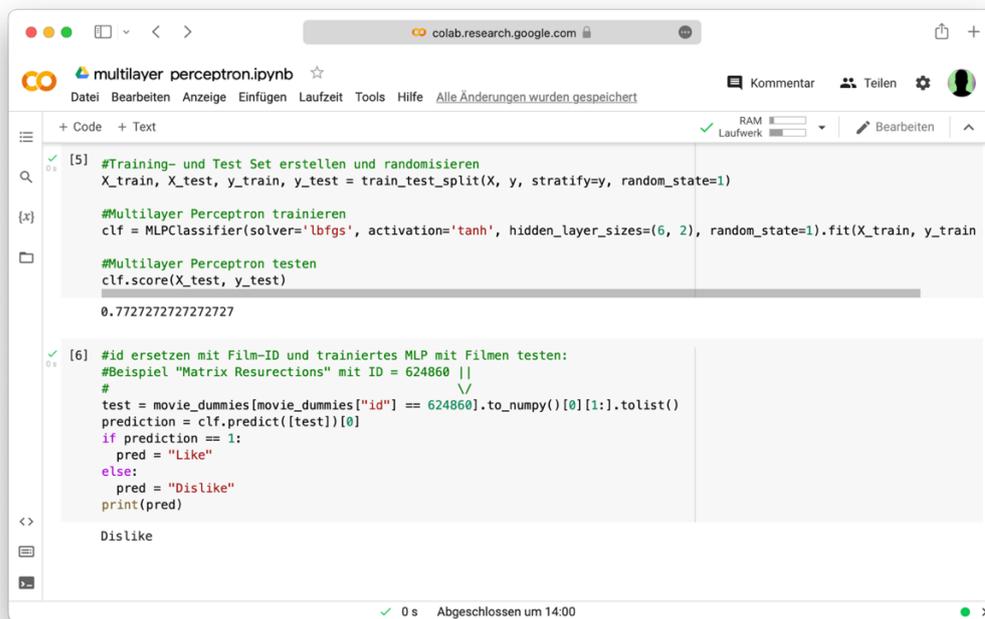
⁵² Die Filmdaten wurden am 11.02.2022 über die Movie DB API (<https://www.themoviedb.org/documentation/api>) heruntergeladen und so aufbereitet, dass nur die eindeutige Film-ID, der Titel, das Erscheinungsjahr, die Beliebtheit, die durchschnittliche Bewertung sowie die Anzahl der Bewertungen und Genres gespeichert wurden. Filme für Erwachsene (Genre: *Adult*) sowie Filme ohne Genrebeschreibung oder Bewertungen wurden nicht aufgenommen.

⁵³ Pandas (<https://pandas.pydata.org/>) ist eine Programmbibliothek für die Darstellung grosser Datenmengen in Python.

⁵⁴ Ein MLP ist ein einfaches KNN mit mindestens einer versteckten Schicht.

Abbildung 13

Prototyp eines Movie-Recommenders auf Basis eines KNN



```
[5] #Training- und Test Set erstellen und randomisieren
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify=y, random_state=1)

#Multilayer Perceptron trainieren
clf = MLPClassifier(solver='lbfgs', activation='tanh', hidden_layer_sizes=(6, 2), random_state=1).fit(X_train, y_train)

#Multilayer Perceptron testen
clf.score(X_test, y_test)

0.7727272727272727

[6] #id ersetzen mit Film-ID und trainiertes MLP mit Filmen testen:
#Beispiel "Matrix Resurrections" mit ID = 624860 ||
#
test = movie_dummies[movie_dummies["id"] == 624860].to_numpy()[0][1:].tolist()
prediction = clf.predict([test])[0]
if prediction == 1:
    pred = "Like"
else:
    pred = "Dislike"
print(pred)

Dislike
```

Um das MLP zu trainieren, wird eine CSV-Datei mit bewerteten Filmen hochgeladen. Die Erstellung einer solchen Datei funktioniert, indem auf themoviedb.org ein Konto angelegt wird und Filme bewertet werden. Für den Test des Prototyps wurden 110 Filme bewertet und das MLP aus der SciKit-Learn⁵⁵ Bibliothek trainiert. Das trainierte MLP sollte gemäss berechneter Genauigkeit in der Lage sein, zu 77 % korrekt vorherzusagen, ob die Person, welche die 110 Filme bewertet hat, die restlichen 3890 Filme ebenfalls mag.

Wie aus der Beschreibung des Prototyps deutlich wird, ist für die Erstellung eines solchen Programms umfangreiches Vorwissen nötig, das in der beschränkt zur Verfügung stehenden Zeit nicht mit Lernenden der Sekundarschule erarbeitet werden kann. Um aus dem vorliegenden Code eine grafisch aufbereitete Nutzeroberfläche zu entwickeln, mit der Lernende ohne Programmiererfahrung in Python interagieren könnten, fehlte dem Autor dieser Masterarbeit sowohl das nötige Grundlagenwissen als auch die Zeit, sich dieses

⁵⁵ Scikit-learn (<https://scikit-learn.org/>) ist eine Programmibibliothek für die Datenanalyse in Python.

anzueignen. Die Entwicklung eines funktionierenden MLP brachte ihn jedoch in zwei Punkten weiter: Das Verständnis für die Funktionsweise von einfachen KNN wurde gestärkt und der Lernalgorithmus der Unterrichtseinheit TRApp (siehe Kapitel 3.3.1) konnte beurteilt bzw. überarbeitet werden. Unter Berücksichtigung der neuen Kenntnisse wurde deutlich, dass mit TRApp kein KNN trainiert wurde und die Übung daher aus den Unterrichtsmaterialien gestrichen werden sollte. Stattdessen soll in der dritten Entwicklungsphase versucht werden, ein lernendes Perzeptron so zu vereinfachen, dass dieses mit analogen Materialien dargestellt werden kann.

3.3.3. Perzeptron unplugged

Als Einstieg ins Thema KNN wählen viele Lehrbuchschaffende das einschichtige Perzeptron (Ekman, 2021; Rashid, 2017; Steinwendner, 2020; Trask, 2020). Auch Marc Bravin, Doktorand an der Hochschule Luzern im Bereich Data-Science, bestätigte, dass er im Studium als einfachstes KNN zuerst die Funktionsweise des einschichtigen Perzeptrons von Frank Rosenblatt aus dem Jahr 1957 (Loiseau, 2022) kennengelernt hatte. In den für diese Masterarbeit begutachteten Lehrmitteln, die das Thema *KNN* für Lernende der Sekundarstufe aufbereiten, wurde ebenfalls mit einer vereinfachten Version von Rosenblatts Wahrnehmungsmaschine gearbeitet (Janssen, 2020; Lardelli, 2021). Bei der Recherche konnte sogar eine scratchbasierte Lernumgebung für Lernende der Primarschule gefunden werden, welche aufzeigen soll, dass der gleiche Perzeptron-Mechanismus für die Implementierung variierender logischer Funktionen verwendet werden kann, wenn er mit unterschiedlichen Datensätzen trainiert wird (Shamir & Levin, 2021).

Da sich der Algorithmus zum Trainieren eines Perzeptrons leicht vereinfachen lässt (siehe Anhang C), wurde in der vorliegenden Arbeit versucht, diesen mit unplugged⁵⁶ Materialien nachzustellen. Dabei wurde zu Beginn festgelegt, dass die Trainingsdaten austauschbar sein

⁵⁶ Computer Science unplugged ist ein informatikdidaktischer Ansatz, der versucht, fundamentale Konzepte der Informatik so weit herunterzubrechen, dass sie ohne den Einsatz digitaler Geräte enaktiv erlebbar gemacht werden können (Nishida et al., 2009).

sollen und sich die Gewichte durch Überprüfen der berechneten Voraussage des Perzeptrons verändern lassen sollen. Ausserdem war es für den Autor von Bedeutung, ein alltagsnahes Beispiel für die Klassifikationsaufgabe zu finden, auf die das Perzeptron trainiert werden soll. Vor dem ersten Prototyp wurde mit Würfeln und Plättchen als veränderbare Gewichte experimentiert. Dies hätte den Vorteil, dass viele Klassenzimmer bereits ausgerüstet sind und die Vorbereitungsarbeit minimiert werden könnte. Beim Experimentieren mit Logikgattern konnten jedoch keine Parameter für den Lernalgorithmus gefunden werden, damit dieser mit Gewichten zwischen 1 und 6 terminiert.⁵⁷ Der Nachteil von Plättchen ist der zusätzliche Rechenschritt. Zwar könnten je drei negative und drei positive Plättchen jeweils ein Gewicht abbilden. Dies verlangsamt jedoch die Berechnung der Voraussage, da jeweils alle Plättchen zusammengezählt werden müssen. Aufgrund dieser Vorüberlegungen entstand der erste Prototyp mit einem verschiebbaren Zahlenstrahl und Filmkärtchen, die jeweils angaben, ob eine Kategorie aktiv (1) oder inaktiv (0) ist (siehe Anhang D).

Beim ersten Prototyp mussten somit für jedes Filmkärtchen drei Multiplikationen und zwei Additionen (teilweise mit negativen Zahlen) gerechnet werden. Bei einer Erprobung im Bekanntenkreis des Autors dieser Arbeit wurde festgestellt, dass dies den Prozess verlangsamt, dass dieser fehleranfällig ist und dass die Motivation der Versuchspersonen, das Perzeptron zu trainieren, bereits nachlässt, bevor es alle Filme korrekt klassifiziert. Somit wurde ein zweiter Prototyp angefertigt, bei dem die Multiplikation bereits in die Filmkärtchen integriert ist. Anstelle einer Multiplikation mit 1 ist das Fenster ausgeschnitten und das Gewicht wird zur Summe dazugezählt. Bei einer Multiplikation mit 0 wird das Gewicht gar nicht erst angezeigt (siehe Prototyp 2, Anhang D). Eine zweite Erprobung im Bekanntenkreis machte deutlich, dass diese Vereinfachung den Lernprozess des Perzeptrons massgeblich beschleunigte.

⁵⁷ Eine Excel-Datei zum Experimentieren kann hier heruntergeladen werden:
[https://github.com/ml4teachers/perceptron_in_excel/blob/5fbb09191310ff7881902602d856255f4f7732b8/Lernendes Perzeptron.xlsx](https://github.com/ml4teachers/perceptron_in_excel/blob/5fbb09191310ff7881902602d856255f4f7732b8/Lernendes%20Perzeptron.xlsx).

Auf Basis des zweiten Prototyps wurden in einem nächsten Schritt Grafiken für die Webseite und eine Kopiervorlage⁵⁸ erstellt. Diese wurden auch in der zweiten Expertenrunde Michael Hielscher und Werner Hartmann gezeigt. Letzterer gab zu bedenken, dass der Aufwand, die Filmkärtchen mit den Fenstern auszuschneiden und die Gewichtsstreifen einzufahren, Lehrpersonen abschrecken könnte. So wurde für die Erstellung der zweiten Kopiervorlage ein Produktdesigner aus dem Bekanntenkreis des Autors⁵⁹ um Rat gefragt, welcher die Idee hatte, das Perzeptron doppellagig zu machen und mit Heftklammern zu fixieren. Die Filmkarten wurden ebenfalls so umgestaltet, dass diese ohne Cutter hergestellt werden konnten (siehe Anhang E).

⁵⁸ Siehe Kopiervorlage 1: <https://www.ml4t.ch/lehrperson/unterrichtsmaterial>.

⁵⁹ <https://mladendjakovic.ch/>.

4 Diskussion

Im folgenden Kapitel werden die entwickelten Lehr- und Lernmaterialien sowie die Experteneinschätzung zusammengefasst. Darauf folgen die Limitationen der Ergebnisse sowie Vorschläge für weiterführende Forschung.

4.1 Zusammenfassung der entwickelten Lehr- und Lernmaterialien

Im Rahmen der vorliegenden Arbeit wurde eine unplugged Aktivität für Lernende ab der Sekundarstufe 1 entwickelt, die es ihnen erlaubt, aus der Rolle des Agenten die Anpassung der numerischen Wissensrepräsentation nachzuvollziehen. Mithilfe der Lernaktivität sollen Lernende nachvollziehen, wie ein künstliches Neuron die Gewichtung seiner Eingabedaten im Laufe eines Trainings anpasst, um damit bessere Voraussagen treffen zu können. Die Aufgabe wurde in eine webbasierte Unterrichtseinheit eingebettet, die alle fünf *Big Ideas of AI4K12* nach Touretzky et al. (2019) abdeckt und insgesamt in drei Doppellektionen durchführbar ist. Im Folgenden werden die drei Teile anhand ihrer Kernaufgaben beschrieben.

4.1.1. Empfehlungssysteme

Zentrale Erkenntnis der Einstiegsaufgabe⁶⁰ ist, dass ein Recommendersystem Vorschläge aufgrund vorhandener Daten generiert. Damit hat es zwei Schwachstellen: Es braucht sowohl <gute> Daten als auch einen Algorithmus, der aus den Daten möglichst <gute> Empfehlungen generiert. Die Kriterien dieser normativen Beschreibung entdecken die Lernenden anhand eines vortrainierten Empfehlungssystems. Durch einen Blick in den überschaubaren Datensatz sollen die Lernenden entdecken, dass die Empfehlungen des Systems anhand einer unpassenden Klassifizierung vorgenommen wurden. Derselbe Datensatz könnte jedoch auch für eine komplett andere Empfehlung genutzt werden, indem die Klassifikationsmerkmale angepasst werden.

⁶⁰ ml4t.ch/uebungen/song-recommender.

4.1.2. Be-greifbares Perzeptron

Ziel dieser Aufgabe ist es, den Lernalgorithmus von KNN für Lernende ab der Sekundarstufe 1 be-greifbar zu machen. Dabei stellt sich die Frage, wie stark dieser vereinfacht werden kann, damit er von den Lernenden handelnd untersucht werden kann. Die Schwierigkeit liegt im Ausloten der Vereinfachung, damit die Lernfähigkeit des Systems erhalten bleibt und als solche erkannt werden kann. Bei der Literaturrecherche wurden mehrere Unterrichtseinheiten gefunden, die dasselbe Ziel verfolgten, jedoch diese Grenzen weniger ausloteten und sich daher eher an technisch Interessierte richten.⁶¹ In Anhang C wird dargelegt, welche Entscheidungen zur Vereinfachung des in der Übung *Movie Recommender*⁶² verwendeten Modells geführt haben. Der Abschnitt ist so formuliert, dass Lehrpersonen, welche die Unterrichtseinheit behandeln möchten, die mathematischen Überlegungen nachvollziehen können.

4.1.3. Bilder erkennen

In der dritten Übung⁶³ sollen die Lernenden erkennen, dass die Qualität der Voraussagen eines KNN von den Daten abhängt, mit denen es trainiert wurde. Die Übung wurde von Tobias Rikenmann im Rahmen des Moduls *Wissenschaftstransfer 2021* erstellt und war bereits Teil der ersten Version von ml4t. Im Rahmen der vorliegenden Masterarbeit wurde die Übung leicht angepasst und mit einem didaktischen Kommentar hinterlegt.

4.2 Zusammenfassung der Experteneinschätzung

Das Ziel, eine Übung zu erstellen, die es Lernenden ermöglicht, nachzuvollziehen, was in einem KNN passiert, wenn von *ML* gesprochen wird, wurde mithilfe des be-greifbaren Perzeptrons gemäss Experteneinschätzung erreicht. Kritisiert wurde, dass es sich beim Lernprozess streng

⁶¹ <https://www.science-on-stage.de/material/machine-learning-der-schule> (Janssen, 2020); <https://ki-kit.ch/ki-rollenspiel.html> (Lardelli, 2021).

⁶² ml4t.ch/uebungen/movie-recommender.

⁶³ ml4t.ch/uebungen/bilder-erkennen.

genommen nicht um ML handelt, da das Perzeptron zwar durch Anpassung der Gewichte eine Klassifikation eines Trainingsdatensets erlernt, diese jedoch nicht auf Daten ausserhalb der Trainingsdaten angewandt werden kann. Da Training, Test und Anwendung mit den gleichen Daten vorgenommen werden, findet eine Überanpassung (Overfitting) auf die Trainingsdaten statt. Dies wird im didaktischen Kommentar⁶⁴ aufgegriffen und sollte durch die Lehrperson thematisiert werden, damit die Lernenden kein Fehlkonzept von ML entwickeln.

Um den gesamten ML-Prozess vom Training zur Anwendungsphase nachzuvollziehen, wurde als dritte Übung⁶⁵ auf ml4t die Lernaufgabe von Tobias Rikenmann hinzugefügt. Darin trainieren die Lernenden mithilfe von Google Teachable Machine⁶⁶ ein Modell zur Bilderkennung, das sie anschliessend testen und mithilfe der Scratch-Erweiterung von «Machine Learning for Kids»⁶⁷ (Lane, 2021) in ein eigenes Programm einbinden.

Bezogen auf die fünf grossen Ideen der KI ist festzuhalten, dass der Schwerpunkt von ml4t auf ML mit KNN liegt. Es lassen sich jedoch alle Perspektiven mithilfe der Übungen abdecken (vgl. Tabelle 2).

Tabelle 2

Verortung der Big Ideas of AI in den Übungen von ml4t.ch

Big Ideas of AI (Touretzky et al., 2019)	Umsetzung auf ml4t.ch	Beschreibung der Tätigkeit und Erkenntnis aus der Übung
Wahrnehmung	Bilder erkennen mit Google Teachable Machine	Ein Modell selbst trainieren und Einfluss der Trainingsdaten auf Vorhersagequalität erfahren
Repräsentation und Schlussfolgerung	Song-Rec recommender testen und verbessern	Schlussfolgerung aus numerischer Wissensrepräsentation eines Systems durch Anpassung des Datensatzes und/oder Auswertungsmethode beeinflussen
Maschinelles Lernen	Perzeptron als Movie-Rec recommender verstehen und beurteilen	Maschinelles Lernen in einem einschichtigen KNN aus der Rolle des lernenden Systems nachvollziehen und auf Tauglichkeit prüfen
Maschinelles Lernen	Selbsttrainiertes Modell in Scratch einbinden	Ein trainiertes Modell auf neue Daten anwenden
Interaktion	Movie-Rec recommender verbessern (A12)	Klassifizierungsmerkmale eines verbesserten Movie-Rec recommenders definieren und mit entsprechenden Design-Überlegungen eigenen Prototyp erstellen
Gesellschaftliche Auswirkungen	Vertiefungsaufgabe Song-Rec recommender	Überlegungen zu Datenerhebung und -auswertung, um Empfehlungen zu optimieren

⁶⁴ <https://www.ml4t.ch/lehrperson/didaktischer-kommentar>.

⁶⁵ <https://www.ml4t.ch/uebungen/bilder-erkennen>.

⁶⁶ <https://teachablemachine.withgoogle.com>.

⁶⁷ <https://machinelearningforkids.co.uk>.

4.3 Limitationen und weiterführende Forschung

In der Projektskizze und im Exposé für die vorliegende Masterarbeit wurde festgehalten, dass die entwickelten Materialien für eine Weiterbildung mit Lehrpersonen genutzt werden sollen, die im Anschluss ausgewertet wird. Diese wurde jedoch nicht durchgeführt, da die fachliche Vertiefung nach dem ersten Experteninterview mit Michael Hielscher mehr Zeit in Anspruch nahm als angenommen. Die vertiefte Auseinandersetzung führte zur Entwicklung zwei weiterer Lernaufgaben, wovon die erste (siehe Kapitel 3.3.2) mit Lernenden ab Sek2-Stufe und die zweite ab Sek1-Stufe (Perzeptron unplugged) durchgeführt werden kann. Eine Erprobung dieser Aufgaben auf der Zielstufe fand jedoch nicht statt.

Um die Materialien auf ihre Tauglichkeit zu testen, müssten die KI-Konzepte der Lernenden nach der Intervention geprüft werden. Im Idealfall würden die Lehrpersonen dieser Lernenden zuerst die Weiterbildung besuchen und sich mit den Materialien der Webseite ml4t.ch vertraut machen. Nach der Unterrichtseinheit könnten die gesammelten Artefakte, die beim Lösen der Aufgaben entstehen, begutachtet werden. Interviews mit den Lernenden können zeigen, ob die beschriebenen Lernziele für den selbstbestimmten und kritischen Umgang mit KI-Systemen erreicht wurden.

5 Fazit

Die vorliegende Arbeit zeigt am Beispiel *KI*, wie ein aktuelles und sich schnell veränderndes technologisches Thema für Lehrpersonen zugänglich gemacht werden kann. Da *KI* nicht explizit im aktuellen Lehrplan der Schweiz verortbar ist, gibt es bisher wenig nationale Bestrebungen, den Zugang zur Thematik für Lernende zu optimieren. Entsprechend bringen Lehrpersonen kaum Vorwissen aus ihrer Ausbildung mit, falls sie das Thema im Unterricht behandeln möchten, um den selbstbestimmten und kritischen Umgang mit *KI*-Systemen ihrer Schülerinnen und Schüler zu fördern. Die im Rahmen der vorliegenden Arbeit entwickelten Unterrichtsmaterialien sollen Lehrpersonen den Einstieg in dieses umfassende Thema erleichtern. Mithilfe haptischer Lernerfahrungen und offener Aufgaben zur kritischen Auseinandersetzung sollen Lernende ab Sekundarstufe 1 dazu angeregt werden, hinter die Funktionsweise von *KNN* zu blicken.

Die Lernmaterialien wurden von Experten in den Bereichen *Informatikdidaktik*, *computergestützte Lernumgebungen* und *Data-Science* begutachtet und im iterativen Prozess überarbeitet. Aufgrund der Expertengespräche und Rückmeldungen wurde klar, dass eine vertiefte fachliche Auseinandersetzung mit *KNN* unumgänglich war, um das Konzeptwissen zu destillieren. Den Aufwand für diese Vertiefung gilt es nicht zu unterschätzen. Erst durch den Vergleich verschiedener Grundlagenliteratur war es dem Autor möglich, die Funktionsweise lernfähiger *KNN* so weit zu vereinfachen, dass diese für Lernende ab Sekundarstufe 1 adaptiert werden konnte. Ausserdem war diese Auseinandersetzung eine unumgängliche Voraussetzung für die Expertengespräche, da bei der Evaluation der Materialien jede Design-Entscheidung begründbar sein soll. Diese Erkenntnisse können für Lehrende von Bedeutung sein, die sich mit dem Thema *KI* oder weiteren technologischen Neuentwicklungen auseinandersetzen und eigene Lernumgebungen entwickeln möchten.

Da die fachliche Vertiefung bedeutend mehr Zeit als erwartet in Anspruch nahm, wurden die Materialien nur mit Einzelpersonen aus dem näheren Umfeld des Autors erprobt. Die Überprüfung der Praxistauglichkeit und Lernwirksamkeit auf Ebene der Lehrpersonen und

mit Lernenden ab Sekundarstufe 1 wäre angezeigt. Eine Weiterbildung mit Lehrpersonen für den Einsatz der Materialien wurde im Programm der Pädagogischen Hochschule Zug ausgeschrieben und soll im Herbst 2022 zum ersten Mal stattfinden.

Offen bleibt die Frage, ob das Thema *KI* den offiziellen Einzug in die Schweizer Klassenzimmer schafft. Im Hinblick auf internationale Vorstösse und die umfangreiche Menge an Neuentwicklungen im englischsprachigen Raum lässt sich zwar ein klarer Trend ausmachen; Solange aber in einigen Kantonen noch darüber gestritten wird, ob das Modul *Medien und Informatik* ab der fünften Klasse eine eigene Wochenlektion verdient, ist nicht davon auszugehen, dass die Schweizer Bildungslandschaft dazu bereit ist, Raum für KI-Kompetenzen im Lehrplan zu schaffen. Für den Autor und die befragten Experten steht jedoch fest, dass KI ab Sekundarstufe 1 thematisiert werden soll. Entsprechende Lehr- und Lernmaterialien sind inzwischen vielzählig vorhanden und werden durch Beiträge wie die vorliegende Arbeit fortlaufend ergänzt.

6 Literaturverzeichnis

- AI4K12. (2020). *Big Ideas Poster*. AI4K12. <https://ai4k12.org/resources/big-ideas-poster/>
- Alpaydm, E. (2021). *Machine learning* (Revised and updated edition). The MIT Press.
- Bieri, A. (2020). *Globi und die Roboter. Über Datenströme, künstliche Intelligenz und Maschinen* (1. Auflage). Globi Verlag.
- Chatterjee, M. (2022, 2. Januar). *Top 20 Applications of Deep Learning in 2022 Across Industries*. GreatLearning Blog: Free Resources What Matters to Shape Your Career! <https://www.mygreatlearning.com/blog/deep-learning-applications/>
- Dastin, J. (2018, 10. Oktober). *Amazon scraps secret AI recruiting tool that showed bias against women*. Reuters. <https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G>
- Daum, T. (2019). *Die künstliche Intelligenz des Kapitals*. Nautilus.
- Dickmann, N. (2020). *Roboter und künstliche Intelligenz Computertechnik verstehen (CORONA Sachbücher)* (1. Auflage). Ars Scribendi Verlag.
- Druga, S. (2018). *Growing up with AI. Cognimates: from coding to teaching machines* (Dissertation, Media Arts and Sciences, School of Architecture and Planning). Massachusetts Institute of Technology. https://dam-prod2.media.mit.edu/x/2018/08/29/Growin_up_with_AI_Stefania_Druga_2018.pdf
- Ekman, M. (2021). *Learning deep learning theory and practice of neural networks, computer vision, natural language processing, and transformers using TensorFlow*. Addison-Wesley.
- Elkins, K., & Chun, J. (2020). Can GPT-3 Pass a Writer's Turing Test? *Journal of Cultural Analytics*, 5(2). <https://doi.org/10.22148/001c.17212>
- Fan, S. (2020). *Macht Künstliche Intelligenz uns überflüssig? Große Fragen des 21. Jahrhunderts* (M. Taylor, Hrsg.; M. Hesse-Hujber, Übers.). DK.
- Flessner, B. (2020). *Künstliche Intelligenz*. Tessloff.
- Gallenbacher, J. (2021). *Abenteuer Informatik: IT zum Anfassen für alle von 9 bis 99 – vom Navi bis Social Media*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-63739-5>

- Gardner, M. (1991). *The unexpected hanging, and other mathematical diversions: With a new afterword and expanded bibliography* (University of Chicago Press ed). University of Chicago Press.
- GI. (2016). *Dagstuhl-Erklärung. Bildung in der digitalen vernetzten Welt*. Gesellschaft für Informatik. <https://www.gi.de/aktuelles/meldungen/detailansicht/article/dagstuhl-erklaerung-bildung-in-der-digitalen-vernetzten-welt.html>
- Gresse von Wangenheim, C., Hauck, J. C. R., Pacheco, F. S., & Bertoneceli Bueno, M. F. (2021). Visual tools for teaching machine learning in K-12: A ten-year systematic mapping. *Education and Information Technologies*, 26(5), 5733–5778. <https://doi.org/10.1007/s10639-021-10570-8>
- Haffner, E.-G. (2017). *Informatik: Das Lehrbuch für Dummies* (1. Auflage). Wiley-VCH Verlag GmbH & Co. KGaA.
- Hartmann, W., Näf, M., & Reichert, R. (2006). *Informatikunterricht planen und durchführen*. Springer.
- Heintz, F. (2021). Three Interviews About K-12 AI Education in America, Europe, and Singapore. *KI - Künstliche Intelligenz*, 35(2), 233–237. <https://doi.org/10.1007/s13218-021-00730-w>
- Holzinger, A. (2018). Explainable AI (ex-AI). *Informatik-Spektrum*, 41(2), 138–143. <https://doi.org/10.1007/s00287-018-1102-5>
- Janssen, D. (2020). *Machine Learning in der Schule*. Science on Stage Deutschland e.V.
- Johnson, S., & Iziev, N. (2022). *A.I. Is Mastering Language. Should We Trust What It Says?* <https://www.nytimes.com/2022/04/15/magazine/ai-language.html>
- Kahn, K., Lu, Y., Zhang, J., Winters, N., & Gao, M. (2020). *Deep learning programming by all*. Construcitonism 2020.
- Kahn, K., Prasad, R., Veera, G. (2022). AI Snap! Blocks for speech input and output, computer vision, word embeddings, and neural net creation, training, and use. *The Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI-22)*, 12861

- Kahn, K., & Winters, N. (2021). Constructionism and AI: A history and possible futures. *British Journal of Educational Technology*, 52(3), 1130–1142. <https://doi.org/10.1111/bjet.13088>
- Kandlhofer, M., & Steinbauer, G. (2021). AI K–12 Education Service. *KI - Künstliche Intelligenz*, 35(2), 125–126. <https://doi.org/10.1007/s13218-021-00715-9>
- Kersting, K., Lampert, C., & Rothkopf, C. (Hrsg.). (2019). *Wie Maschinen lernen: Künstliche Intelligenz verständlich erklärt*. Springer Fachmedien. <https://doi.org/10.1007/978-3-658-26763-6>
- Kim, S., Jang, Y., Kim, W., Choi, S., Jung, H., Kim, S., & Kim, H. (2021). Why and What to Teach: AI Curriculum for Elementary School. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(17), 15569–15576. <https://ojs.aaai.org/index.php/AAAI/article/view/17833>
- Konecny, J. (2020). *Ist das intelligent oder kann das weg?* LMV.
- Krohn, J., Beyleveld, G., & Bassens, A. (2020). *Deep Learning illustriert: Eine anschauliche Einführung in Machine Vision, Natural Language Processing und Bilderzeugung für Programmierer und Datenanalysten*. dpunkt.verlag.
- Kurzweil, R. (2005). *The Singularity Is Near: When Humans Transcend Biology*. Viking.
- Lane, D. (2021). *Machine Learning for Kids: A Project-Based Introduction to Artificial Intelligence*. No Starch Press. <https://books.google.ch/books?id=g3ISEAAAQBAJ>
- Lardelli, M. (2021). *KI und Robotik für Jugendliche*. www.ki-kit.ch
- Lemoine, B. (2022, Juni 11). *Is LaMDA Sentient? — An Interview*. Medium. <https://cajundiscordian.medium.com/is-lambda-sentient-an-interview-ea64d916d917>
- Li, S., & Deng, W. (2020). Deep Facial Expression Recognition: A Survey. *IEEE Transactions on Affective Computing*, 1–1. <https://doi.org/10.1109/TAFFC.2020.2981446>
- Lindner, A., & Berges, M. (2020). Can you explain AI to me? Teachers' pre-concepts about Artificial Intelligence. In *2020 IEEE Frontiers in Education Conference (FIE)*. Uppsala, Se. <https://cris.fau.de/bibtex/publication/244765205.bib>
- Liukas, L. (2019). Hello Ruby. Wenn Roboter zur Schule gehen. In *Hello Ruby Wenn Roboter zur Schule gehen* (1. Auflage). Bananenblau.

- Loiseau, J.-C. B. (2022, 10. Januar). *Rosenblatt's perceptron, the very first neural network*. Medium. <https://towardsdatascience.com/rosenblatts-perceptron-the-very-first-neural-network-37a3ec09038a>
- Long, D., & Magerko, B. (2020, April 21). What is AI Literacy? Competencies and Design Considerations. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu HI USA. <https://doi.org/10.1145/3313831.3376727>
- Marcus, G. (2022, März 10). *Deep Learning Is Hitting a Wall*. Nautilus | Science Connected. <https://nautil.us/deep-learning-is-hitting-a-wall-14467/>
- Marques, L. S., Gresse Von Wangenheim, C., & Hauck, J. C. R. (2020). Teaching Machine Learning in School: A Systematic Mapping of the State of the Art. *Informatics in Education*, 283–321. <https://doi.org/10.15388/infedu.2020.14>
- McCarthy, J., Minsky, M. L., Rochester, N., & Shannon, C. E. (2006). A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence, August 31, 1955. *AI Mag.*, 27, 12–13.
- Michaeli, T., Seegerer, S., & Jatzlau, S. (2020). Kein Hexenwerk - Ideen des maschinellen Lernens in Snap! *LOG IN - Informatische Bildung und Computer in der Schule*, 76–80.
- Narang, S., & Chowdhery, A. (2022). Pathways Language Model (PaLM): Scaling to 540 Billion Parameters for Breakthrough Performance. *Google AI Blog*. <http://ai.google-blog.com/2022/04/pathways-language-model-palm-scaling-to.html>
- Newell, A. (1982). *Intellectual Issues in the History of Artificial Intelligence*: Defense Technical Information Center. <https://doi.org/10.21236/ADA125318>
- Nishida, T., Kanemune, S., & Idosaka, Y. (2009). A CS Unplugged Design Pattern. *ACM SIGCSE Bulletin* 41(1), 231–235. <http://dx.doi.org/10.1145/1539024.1508951>
- Nüesch, D. (2016, März 10). *Gesellschaft & Religion - Humane Blamage beim Brettspiel? Was der Sieg von AlphaGo bedeutet*. Schweizer Radio und Fernsehen (SRF). <https://www.srf.ch/kultur/gesellschaft-religion/humane-blamage-beim-brettspiel-was-der-sieg-von-alphago-bedeutet>

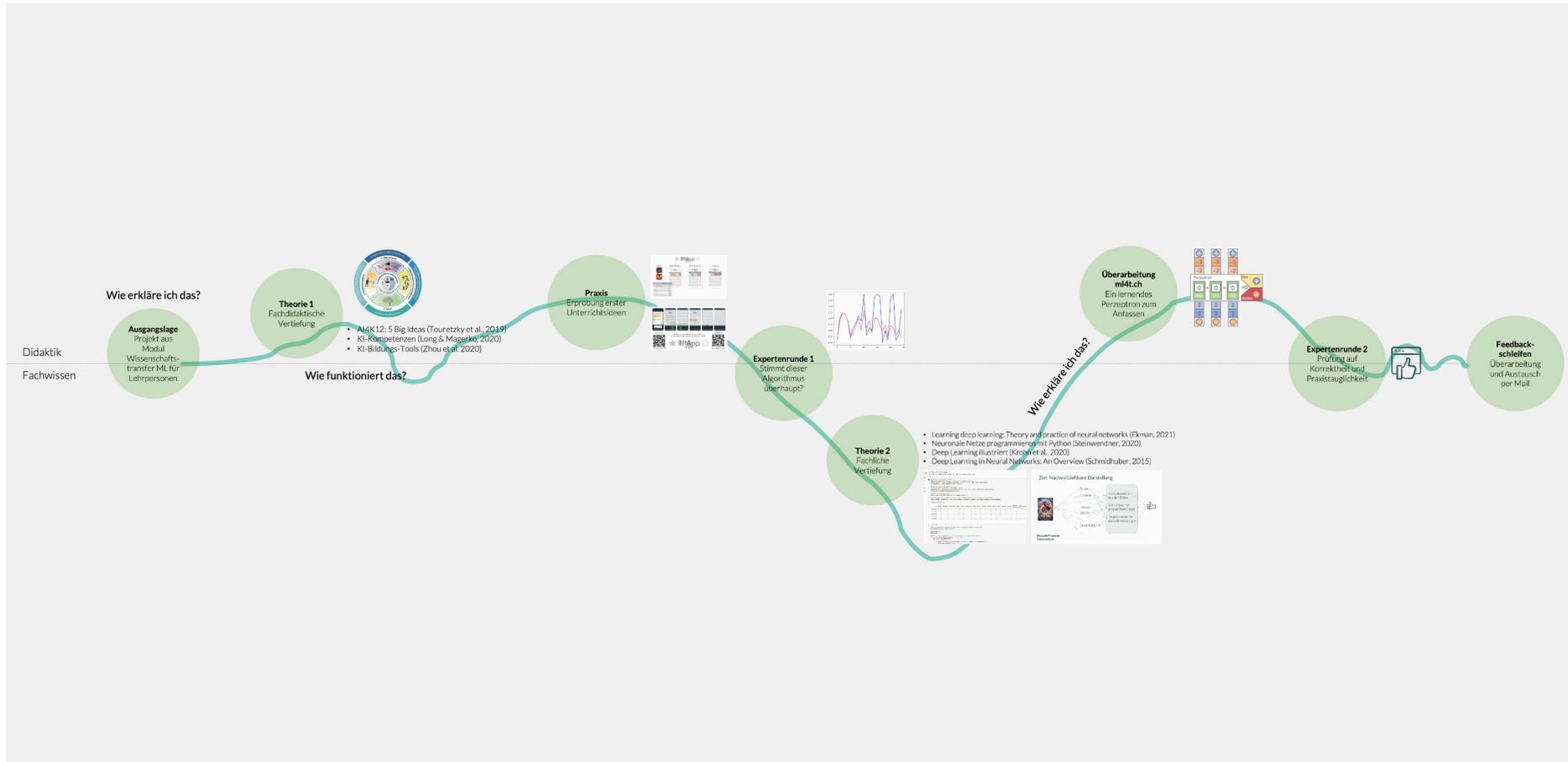
- Ormond, J. (2019). *Fathers of the Deep Learning Revolution Receive ACM A.M. Turing Award*.
<https://www.acm.org/media-center/2019/march/turing-award-2018>
- Ouchchy, L., Coin, A., & Dubljević, V. (2020). AI in the headlines: The portrayal of the ethical issues of artificial intelligence in the media. *AI & SOCIETY*, 35(4), 927–936.
<https://doi.org/10.1007/s00146-020-00965-5>
- Pandolfini, B. (1997). *Kasparov and Deep Blue: The Historic Chess Match Between Man and Machine*. Touchstone. <https://books.google.ch/books?id=j2wWi5Cc6JEC>
- Payne, B. H., & Breazeal, S. C. (2019). *An Ethics of Artificial Intelligence Curriculum for Middle School Students*. <https://www.media.mit.edu/projects/ai-ethics-for-middle-school/overview/>
- Pickover, C. A. (2021). *Künstliche Intelligenz. Eine illustrierte Geschichte von mittelalterlichen Robotern zu neuronalen Netzen*. Librero.
- Plummer, L. (2017, 22. August). This is how Netflix’s top-secret recommendation system works. *Wired UK*. <https://www.wired.co.uk/article/how-do-netflixs-algorithms-work-machine-learning-helps-to-predict-what-viewers-will-like>
- PseudoIntellectual. (2008, 7. Oktober). *Perceptron Research from the 50’s & 60’s, clip*.
https://www.youtube.com/watch?v=cNxadbrN_al
- Rashid, T. (2017). *Neuronale Netze selbst programmieren: Ein verständlicher Einstieg mit Python* (F. Langenau, Übers.; 1. Auflage). O’Reilly.
- SBFI. (2019). *L’intelligence artificielle dans la formation*. Staatssekretariat für Bildung, Forschung und Innovation. https://www.sbf.admin.ch/dam/sbf/de/dokumente/2019/12/k-i_bildung.pdf.download.pdf/k-i_bildung_f.pdf
- Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks*, 61, 85–117. <https://doi.org/10.1016/j.neunet.2014.09.003>
- Schwill, A. (1993). Fundamentale Ideen der Informatik. *Zentralblatt für Didaktik der Mathematik*.
- Seegerer, S., Lindner, A., & Romeike, R. (2019). AI Unplugged – Wir ziehen Künstlicher Intelligenz den Stecker. In A. Pasternak (Hrsg.), *Informatik für alle* (S. 10). Gesellschaft der Informatik.
- Shamir, G., & Levin, I. (2021). Neural Network Construction Practices in Elementary School. *KI -*

Künstliche Intelligenz, 35(2), 181–189. <https://doi.org/10.1007/s13218-021-00729-3>

- Steinwendner, J. (2020). *Neuronale Netze programmieren mit Python* (2., aktualisierte und überarbeitete Auflage). Rheinwerk Verlag.
- Tadeusiewicz, R. (2020). *Ava im Land der Zukunft oder Wie künstliche Intelligenz funktioniert* (Erstausgabe). Helvetiq.
- Touretzky, D., Gardner-McCune, C., Martin, F., & Seehorn, D. (2019). Envisioning AI for K-12: What Should Every Child Know about AI? *Proceedings of the AAAI Conference on Artificial Intelligence*, 33, 9795–9799. <https://doi.org/10.1609/aaai.v33i01.33019795>
- Trask, A. W. (2020). *Neuronale Netze und Deep Learning kapieren: Der einfache Praxiseinstieg mit Beispielen in Python* (K. Lorenzen, Übers.; 1. Auflage 2020). mitp.
- Wirth, M. (1993). *Ideen für den Informatikunterricht. Eine Sammlung von Unterrichtseinheiten für den Informatikunterricht an der Volksschuloberstufe des Kantons Solothurn*. Kantonaler Lehrmittelverlag Solothurn.
- Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering - EASE '14*. the 18th International Conference, London, England, United Kingdom. <https://doi.org/10.1145/2601248.2601268>
- Zhou, X., Van Brummelen, J., & Lin, P. (2020). Designing AI Learning Experiences for K-12: Emerging Works, Future Opportunities and a Design Framework. *ArXiv:2009.10228 [Cs]*. <http://arxiv.org/abs/2009.10228>
- Zhu, K. (2019). *An Educational Approach to Machine Learning with Mobile Applications* (Thesis, Department of Electrical Engineering and Computer Science). Massachusetts Institute of Technology. <https://hdl.handle.net/1721.1/122989>
- Zitzler, E. (2017). *Dem Computer ins Hirn geschaut*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-53666-7>
- Zitzler, E. (2019). *Basiswissen Informatik—Grundideen einfach und anschaulich erklärt*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-59281-6>

Anhang

A Forschungs- und Entwicklungsprozess der vorliegenden Masterarbeit



https://miro.com/app/board/uXjVO4xTslw=?share_link_id=278208287938

B Interviewleitfaden

Einstiegsfragen

Das Forschungsinteresse meiner Masterarbeit liegt darin, die technologische Perspektive des maschinellen Lernens für Lernende der Sekundarschule (be-)greifbar zu machen. Insbesondere sollen die Lernenden verstehen, wie ein künstliches Neuron die Gewichtung seiner Eingabedaten im Laufe eines Trainings anpasst, um damit bessere Voraussagen treffen zu können.

1. Weshalb ist das Thema «maschinelles Lernen mit künstlichen neuronalen Netzen» relevant für Lernende der Sekundarschule?
2. Wie stark soll man im Unterricht mit Lernenden der Sekundarschule in die Tiefe gehen? Folgende Darstellung kann bei der Einordnung helfen.

Keine Erwähnung von künstlichen neuronalen Netzen	Ein künstliches Neuron mit veränderbaren Gewichten erklären können	Ein tiefes neuronales Netz mit mindestens einer versteckten Schicht erklären können	Topologien unterscheiden und Aktivierungsfunktionen erklären können	...
0	1	2	3	4

Fragen zur Unterrichtssequenz (ml4t.ch)

Die Zeit, die im Unterricht für das Thema «künstliche Intelligenz» zur Verfügung steht, ist knapp. Im Kanton Luzern begegnet man dem Begriff KI im Lehrplan des neuen Wahlfachs MINT (Mathematik, Informatik, Naturwissenschaften, Technik) im Kompetenzbereich Maschinenbau und Elektrotechnik. Dieser ist jedoch derart voll, dass es kaum möglich ist, alle Kompetenzstufen zu berücksichtigen. Wenn sich somit eine Lehrperson dafür entscheidet, das Thema aufzugreifen, stellt sich die Frage, welche grundlegenden Konzepte der KI sie in der beschränkten Zeit vermitteln soll.

Im Zusammenhang mit KI im Unterricht stösst man in der fachdidaktischen Literatur oft auf die Formulierung, die «Black Box» des maschinellen Lernens aufzubrechen. Dies ist jedoch aus meiner Sicht nur dann möglich, wenn die lernenden Algorithmen so stark vereinfacht werden, dass sie mit den tiefen künstlichen neuronalen Netzen, die wir heute meinen, wenn wir von KI sprechen, wenig zu tun haben. Bei den bisher von mir rezipierten Lehrmaterialien fand ich einige solche Ansätze, jedoch fehlt mir bei diesen jeweils entweder die mathematische Erklärung oder der Bezug zum Alltag der Jugendlichen. Mit meiner Unterrichtsidee möchte ich beiden Ansprüchen gerecht werden.

Fragen zur Lernumgebung «TRApp unplugged» / «Movie Recommender»

1. Wie tauglich ist die Lernumgebung für die Zielgruppe?
2. Wie könnte die Lernumgebung verbessert werden, um den beiden formulierten Ansprüchen gerecht zu werden:
 - a. den Bezug zum Alltag der Jugendlichen besser herzustellen
 - b. das maschinelle Lernen mathematisch nachvollziehbar zu machen
3. Sonstige Bemerkungen

C Überlegungen zur Vereinfachung des lernenden Perzeptrons

Das Ziel der Übung «Movie Recommender» ist es, den Lernalgorithmus von künstlichen neuronalen Netzen für Lernende ab der Sekundarstufe 1 begreifbar zu machen. Dabei stellt sich die Frage, wie stark dieser vereinfacht werden kann, damit er von den Lernenden handelnd untersucht werden kann. Die Schwierigkeit liegt im Ausloten der Vereinfachung, damit die Lernfähigkeit des Systems erhalten bleibt und als solche erkannt werden kann.

C.1 Künstliches Neuron mit einem Eingabewert

In seiner einfachsten Form ist ein künstliches Neuron mit nur einem gewichteten Eingabewert ohne Aktivierungsfunktion nicht von einer allgemeinen linearen Funktion zu unterscheiden. In dieser Form steht x für den Eingabewert, w für das Gewicht, b für den Schwellenwert und z ist der berechnete Ausgabewert:

$$z = x \cdot w + b$$

Mit dem in der Fragestellung beschriebenen Ziel, bietet es sich an, binäre Eingabewerte zu wählen. Diese vereinfachen das Lernverfahren bedeutsam, da der gewichtete Eingabewert im Falle einer 0 (Null) dem Schwellenwert entspricht und die Grösse sowie Polung des Gewichts entfallen. Im anderen Fall entspricht z der Summe aus w und b :

$$z = w + b, \text{ wenn } x = 1$$

$$z = b, \text{ wenn } x = 0$$

Beim überwachten Lernen wird ein sog. Trainingssatz mit bekannten Ein-/Ausgabepaaren (x/y) mit den berechneten Ausgabewerten (z) verglichen. Die Abweichung $y - z$ entspricht dem Fehler e . Mithilfe dieses Fehlers und dem Eingabewert wird anschliessend die Veränderung des neuen Gewichts und des Schwellenwerts berechnet:

$$w_{\text{neu}} = w_{\text{alt}} + x \cdot e$$

$$b_{\text{neu}} = b_{\text{alt}} + e$$

Diese Anpassung des Gewichts und des Schwellenwerts wird so häufig vorgenommen, bis die Fehlerrate für alle Ein-/Ausgabepaare im Trainingssatz einen gewünschten Wert

unterschreitet. Zur Veranschaulichung hier ein einfaches Beispiel, auf welches das künstliche Neuron trainiert werden kann:

x	y
0	1
1	0

Üblicherweise werden die Gewichte und Schwellenwerte eines neuronalen Netzes mit zufälligen Werten zwischen 0 und 1 initialisiert. Damit alle Berechnungen von Lernenden ab der Sekundarstufe 1 im Kopf vorgenommen werden können, wird an dieser Stelle eine weitere Vereinfachung vorgenommen: Gewicht und Schwellenwert starten mit dem Wert 0. Damit ergibt sich aus dem ersten Ein-/Ausgabepaar der Tabelle folgende Vorhersage für z:

$$z = 0 \cdot 0 + 0$$

Gemäss der Tabelle ist y jedoch 1. Daher ergibt sich mit der Formel $y - z$ folgender Fehler:

$$e = 1 - 0$$

Mithilfe des Fehlers lassen sich Gewicht und Schwellenwert neu berechnen:

$$w_{\text{neu}} = 0 + 0 \cdot 1$$

$$b_{\text{neu}} = 0 + 1$$

Das Gewicht wird somit aufgrund des inaktiven x nicht verändert, der Schwellenwert hingegen wird um 1 vergrössert. Damit kann nun überprüft werden, ob die Berechnung für das zweite Ein-/Ausgabepaar aus der Tabelle korrekt vorhergesagt wird oder eine weitere Anpassung der Gewichte nötig wird:

$$z = 1 \cdot 0 + 1$$

Da bei $x = 1$ eine Ausgabe von 0 erwartet wird, müssen Schwellenwert und Gewichte mithilfe des Fehlers erneut angepasst werden:

$$e = 0 - 1$$

$$w_{\text{neu}} = 0 + 1 \cdot (-1)$$

$$b_{\text{neu}} = 1 + (-1)$$

Da das künstliche Neuron nun beide Ein-/Ausgabepaare gesehen und die Werte entsprechend angepasst hat, könnte man erwarten, dass es zu beiden Eingabemöglichkeiten den korrekten Ausgabewert vorhersagt. Beim Einsetzen des ersten Wertepaars wird jedoch folgendes berechnet:

$$z = 0 \cdot (-1) + 0$$

Der Fehler beträgt erneut +1, da aber $x = 0$ ist, wird dieses Mal nur der Schwellenwert angepasst und es entsteht folgende Funktionsgleichung, die für beide Eingabewerte die korrekten Ausgangswerte berechnet:

$$z = -1 \cdot x + 1$$

Das künstliche Neuron ist nun auf die Werte aus der Tabelle angepasst. In der Schulmathematik hätte man das Problem mithilfe der beiden Wertepaare als Punkte im kartesischen Koordinatensystem gelöst und die beiden Begriffe Steigung sowie y-Achsenabschnitt benutzt. Das Beispiel mit nur einem Eingabewert eignet sich daher auch mit einem noch so spannenden Alltagsbezug eher nicht, um künstliche Neuronen zu behandeln, wenn die Lernenden bereits mit anderen Lösungswegen vertraut sind.

C.2 Lernrate und Aktivierungsfunktion

Für das vorangegangene Beispiel wurden zu Gunsten der Einfachheit zwei relevante Elemente künstlicher Neuronen weggelassen: Lernrate und Aktivierungsfunktion. Die Lernrate η (eta) bestimmt, wie stark die Gewichte und Schwellenwerte beim Training angepasst werden. Eine optimale Lernrate kann somit die Trainingsphase verkürzen. Für alle weiteren Beispiele wird mit einer Lernrate von $\eta = 1$ gerechnet, da sich die Lernenden auf diese Weise nicht mit einem zusätzlichen Faktor beschäftigen müssen.

$$w_{\text{neu}} = w_{\text{alt}} + x \cdot e \cdot \eta$$

$$b_{\text{neu}} = b_{\text{alt}} + e \cdot \eta$$

Die zweite Vereinfachung betrifft die Aktivierungsfunktion φ . Diese bestimmt, wie die Summe der gewichteten Eingangswerte z verarbeitet wird. Bisher wurde mit einer

Aktivierungsfunktion $\varphi(z) = z$ gerechnet. Für mehrschichtige Netzwerke ist die lineare Funktion jedoch nicht geeignet, da eine Komposition linearer Funktionen arithmetisch so umgeformt werden kann, dass diese in einer einzigen linearen Funktion dargestellt werden kann (Ekman, 2021). Dies wurde auch mit dem Eingangsbeispiel aufgezeigt. In der Praxis haben sich folgende Funktionstypen bewährt (Tabelle 3: Aktivierungsfunktionen):

Tabelle 3: Aktivierungsfunktionen

Schwellenwertfunktion

$$\varphi^{hlim}(z) = \begin{cases} 1 & \text{wenn } z \geq 0 \\ 0 & \text{wenn } z < 0 \end{cases}$$

Ein künstliches Neuron mit einer Schwellenwertfunktion (Engl. hard limit) gibt bei einer negativen Summe der gewichteten Eingabewerte 0 aus und bei positiven Werten 1.

Sigmoidfunktion

$$\varphi^{sig}(z) = \frac{1}{1 + e^{-az}}$$

Die Sigmoidfunktion wird vor allem aufgrund ihrer einfachen Differenzierbarkeit in künstlichen neuronalen Netzen verwendet. Bei stark positiven oder negativen Summen aus den gewichteten Eingabewerten unterscheidet sie sich jedoch nicht von der Schwellenwertfunktion.

Rectified Linear Unit

$$\varphi^{ReLU}(z) = \max(0, z)$$

Die Rectified Linear Unit (ReLU) wird besonders in Deep-Learning-Modellen erfolgreich eingesetzt. Negative Summen aus den gewichteten Eingabewerten führen wie bei der Schwellenwertfunktion zur Ausgabe 0, die positiven Werte werden hingegen nicht verändert.

Für Lernende ab der Sekundarschule reicht es, mit der einfachen Schwellenwertfunktion zu arbeiten, da sich damit die Anpassung der Gewichte auf Ein-/Ausgabemuster beim

Abbildung 14

Schwellenwertfunktion

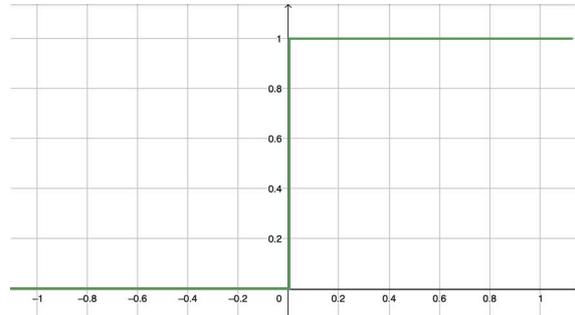


Abbildung 15

Sigmoidfunktion mit a = 5

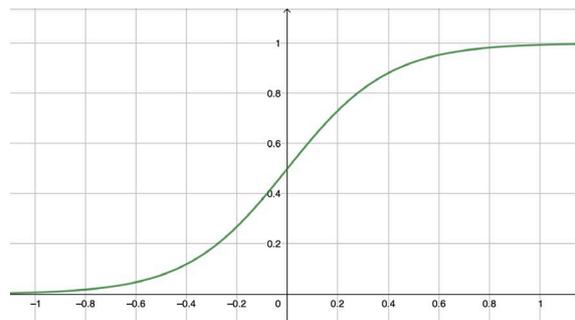
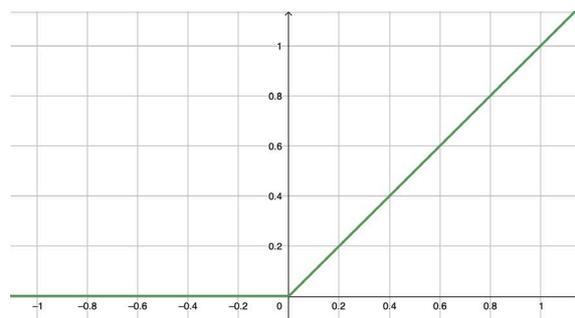


Abbildung 16

ReLU-Aktivierungsfunktion



überwachten Lernen zeigen lässt. Bei der entwickelten Unterrichtseinheit wird darauf verzichtet, die Fachbegriffe Aktivierungs- und Schwellenwertfunktion zu nutzen. Der Schwellenwert muss jedoch eingeführt werden, da sonst zu stark eingeschränkt wird, welche booleschen Funktionen das künstliche Neuron erlernen kann. Auf die Begründung für diese Wahl wird nachfolgend detailliert eingegangen.

C.3 Schwellenwert

Im Eingangsbeispiel lernt das künstliche Neuron nach drei Trainingsiterationen die Negation von booleschen Eingabewerten. Ohne einen variablen Schwellenwert würde der Lernalgorithmus nicht terminieren, da das Gewicht immer zwischen 0 und 1 oszillieren würde. Anstelle eines sich verändernden Schwellenwerts wird in der Praxis ein on-Neuron mit einem konstanten Eingang $x_0 = 1$ hinzugefügt: Der Bias. Die Veränderung des Schwellenwerts wird auf diese Weise mithilfe des Bias-Gewichts w_0 gesteuert. Wie relevant ist der Bias aber bei mehr als einem Eingabewert? Um dies zu untersuchen, wird der Einfluss des Bias auf die Lösbarkeit zweistelliger boolescher Funktionen mit heuristischen Lernverfahren betrachtet. Bei zwei Eingabewerten ergeben sich 16 mögliche Wahrheitstabellen (Tabelle 4).

Tabelle 4: Zweistellige Boolesche Funktionen⁶⁸

<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Kontradiktion</p>	X ₁	X ₂	y	0	0	0	0	1	0	1	0	0	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Konjunktion (AND)</p>	X ₁	X ₂	y	0	0	0	0	1	0	1	0	0	1	1	1	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Inhibition von x₁</p>	X ₁	X ₂	y	0	0	0	0	1	0	1	0	1	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Identität von x₁</p>	X ₁	X ₂	y	0	0	0	0	1	0	1	0	1	1	1	1
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	0																																																													
1	0	0																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	0																																																													
1	0	0																																																													
1	1	1																																																													
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	0																																																													
1	0	1																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	0																																																													
1	0	1																																																													
1	1	1																																																													
<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Inhibition von x₂</p>	X ₁	X ₂	y	0	0	0	0	1	1	1	0	0	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Identität von x₂</p>	X ₁	X ₂	y	0	0	0	0	1	1	1	0	0	1	1	1	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Antivalenz (XOR)</p>	X ₁	X ₂	y	0	0	0	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Disjunktion (OR)</p>	X ₁	X ₂	y	0	0	0	0	1	1	1	0	1	1	1	1
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	1																																																													
1	0	0																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	1																																																													
1	0	0																																																													
1	1	1																																																													
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	1																																																													
1	0	1																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	0																																																													
0	1	1																																																													
1	0	1																																																													
1	1	1																																																													
<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Nihilation (NOR)</p>	X ₁	X ₂	y	0	0	1	0	1	0	1	0	0	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Äquivalenz (XNOR)</p>	X ₁	X ₂	y	0	0	1	0	1	0	1	0	0	1	1	1	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Negation von x₂ NOT(x₂)</p>	X ₁	X ₂	y	0	0	1	0	1	0	1	0	1	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Replikation</p>	X ₁	X ₂	y	0	0	1	0	1	0	1	0	1	1	1	1
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	0																																																													
1	0	0																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	0																																																													
1	0	0																																																													
1	1	1																																																													
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	0																																																													
1	0	1																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	0																																																													
1	0	1																																																													
1	1	1																																																													
<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Negation von x₁ NOT(x₁)</p>	X ₁	X ₂	y	0	0	1	0	1	1	1	0	0	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Implikation</p>	X ₁	X ₂	y	0	0	1	0	1	1	1	0	0	1	1	1	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> <p>Exklusion (NAND)</p>	X ₁	X ₂	y	0	0	1	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr><th>X₁</th><th>X₂</th><th>y</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Tautologie</p>	X ₁	X ₂	y	0	0	1	0	1	1	1	0	1	1	1	1
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	1																																																													
1	0	0																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	1																																																													
1	0	0																																																													
1	1	1																																																													
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	1																																																													
1	0	1																																																													
1	1	0																																																													
X ₁	X ₂	y																																																													
0	0	1																																																													
0	1	1																																																													
1	0	1																																																													
1	1	1																																																													

Die Ein-/Ausgabepaare der 16 Wahrheitstabellen wurden mithilfe des eingangs beschriebenen heuristischen Lernalgorithmus geprüft. Grün steht für jene booleschen Funktionen, deren Fehlerrate auch ohne variablen Schwellenwert konvergiert (Abbildung 17 und Abbildung 18),

⁶⁸ https://de.wikipedia.org/wiki/Boolesche_Funktion

während für die blauen Kombinationen der Bias zwingend gebraucht wird, um den Fehler zu minimieren (Abbildung 19 und Abbildung 20). Die Fehlerrate bei den sechs hellgrünen Varianten konvergiert im Durchschnitt ohne Bias etwas schneller.

Abbildung 17

Inhibition von x_1 mit Bias

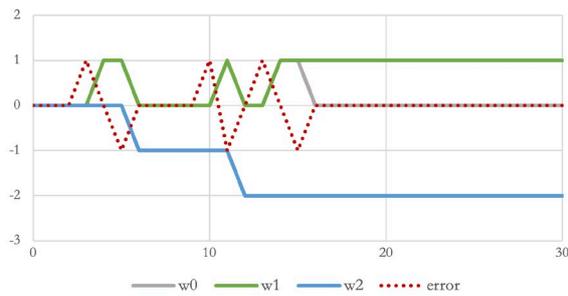
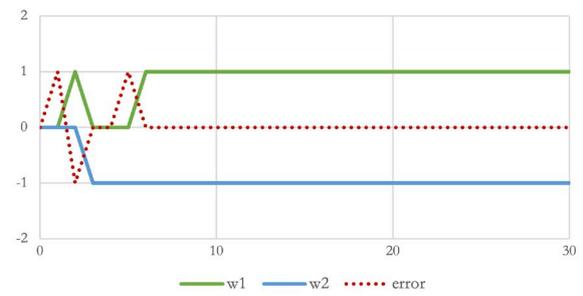


Abbildung 18

Inhibition von x_1 ohne Bias



Da bei heuristischen Lernverfahren jeweils zufällige Ein-/Ausgabepaare gewählt werden, unterscheiden sich die Trainingsdauer und teilweise auch die Gewichte am Ende einer Trainingseinheit. Eine Trainingseinheit gilt als beendet, wenn der berechnete Ausgabewert des künstlichen Neurons mit dem erwarteten Wert für alle vier Ein-/Ausgabepaare übereinstimmt. Auf den Abbildungen ist dies daran erkennbar, dass sich der Fehler nicht mehr verändert.

Abbildung 19

Konjunktion (AND) mit Bias

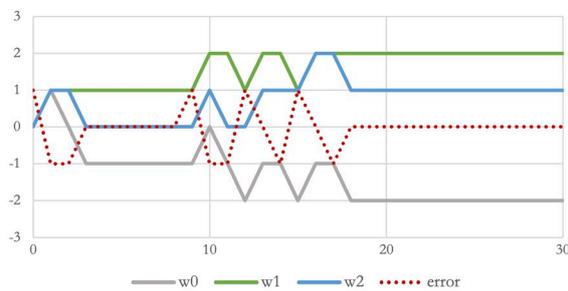
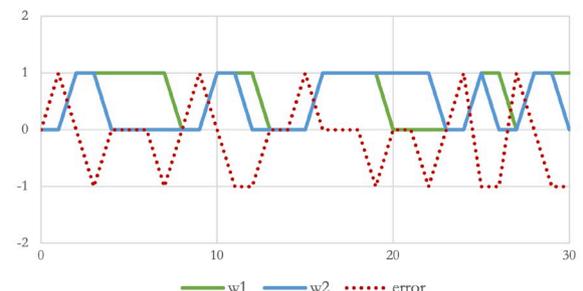


Abbildung 20

Konjunktion (AND) ohne Bias



Auf Abbildung 21 ist zu sehen, dass die Gewichte bei der Antivalenz keine Werte finden, welche den Fehler für alle vier Zustände minimiert. Der Grund dafür liegt darin, dass das künstliche Neuron im Grunde nichts weiter als eine lineare Klassifikation vornimmt. Die vier Zustände der Antivalenz lassen sich jedoch nicht linear separieren. Diese Unlösbarkeit führte in den 1970er-Jahren dazu, dass die Forschung mit künstlichen neuronalen Netzen für mehr als zehn Jahre fast eingestellt wurde.⁶⁹

Abbildung 21

Antivalenz (XOR) mit Bias

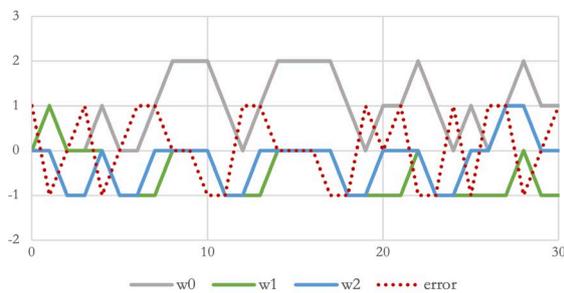
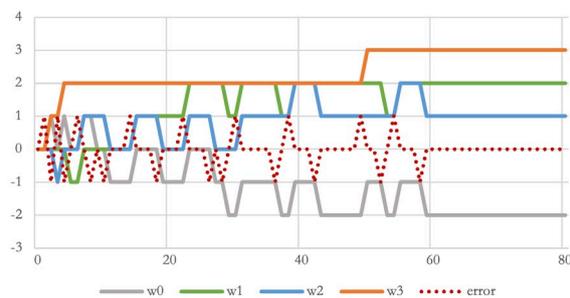


Abbildung 22

Training mit 3 Inputs



C.4 Ein künstliches Neuron mit mehr als 2 Eingabewerten

Gab es bei einem künstlichen Neuron mit zwei binären Eingabewerten und einem binären Ausgabewert noch 16 verschiedene Ausprägungen (Tabelle 4), so sind es bei drei Eingabewerten bereits $2^3 = 8$. Um die abstrakten Zusammenhänge mit Bedeutung zu füllen, wird in der vorliegenden Arbeit das Beispiel Movie Recommender herangezogen. Die Eingabewerte entsprechen drei sich möglichst nicht überschneidenden Film-Genres, z. B. $x_1 = \text{Comedy}$, $x_2 = \text{Action}$ und $x_3 = \text{Horror}$ und der Ausgabewert gibt an, ob eine Person das Genre oder die Genrezusammensetzungen mag oder nicht.

⁶⁹ Vgl. Kapitel 1.1.3

Tabelle 5: Beispiel einer möglichen Filmgeschmacksausprägung

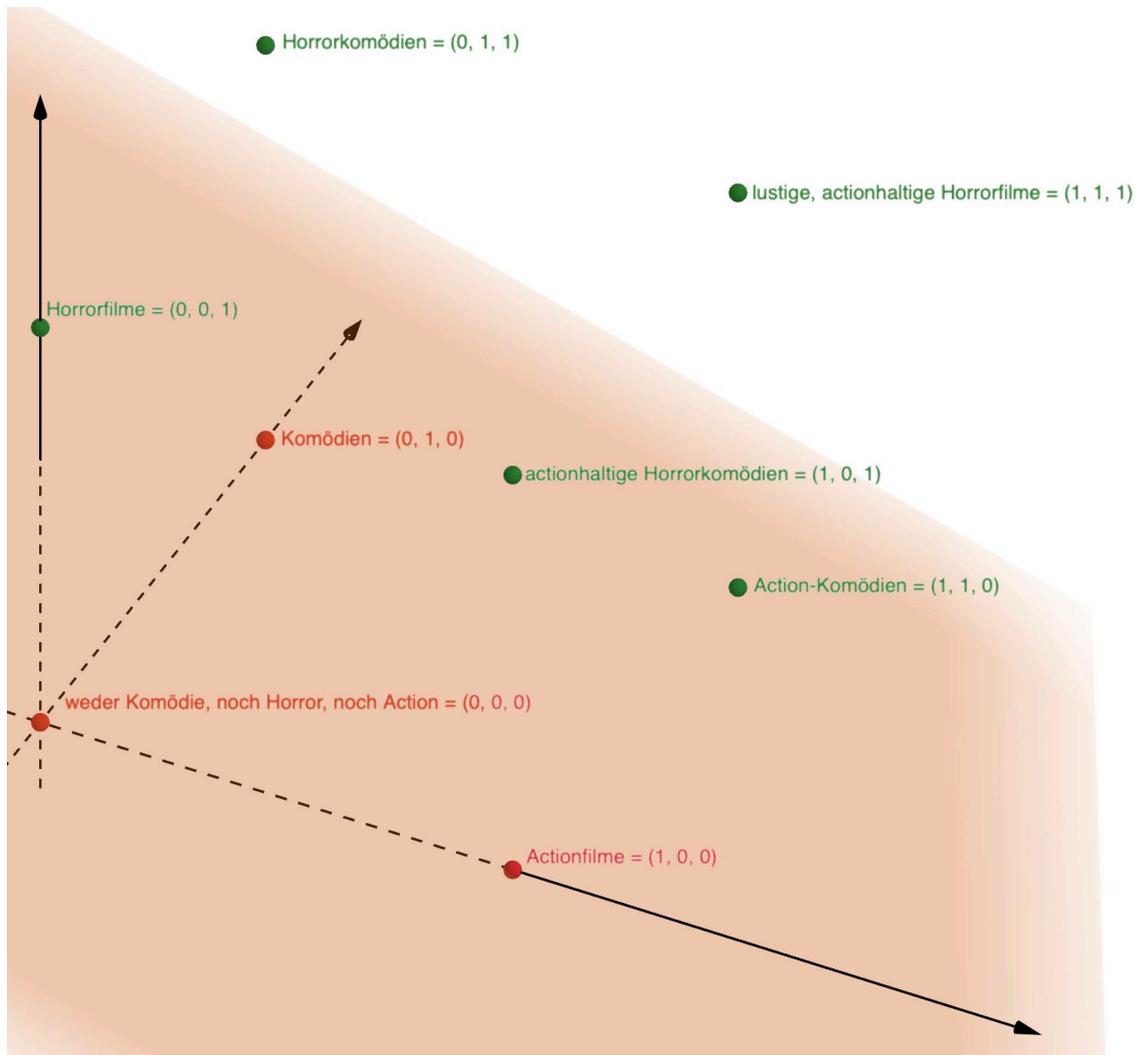
x_1	x_2	x_3	y	Beschreibung der Ein-/Ausgabezuordnung <i>Person mag...</i>
0	0	0	0	<i>...weder Komödien noch Horror- oder Actionfilme</i>
0	0	1	1	<i>...Horrorfilme</i>
0	1	0	0	<i>...Komödien</i>
0	1	1	1	<i>...Horrorkomödien</i>
1	0	0	0	<i>...Actionfilme</i>
1	0	1	1	<i>...actionbaltige Horrorfilme</i>
1	1	0	1	<i>...Action-Komödien</i>
1	1	1	1	<i>...lustige actionbaltige Horrorfilme</i>

Obwohl der tatsächliche Netflix-Algorithmus viele weitere Zuschauerdaten einbezieht, welche die Empfehlungen bestimmen, die den Zuschauenden angezeigt werden (Plummer, 2017), so ist er doch der oben beschriebenen Ausgangslage in einem Punkt ähnlich: Er bildet Zuschauercluster mit ähnlichen Filmvorlieben. Um eine Nähe zu den Lernenden aufzubauen, können für die verschiedenen Cluster Personas entwickelt werden, welche ihnen dabei helfen, den eigenen Filmgeschmack einem Zuschauercluster zuzuordnen.

Tabelle 5 zeigt eine von 256 verschiedenen Filmgeschmacksausprägungen, welche in drei Genres klassifiziert werden kann. Um die Gewichte eines künstlichen Neurons mithilfe des oben beschriebenen heuristischen Lernverfahrens so anzupassen, dass es die Filme für diese Ausprägung korrekt klassifiziert, braucht es ungefähr 60 Trainingsiterationen (siehe Abb. 9). Grafisch dargestellt lautet die Aufgabe für das künstliche Neuron, jene Parameter der Ebenengleichung zu finden, welche die positiv bewerteten Filmgenres von den negativen trennt (siehe Abbildung 23).

Abbildung 23

Filmklassen als Punkte im dreidimensionalen kartesischen Koordinatensystem mit trennender Ebene



In Abbildung 23 sind die positiv bewerteten Filmklassen grün eingefärbt, während die negativen rot sind. Die Ebenengleichung lautet mit den in Abbildung 22 berechneten Parametern:

$$2x + 1y + 3z = 2$$

C.5 Zusammenfassung

Eingangs wurde die Frage gestellt, wie stark der Lernalgorithmus eines künstlichen neuronalen Netzes vereinfacht werden kann, damit er mit Lernenden ab der Sekundarstufe 1

behandelt werden kann. Als erste Vereinfachung wurde anstelle eines mehrschichtigen Netzes nur ein einzelnes Neuron verwendet. Auf diese Vereinfachung wurde im Text nicht eingegangen. Gegen ein Netz mit mehreren Schichten spricht jedoch, dass für den Lernprozess das Prinzip der Backpropagation hätte eingeführt werden müssen, was wiederum ohne partielle Ableitung nicht möglich ist.

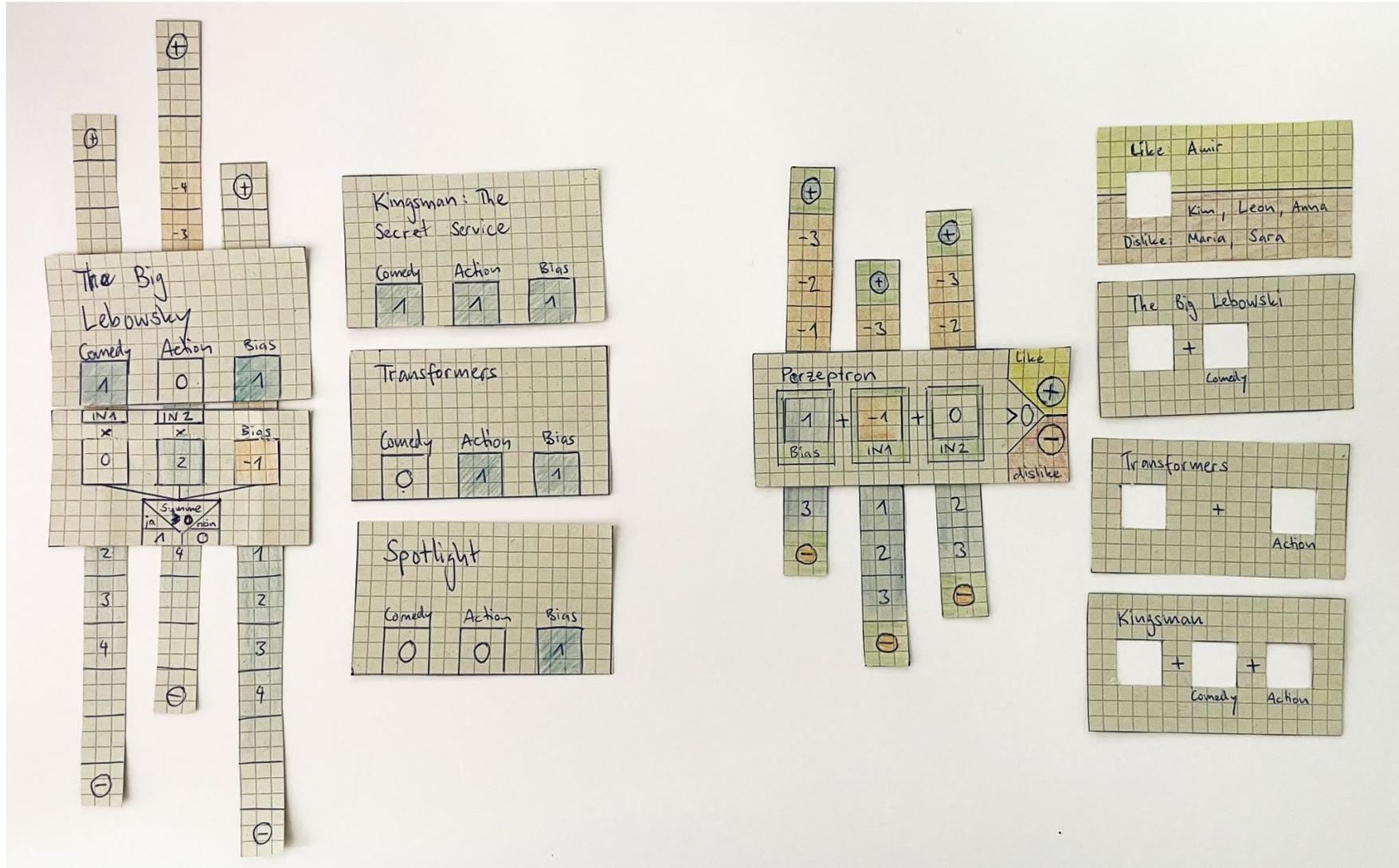
Die zweite und dritte Vereinfachung bestand darin, als Eingabe nur dichotome Werte und als Gewichte nur ganze Zahlen zuzulassen. Damit lässt sich die Summe der gewichteten Eingabewerte ohne Taschenrechner von Lernenden ab der Sekundarschule berechnen. Ausserdem wird die Anpassung der Gewichte beim Training vereinfacht, da die Fehlerrate nur drei Zustände einnehmen kann: -1 , 0 und 1 .

Vereinfachungen vier und fünf betreffen die Lernrate und Aktivierungsfunktion. Wieder wurden möglichst einfache Werte gewählt, damit alle Berechnungen im Kopf gelöst werden können. Durch eine Lernrate 1 kann diese ohne Erklärung ausgeblendet werden. Als Aktivierungsfunktion wurde die Schwellenwertfunktion gewählt. Damit wird wiederum die Ausgabe des künstlichen Neurons auf die Werte 0 und 1 reduziert, was im beschriebenen Unterrichtsbeispiel «Movie Recommender» mit «dislike» und «like» übersetzt werden kann. Als nächstes wurde der Einfluss des Schwellenwerts beschrieben. Es wurde aufgezeigt, dass sich durch das Hinzufügen eines on-Neurons (Bias) der Schwellenwert indirekt anpassen lässt. Wird der Bias weggelassen, hat dies zur Folge, dass 10 von 16 zweistellige boolesche Funktionen durch den heuristischen Lernalgorithmus mit den beschriebenen Vereinfachungen nicht erlernt werden können. Diese starke Einschränkung führt dazu, dass das ohnehin eingeschränkte Lernverhalten des stark vereinfachten Neurons von den Lernenden vermutlich nicht als Lernen wahrgenommen wird. Daher wird für das Unterrichtsbeispiel «Movie Recommender» nicht auf den Bias verzichtet.

Die letzte Frage bezüglich optimaler Vereinfachung betrifft die Anzahl Eingabewerte. Es wurde aufgezeigt, dass das künstliche Neuron mit Bias 15 von 16 zweistellige boolesche Funktionen in 10 bis 20 Iterationen erlernen kann. Bei drei Eingabewerten wurden für gewisse

Ausprägungen bereits 60 Trainingsschritte benötigt (Abbildung 22). Da die manuelle Anpassung der Gewichte durch Vergleich der berechneten und tatsächlichen Ausgabewerte mit etwas Übung etwa 20 Sekunden dauert, würde dieser Prozess mit 60 Iterationen im Unterricht zu viel Zeit in Anspruch nehmen. Ausserdem kann davon ausgegangen werden, dass die Fehlerquote mit jedem zusätzlichen Trainingsschritt ebenfalls ansteigt. Aus diesem Grund wird für das Unterrichtsbeispiel «Movie Recommender» mit zwei Eingabewerten gerechnet. Dies entspricht einer Zuordnung von Filmklassen mit zwei Ausprägungen in «like» und «dislike». Als Filmklassen wurden die beiden Genres «Action» und «Comedy» gewählt.

D Perzeptron unplugged Prototyp 1 und 2



E Perzeptron unplugged final 3 und 4

