Deep Learning

Manuel Renold

In diesem Kapitel wird eine Einführung in das Gebiet des Deep Learnings gegeben. Der Fokus liegt auf dem potenziellen Einsatz dieser neuen Technologie in kleinen und mittleren Unternehmen. Im ersten Teil des Kapitels werden jedoch zuerst die grundlegenden Prinzipien des Deep Learning erörtert. Der zweite Teil des Kapitels widmet sich den heutigen Möglichkeiten und den zukünftigen Entwicklungen der Technologie für verschiedene Geschäftsbereiche, einschliesslich der Betrachtung von potenziellen Chancen und Herausforderungen für KMU.

Maschinen Lernen!

Das Interesse an maschinellem Lernen und insbesondere an Deep Learning [2][10] ist im letzten Jahrzehnt exponentiell gestiegen. Trotz der intensiven Diskussion in der breiten Öffentlichkeit über die Möglichkeiten der neuen Technologie wird oft verkannt, was tatsächlich möglich ist.

Im Kern geht es beim maschinellen Lernen und insbesondere beim Deep Learning darum, mithilfe von Algorithmen Informationen aus Rohdaten zu extrahieren und diese in Form eines Modells abzubilden [5]. Dieses Modell wird anschliessend genutzt, um Rückschlüsse auf andere, noch nicht modellierte Daten zu ziehen. Die Methodik stammt aus dem statistischen Lernen, genauer gesagt aus der statistischen Modellbildung. Auch dort versucht man, aus Stichproben einer Grundgesamtheit (z. B. der Bevölkerung) Merkmale herauszufiltern und diese in einem statistischen Modell (z. B. einer Verteilungsfunktion des Einkommens der Bevölkerung) abzubilden. So kann man z. B. die durchschnittliche Lohnentwicklung über Jahre beobachten und dann statistische Vorhersagen treffen, indem man diese Modelle befragt.

Neuronale Netze als Hauptkomponente des Deep Learnings stellen dabei eine spezifische Modellklasse des maschinellen Lernens dar und werden seit fast 80 Jahren erforscht [1]. Die grundlegende Einheit eines neuronalen Netzes ist dabei ein Knoten (Node), der Ähnlichkeiten zu einem biologischen Neuron aufweist. Auch die Verbindungen zwischen den Neuronen (Knoten) sowie deren Entwicklung über die Zeit durch Lernen sind an biologischen Vorbildern orientiert. In den folgenden Kapiteln werden wir nun detaillierter auf die Funktionsweise dieser Elemente eingehen.

Die Geschichte des Perceptrons

Die Anfänge

Während des Zweiten Weltkrieges legten der Neurophysiologe Warren McCulloch (1889–1969) und der Logiker Walter Pitts den Grundstein für künstliche neuronale Netze [1][16]. Warren McCulloch war ein Pionier auf dem Gebiet der theoretischen Neurowissenschaften und der künstlichen Intelligenz. McCulloch interessierte sich insbesondere auch für die philosophischen Implikationen seiner Arbeit. Er untersuchte Fragen wie «Wie entsteht Bewusstsein aus der Aktivität von Neuronen?» oder ob man menschliches Denken durch logische Systeme nachbilden kann. Seine Arbeiten trugen dazu bei, die Idee zu entwickeln, dass das Gehirn eine Art «biologische Maschine» ist, dessen Funktionen prinzipiell nachgebildet werden kann.

Seine Forschung konzentrierte sich insbesondere auf das Verständnis der Funktionsweise des menschlichen Gehirns und die mathematische Modellierung neuronaler Prozesse. Zusammen mit Pitts entwickelte er im Jahre 1943 die sogenannte McCulloch-Pitts-Zelle. Diese Zelle war im Prinzip ein einfaches Modell eines künstlichen Neurons, das als logisches **Schwellenwertelement** fungiert. (Der Schwellwert ist vergleichbar mit dem **Aktivierungspotential** eines biologischen Neurons.) Dieses Modell verfügt über mehrere binäre Eingänge («aktiv» oder «inaktiv») und einen einzigen Ausgang, der entweder den Zustand «wahr/aktiv» oder «falsch/inaktiv» annehmen kann. [1]

Ein Neuron «feuert» (d. h. gibt «wahr» aus), wenn die Summe der Eingangssignale einen bestimmten **Schwellenwert** überschreitet. Dieses Konzept ist eine direkte Analogie zum biologischen **Aktionspotenzial**, das eine Nervenzelle bei einer kritischen Änderung ihres Membranpotentials auslöst. [13]

McCulloch war davon überzeugt, dass sich die Funktionsweise des Gehirns durch mathematische und logische Prinzipien beschreiben lässt. Dies war ein radikaler Ansatz zu einer Zeit, in der das Gehirn hauptsächlich aus biologischer und chemischer Perspektive betrachtet wurden. Er und Pitts konnten zeigten, dass sich durch die Kombination mehrerer solcher Neuronen grundlegende logische Funktionen wie **UND**, **ODER** und **NICHT** abbilden lassen [16].

Beim Kombinieren und Hintereinanderschalten von diesen McCulloch-Pitts-Zellen, im Prinzip künstliche Neuronen, konnten also einfache logische Schaltungen aufgebaut werden, vergleichbar mit den logischen Schaltungen in der Digitalelektronik in heutigen Prozessoren.

Zu dieser Zeit hätte also die KI-Revolution beginnen können. 1943 waren immer noch einfache, aber energiehungrige Computer mit Röhren in Betrieb, die Grundlagen der heutigen auf Silizium-Transistoren beruhenden Technologie wurden gerade erfunden. Im Jahr 1925 reichte Julius Edgar Lilienfeld die ersten Patente für ein Prinzip ein, das dem heutigen Feldeffekttransistor (FET) ähnelt. In seinen Arbeiten beschrieb er ein elektronisches Bauelement, das Funktionen einer Elektronenröhre übernimmt. Allerdings war die technologische Umsetzung von Feldeffekttransistoren zu dieser Zeit noch nicht machbar. Erst 1947 präsentierten dann John Bardeen, William Shockley und Walter Brattain von den Bell Laboratories den ersten funktionierenden Transistor. Für diese bahnbrechende Erfindung, die unabhängig von den früheren theoretischen Arbeiten Lilienfelds und Heils aus den 1920er Jahren entwickelt wurde, erhielten sie 1956 den Nobelpreis für Physik.

Warum das Gebiet der künstlichen neuronalen Netze nach der Entdeckung der McCulloch-Pitts-Zelle trotzdem vollständig zum Erliegen kam, ist heute als XOR-Problem bekannt und sollte erst mit sogenannten mehrlagigen neuronalen Netzen gelöst werden.

Wenn man demnach logische Schaltungen oder «künstliche Neuronenhirne» aufbauen will, genügt es nicht, nur die Funktionen **UND**, **ODER** und **NICHT** abzubilden zu können, sondern man benötigt auch eine sogenannte **XOR**-Funktion. Genauer betrachtet konnte die McCulloch-Pitts-Zelle nur mathematisch linear trennbare Probleme lösen, während das XOR-Problem (exklusives ODER) nicht linear trennbar ist. Dies bedeutete, dass eine einfache McCulloch-Pitts-Zelle das XOR-Problem nicht lösen konnte.

1949 erweiterte der Psychologe Donald O. Hebb die Ideen von McCulloch und Pitts durch seine Hypothese, dass Lernen auf der Veränderung der synaptischen Verbindungen zwischen Neuronen beruht [4]. Nach Hebb wird die Stärke einer Synapse durch die gleichzeitige Aktivität der prä- und postsynaptischen Neuronen bestimmt. Diese Theorie, bekannt als **Hebb'sche Lernregel**, lieferte eine wichtige Grundlage für das Verständnis von Lernprozessen in biologischen und künstlichen Systemen. Das XOR-Problem bestand aber weiterhin.

Erst in den 1960er Jahren näherte man sich der Lösung durch die Arbeit von Marvin Minsky und Seymour Papert in ihrem Buch «Perceptrons» (1969) theoretisch analysiert hatten. Sie zeigten, dass einlagige Perceptronen das XOR-Problem nicht lösen können, und wiesen auf die Notwendigkeit von mehrlagigen Netzwerken hin. [2]

Die praktische Lösung des XOR-Problems erfolgte jedoch erst mit der Entwicklung von Trainingsalgorithmen für mehrlagige neuronale Netze, insbesondere durch die Einführung des Backpropagation-Algorithmus in den 1980er Jahren [7] [17]. Dieser Algorithmus, der von Forschern wie David Rumelhart, Geoffrey Hinton und Ronald Williams populär gemacht wurde, ermöglichte es, die Gewichte in mehrlagigen Netzwerken effizient anzupassen und so das XOR-Problem zu lösen.

KI-Winter und der Aufstieg von Deep Learning

In den 1980er und frühen 1990er Jahren wurden dann bedeutende Fortschritte in der Architektur und den Trainingsmethoden neuronaler Netze erzielt. Die Einführung von mehrlagigen Perceptronen (Multi-Layer Perceptrons, MLPs) und die Entwicklung des Backpropagation-Algorithmus ermöglichten es, komplexe nichtlineare Zusammenhänge mit neuronalen Netzen zu modellieren. Diese Innovationen erweiterten die Anwendungsmöglichkeiten der Netze erheblich und führten zu Erfolgen in Bereichen wie Mustererkennung und Signalverarbeitung.

Trotz dieser Fortschritte blieb die breite Anwendung neuronaler Netze jedoch begrenzt. Die damalige Hardware war nicht leistungsfähig genug, um die rechenintensiven Trainingsprozesse effizient zu bewältigen. Zudem erforderte das Training grosser Modelle enorme Datenmengen, die oft nicht verfügbar waren. Diese praktischen Herausforderungen führten zu einem vorübergehenden Rückgang des Interesses an neuronalen Netzen, der als «zweiter KI-Winter» bezeichnet wird.

Zu Beginn des 21. Jahrhunderts erlebte das Feld der neuronalen Netze eine spektakuläre Renaissance. Der exponentielle Anstieg der Rechenleistung, insbesondere durch die Nutzung von GPUs (Grafikprozessoren) und später TPUs (Tensor Processing Units), ermöglichte die Verarbeitung grosser Datenmengen in bisher unvorstellbarer Geschwindigkeit. Gleichzeitig führte die zunehmende Verfügbarkeit von Big Data und Internet zu einer Fülle von Trainingsdaten, die für das Training komplexer Modelle unerlässlich sind.
Diese technologischen Fortschritte lösten eine «kambrische Explosion» von Innovationen im Bereich des maschinellen Lernens aus. Deep Learning, eine Unterkategorie neuronaler Netze, etablierte sich als führender Ansatz und revolutionierte zahlreiche Anwendungsbereiche. Durch die Verwendung von tiefen neuronalen Netzen (Deep Neural Networks, DNNs) mit vielen versteckten Schichten (hidden layers) konnten bisher unerreichte Genauigkeiten in Aufgaben wie Bilderkennung, Sprachverarbeitung und autonomen, selbststeuernden Systemen erzielt werden. [2]

Deep Learning erlangte besondere Aufmerksamkeit durch spektakuläre Erfolge in internationalen Wettbewerben. So gewann das AlexNet-Modell 2012 [14] den ImageNet-Wettbewerb und demonstrierte die Überlegenheit tiefer neuronaler Netze in der Bildklassifizierung. Ähnliche Erfolge folgten in Bereichen wie Spracherkennung (z. B. durch Google DeepMind) und strategischen Spielen (z. B. AlphaGo).

Bis 2017 hatte sich Deep Learning als dominierende Technologie im Bereich des maschinellen Lernens etabliert. Heute ist es in nahezu allen Anwendungsfeldern präsent, von der Medizin und Robotik bis hin zur Finanzindustrie und Unterhaltung. Die kontinuierliche Weiterentwicklung von Architekturen wie Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) und Transformer-Modellen treibt die Grenzen des Möglichen immer weiter hinaus. [14]

Neuronale Netze

Künstliche neuronale Netze (KNN) sind ein zentrales Konzept im Bereich des maschinellen Lernens und der künstlichen Intelligenz. Wir haben gesehen, dass ihre Entwicklung bis in die Mitte des 20. Jahrhunderts zurückreicht und massgeblich durch die Arbeit von Pionieren wie Warren McCulloch, Walter Pitts, Donald Hebb und Frank Rosenblatt geprägt wurde. Diese Fortschritte bilden die Grundlage für moderne Architekturen wie Deep Learning, die heute in nahezu allen Bereichen der KI-Technologie Anwendung finden.

Um das heutige und zukünftige Potenzial der Technologie der neuronalen Netze und insbesondere des Deep Learning abzuschätzen zu können, ist es sehr hilfreich, die grundlegenden Mechanismen von neuronalen Netzen zu betrachten.

Obschon seit der Entwicklung des Perceptrons und der daraus folgenden künstlichen neuronalen Netze diese stark an ihr biologisches Vorbild des Säugetierhirns anlehnt, gibt es dennoch grosse Unterschiede in der Funktionsweise, wie im Folgenden erläutert wird.

Biologische Neuronale Netze

Das Säugetierhirn besteht aus Milliarden von Nervenzellen (Neuronen), die über Synapsen miteinander verbunden sind. Ein Neuron besteht dabei aus drei Hauptkomponenten: den Dendriten, dem Zellkörper (Soma) und dem Axon. Die Dendriten sind verzweigte Fortsätze, die Signale von anderen Neuronen empfangen. Der Zellkörper verarbeitet diese eingehenden Signale, und das Axon leitet die verarbeiteten Signale als Spikes an andere Neuronen weiter. Am Ende des Axons befinden sich die Synapsen, die Verbindungsstellen zwischen den Neuronen, an denen die Signalübertragung stattfindet. [3]

Die Kommunikation zwischen den Neuronen erfolgt durch Aktionspotentiale (auch «Spikes» genannt), kurze elektrische Impulse, die entlang der Nervenfasern wandern. Die Stärke der Verbindungen zwischen Neuronen (Synapsen) unterliegt einer kontinuierlichen Veränderung, die als synaptische Plastizität bezeichnet wird und die Grundlage für Lernen und Gedächtnis bildet. [11]

Dabei sind Spikes die grundlegenden Einheiten der neuronalen Kommunikation im Gehirn. Sie entstehen durch die Depolarisation der Zellmembran, breiten sich entlang der Axone aus und werden an Synapsen in chemische Signale umgewandelt. Vergleichbar wie bei der die Integration exzitatorischer und inhibitorischer Signale bestimmt dies, ob ein Neuron feuert. Diese Prozesse ermöglichen nicht nur die schnelle und effiziente Übertragung von Informationen, sondern auch die Plastizität [3], die Lernen und Gedächtnis ermöglicht. Spikes sind somit das Herzstück der neuronalen Aktivität und der komplexen Funktionen des Gehirns.

Spikes sind durch ihre Impulsartigkeit (es fliesst nur ganz kurz elektrischer Strom) eine sehr energieeffiziente Form der Kommunikation, da sie nur kurze elektrische Impulse und nicht kontinuierlich aktiv sind.

Ein Spike wird ausgelöst, wenn die Summe der eingehenden Signale über die Dendriten einen bestimmten Spannungs-Schwellenwert überschreitet. Dieses Phänomen wird als «Alles-oder-Nichts-Prinzip» bezeichnet: Entweder wird ein Spike ausgelöst, oder es passiert nichts. Die Stärke des Spikes ist immer gleich, unabhängig davon, wie stark das auslösende Signal war. [11]

Sobald ein Spike ausgelöst wird, breitet er sich entlang des Axons aus. Dies geschieht durch die sequenzielle Öffnung und Schliessung von Ionenkanälen, die eine Welle der Depolarisation erzeugen. Viele Axone sind von einer isolierenden Schicht aus Myelin umhüllt, die die Fortleitung des Spikes beschleunigt.

Diese Art von Informationsübertragung ermöglicht es dem Gehirn, komplexe neuronale Vorgänge mit relativ geringem Energieaufwand durchzuführen. Die Effizienz der Spike-basierten Kommunikation ist einer der Gründe, warum das menschliche Säugetierhirn trotz seiner enormen Komplexität nur rund 20 Watt Leistung in der Stunde verbraucht. Zum Vergleich: Das Training eines ChatGPT4-Modells benötigt ca. 1.3 Megawatt Leistung. Eine Abfrage (Interferenz) braucht ca. 0.3 bis 1 kWh.

Ein wichtiger Mechanismus der Neuroplastizität ist die Langzeit-Potenzierung (LTP), bei der die wiederholte Aktivierung einer Synapse ihre Übertragungsstärke erhöht. Dies ermöglicht es dem Gehirn, Informationen energieeffizient und zeitabhängig-dynamisch zu speichern und abzurufen. [11]

Die Methodik, neuronale Netze mit Hilfe von Spikes und neuronaler Plastizität aufzubauen, erscheint vielversprechend. Warum aber die heutigen Netze nicht oder noch nicht so aufgebaut sind, soll nun erklärt werden.

Spike Neuronal Networks (SNN)

Spiking neural networks (SNNs) sind eng an der biologischen Funktionsweise des Gehirns orientiert und nutzen diskrete Spikes (Aktionspotentiale) zur Informationsverarbeitung [13]. Theoretisch sind sie energieeffizienter als klassische neuronale Netze (ANNs), doch mehrere Nachteile schränken ihre breite Anwendung ein. Eines der grössten Probleme ist die Komplexität des Trainings: Herkömmliche Methoden wie Backpropagation, die bei ANNs gut funktionieren, lassen sich nicht direkt auf SNNs anwenden. Stattdessen werden spezielle Trainingsmethoden wie Spike-Timing-Dependent Plasticity (STDP) verwendet, die jedoch rechenintensiv und weniger effizient sind [13]. [12]

Ein weiterer Nachteil ist die fehlende Hardware-Unterstützung. SNNs erfordern spezialisierte Hardware wie neuromorphe Chips (z. B. Intels Loihi), die jedoch noch nicht weit verbreitet und oft teuer sind. Die zeitabhängige Natur von SNNs macht ihre Modellierung und Simulation zudem deutlich komplexer als die von klassischen ANNs, die auf kontinuierlichen Werten basieren und einfacher zu handhaben sind.

SNNs eignen sich besonders gut für Aufgaben, die zeitabhängige Daten verarbeiten, wie Echtzeit-Signalverarbeitung oder neuromorphe Anwendungen. Für viele klassische Probleme des maschinellen Lernens wie Bilderkennung oder NLP (Natural Langauge Processing) sind sie jedoch oft weniger effizient. Hinzu kommt, dass es nur wenige etablierte Entwicklungs-Frameworks und Software-Bibliotheken für SNNs gibt, was die Entwicklung und Implementierung erschwert. Entwickler müssen häufig auf experimentelle oder spezialisierte Tools zurückgreifen.

Zwar sind SNNs theoretisch energieeffizienter, da sie während der Spikes aktiv sind, in der Praxis ist dieser Vorteil jedoch oft nicht ausreichend, um ihre geringere Leistungsfähigkeit bei vielen Aufgaben auszugleichen. Klassische ANNs liefern oft bessere Ergebnisse bei vergleichbarem oder sogar geringerem Energieaufwand, insbesondere auf moderner Hardware. Zudem ist die Forschung zu SNNs noch relativ jung, und viele Konzepte und Methoden sind noch nicht ausgereift. Es gibt offene Fragen zu Skalierbarkeit und Effizienz, und es dauert oft Jahre, bis neue Erkenntnisse aus der Forschung in praktische Anwendungen umgesetzt werden können.

Aktuell sind klassische ANNs die bevorzugte Wahl für die meisten Anwendungen des maschinellen Lernens, da sie bereits hervorragende Ergebnisse liefern und einfacher zu implementieren sind. Der Aufwand, SNNs zu entwickeln und einzusetzen, lohnt sich oft nicht, zumal die Hardware- und Software-Unterstützung noch unzureichend ist. Dennoch bieten SNNs interessantes Potenzial, insbesondere in Nischenanwendungen, bei denen Energieeffizienz und biologische Plausibilität entscheidend sind. Mit Fortschritten in der Forschung und der Entwicklung spezialisierter Hardware könnten SNNs in Zukunft an Bedeutung gewinnen und in spezialisierten Bereichen eine grössere Rolle spielen. Insbesondere in der Robotik, wo zeitabhängige und energiesparende Methoden gefragt sind, haben die Spike Neuronal Networks eine erfolgversprechende Zukunft. [12]

Artifizielle neuronale Netze (ANN)

Der Aufbau eines artifiziellen Neurons ist in Abbildung 1 zu sehen. Ein Single-Layer-Neuron, auch bekannt als Perceptron, ist ein grundlegendes Modell in der künstlichen neuronalen Netzwerkarchitektur. Es besteht aus mehreren Eingängen x1, x2,…, xn, die jeweils mit einem Gewicht ω_1, ω_2, …, ω_n versehen sind. Diese Gewichte bestimmen, wie stark jeder Eingang das Verhalten des Neurons beeinflusst. Zusätzlich gibt es oft einen Bias b, der als eine Art Schwellenwert fungiert und die Aktivierung des Neurons beeinflusst.

Der Netto-Input des Neurons wird berechnet, indem die gewichtete Summe der Eingänge gebildet und der Bias hinzugefügt wird. Dieser Netto-Input wird dann in eine Aktivierungsfunktion eingespeist, um die Ausgabe des Neurons zu bestimmen. Im Fall eines Perceptrons wird häufig eine **Step-Funktion** verwendet. Die Step-Funktion gibt entweder den Wert 0 oder 1 zurück, abhängig davon, ob der Netto-Input einen bestimmten Schwellenwert überschreitet:
Die Step-Funktion wird verwendet, weil das Perceptron ursprünglich als binärer Klassifikator entwickelt wurde. Es soll entscheiden, ob ein Eingabevektor zu einer bestimmten Klasse gehört oder nicht. Die Step-Funktion ermöglicht es dem Neuron, eine klare Entscheidung zu treffen: Entweder wird das Neuron aktiviert (Ausgabe = 1) oder nicht (Ausgabe = 0). Diese binäre Entscheidung ist besonders nützlich in einfachen Klassifikationsproblemen, bei denen die Daten linear separierbar sind (siehe auch XOR-Problem).

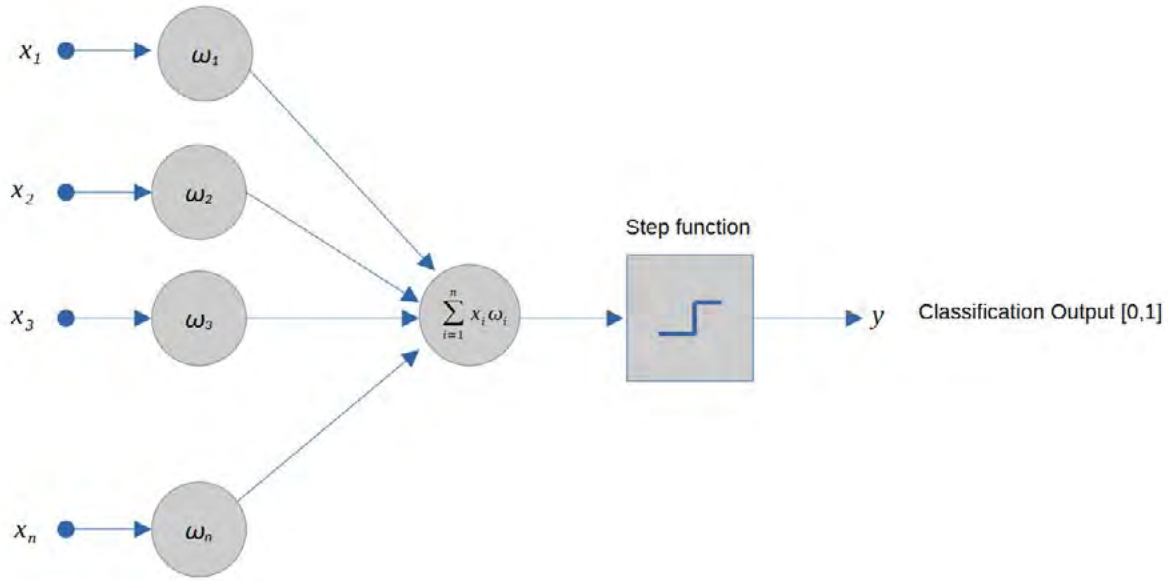

Abbildung 1: Single-Layer-Neuron (Klassifikator)Merkmale (eigene Illustration)

Einfach gesagt funktioniert ein Single-Layer-Neuron, indem es die gewichteten Eingänge summiert, einen Bias hinzufügt und das Ergebnis durch eine Step-Funktion schickt, um eine binäre Ausgabe zu erzeugen. Die Step-Funktion ist entscheidend, um eine klare Entscheidungsgrenze zu schaffen und das Neuron als einfachen Klassifikator zu verwenden.

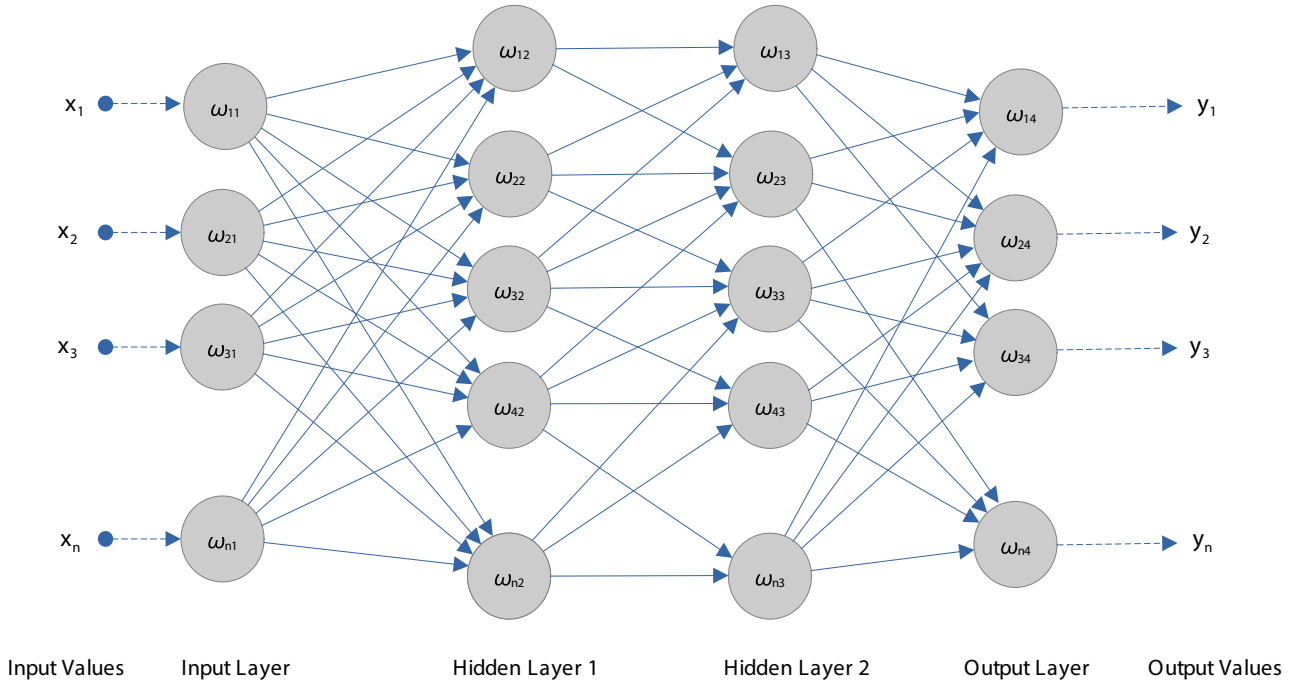

Abbildung 2: Multilayer-Netzwerk, mit zwei Hidden Layers

Ein Multi-Layer-Netzwerk (Abbildung 2) ist eine Erweiterung des Single-Layer-Neurons und besteht aus mindestens drei Schichten: der Eingabeschicht (Input Layer), einer oder mehreren versteckten Schichten (Hidden Layers) und der Ausgabeschicht (Output Layer). Die Eingabeschicht empfängt die Daten, während die versteckten Schichten aus Neuronen bestehen, die jeweils eine gewichtete Summe ihrer Eingänge plus einen Bias berechnen. Diese Summe wird durch eine nichtlineare Aktivierungsfunktion (z. B. ReLU, Sigmoid oder Tanh, nicht in der Grafik gezeigt) transformiert, um komplexe, nichtlineare Beziehungen in den Daten zu modellieren. Die Ausgabeschicht erzeugt die endgültige Vorhersage, wobei die Aktivierungsfunktion hier vom Problemtyp abhängt (z. B. Sigmoid für binäre Klassifikation oder Softmax für Multi-Klassen-Klassifikation).

Während der **Vorwärtspropagation** werden die Daten durch das Netzwerk geleitet, und die Ausgabe wird berechnet. Anschliessend wird der Fehler zwischen Vorhersage und Zielwert mittels einer Verlustfunktion bestimmt. Dieser Fehler wird während der **Rückwärtspropagation** rückwärts durch das Netzwerk propagiert, um die Gradienten der Gewichte und Biases zu berechnen. Mit diesen Gradienten werden die Parameter des Netzwerks aktualisiert, um den Fehler zu minimieren.

Lernen und Trainieren durch Backpropagation

Backpropagation ist ein zentraler Algorithmus zum Training von Feedforward-Neuronalen Netzen und einer Vielzahl von Netzwerk-Geometrien. Backpropagation wurde erst in den 1970er und 1980er Jahren entwickelt und populär. Der Doktorand Paul Werbos beschrieb das Konzept erstmals 1974 in seiner Doktorarbeit, aber es wurde erst 1986 durch die Arbeit von Rumelhart, Hinton und Williams so richtig bekannt. [17]

Die Methode ermöglicht es im Prinzip, die Gewichtungen und Biases des Netzwerks so anzupassen, dass der Fehler zwischen den vorhergesagten und den tatsächlichen Ausgaben minimiert wird. Der Prozess beginnt mit der Vorwärtsverarbeitung (Feedforward), bei der die Eingabedaten durch das Netzwerk geleitet werden. Jedes Neuron berechnet eine gewichtete Summe seiner Eingaben, addiert einen Bias und wendet eine Aktivierungsfunktion an, um sein Ausgangssignal zu erzeugen. Die Ausgabe des Netzwerks wird dann mit den tatsächlichen Zielwerten verglichen, um den Fehler zu berechnen. Dieser Fehler wird durch eine Verlustfunktion (z. B. mittlerer quadratischer Fehler, auch Loss function) quantifiziert. [10]

Das Ziel von Backpropagation ist es dann, diesen Fehler zu minimieren, indem die Gewichtungen und Biases angepasst werden. Dazu wird der Fehler rückwärts durch das Netzwerk propagiert, um die Gradienten (die Unterschiede) der Verlustfunktion in Bezug auf jede Gewichtung und jeden Bias zu berechnen. Diese Gradienten geben an, wie stark sich der **Fehler ändert**, wenn eine **Gewichtung** oder ein **Bias** leicht angepasst wird.
Die Berechnung dieser Gradienten erfolgt mithilfe der Kettenregel aus der Differentialrechnung. Zunächst wird der Fehler in der Ausgabeschicht berechnet. Dann wird dieser Fehler rückwärts durch die Hidden Layers propagiert, wobei für jedes Neuron der Beitrag zum Fehler bestimmt wird. Dies geschieht durch die Multiplikation des Fehlers mit der Ableitung der Aktivierungsfunktion. Die Gradienten für die Gewichtungen und Biases werden dann aus diesen Fehlerbeiträgen abgeleitet. [10]

Die Methode des Gradientenabstiegs (mathematische Optimierungsmethode) wird dann dazu verwendet, die Gewichtungen und Biases in die Richtung zu aktualisieren, die den Fehler **verringert**. Die Grösse der Anpassungen wird durch die Lernrate bestimmt, einen sog. Hyperparameter, der die Geschwindigkeit des Lernprozesses steuert.

Backpropagation ist ein iterativer Prozess, der über viele Epochen (Durchläufe des gesamten Datensatzes) wiederholt wird, bis der Fehler ausreichend klein ist und/oder das Netzwerk gut generalisiert. Ein wichtiger Aspekt ist die Initialisierung der Gewichtungen, die zufällig, aber in einem bestimmten Bereich erfolgen muss, um Probleme wie das Verschwinden oder «Explodieren» von Gradienten zu vermeiden.

Backpropagation revolutionierte das Training von Multi-Layer-Netzwerken, da es komplexe, nichtlineare Modelle praktisch trainierbar machte. Ohne Backpropagation war das Training von Deep-Neuronal-Netzen zu rechenintensiv und ineffizient, weshalb es zu einem zentralen Werkzeug im Bereich des Deep Learning wurde. Man kann sagen, ohne diese Methode könnten neuronale Netze nicht so gross werden, wie wir sie heute z. B. bei Large Langauge Modellen kennen (ChatGPT) [8].

Verschiedene Architekturen von Deep-neural-Netzwerken
Multi-Layer-Netzwerke sind deswegen so leistungsstark, weil sie begründet durch das Universal-Approximation-Theorem in der Lage sind, jede kontinuierliche Funktion zu approximieren. Sie lernen hierarchische Merkmale aus den Daten, wobei frühere Schichten einfache Muster (z. B. Kanten in Bildern) und spätere Schichten komplexere Strukturen (z. B. Objekte) erkennen. Diese Flexibilität macht sie geeignet für eine Vielzahl von Aufgaben wie Klassifikation, Regression, Bilderkennung und Sprachverarbeitung, wie wir im Folgenden sehen werden. [2][14][15]

Dabei basiert Deep Learning auf mehreren verschiedenen Hauptarchitekturen, die je nach Anwendungsfall unterschiedliche Stärken aufweisen. Jede Netzwerk-Architektur dient unterschiedlichen Zwecken und ist auf bestimmte Arten von Daten und zu lösenden Aufgaben zugeschnitten. Unter Architektur oder Netztopologie wird hier die Anordnung der Knoten und Verbindungen (Gewichte) verstanden.

Eine der grundlegendsten und ältesten Architekturen sind die sogenannten **Feedforward Neural Networks** (FNN), bei denen die Information nur in eine Richtung fliesst – von den Eingaben zu den Ausgaben. Sie werden hauptsächlich für einfache Klassifizierungs- und Regressionsprobleme verwendet. [2]

Diese Feedforward Neural Networks (auch **Unsupervised Pretrained Networks**) sind so konzipiert, dass sie Repräsentationen von Daten ohne gelabelte (markierte) Beispiele erlernen, was sie besonders nützlich für Aufgaben macht, bei denen gelabelten Daten knapp oder teuer zu beschaffen sind. Beim unüberwachten Vortraining wird das Netzwerk mit unmarkierten Daten trainiert, um zugrundeliegende Muster oder Merkmale zu erfassen, die dann mit markierten Daten für bestimmte Aufgaben fein abgestimmt werden können.

Für sequenzielle Daten wie Sprache oder Zeitreihen (beispielsweise Börsenkurse) werden häufig rekurrente neuronale Netze (RNN) verwendet (Abbildung 3). Diese Netze haben wiederkehrende Verbindungen zwischen den Neuronen, die es ihnen ermöglichen, «frühere» Informationen zu speichern. Diese Netze sind fähig, wiederkehrende oder typische Muster in sequenzielle Daten wie Zeitreihen, Text oder Sprache zu erkennen. Die Reihenfolge der Daten/Eingaben ist dabei wichtig: Vertauschungen, die in Textreihen vorkommen werden, als neue Sequenz betrachtet. Dies macht sie nur teilweise für Sprachsysteme geeignet. [2][8]

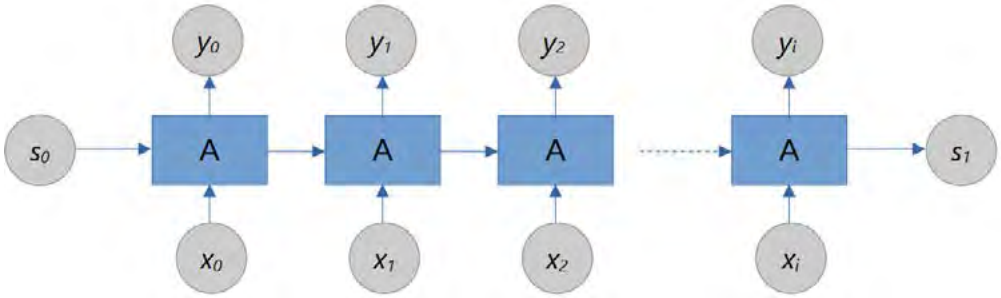

Abbildung 3: Rekurrentes neurales Netzwerk (RNN)

Im Gegensatz zu Feedforward-Netzwerken haben RNNs Verbindungen, die **gerichtete Zyklen** bilden, sodass sie ein «Gedächtnis» für frühere Eingaben ausbilden. Dies macht sie ideal für Aufgaben wie Sprach-modellierung, maschinelle Übersetzung und Spracherkennung.

Ein bekanntes Problem von RNNs ist der **Vanishing-Gradient-Effekt**, weshalb fortgeschrittene Varianten wie Long Sport-Term Memory (LSTM) und Gated Recurrent Unit (GRU) entwickelt wurden, um diese Schwäche zu beheben. Der **Vanishing-Gradient-Effekt** tritt bei diesen Netzen auf, wenn die Gradienten während der Backpropagation immer kleiner werden. Dadurch ändern sich die Gewichte in den vorderen Schichten des Netzes kaum noch, was das Lernen erschwert oder sogar verhindert. [2]

Rekursive neuronale Netze wurden entwickelt, um **hierarchische Strukturen** zu verarbeiten wie z. B. Parse-Bäume in der natürlichen Sprachverarbeitung oder molekulare Strukturen in der Chemie. Im Gegensatz zu RNNs, die Sequenzen linear verarbeiten, arbeiten rekursive Netze mit baumartigen Strukturen, die es ihnen ermöglichen, Beziehungen zwischen Komponenten auf verschiedenen Hierarchieebenen zu erfassen. Abbildung 4 zeigt eine baumartige Struktur, die einen Text prozessiert. Diese Architektur ist besonders nützlich für Aufgaben wie syntaktisches Parsing, Sentimentanalyse und die Modellierung komplexer Beziehungen in strukturierten Daten. [2]

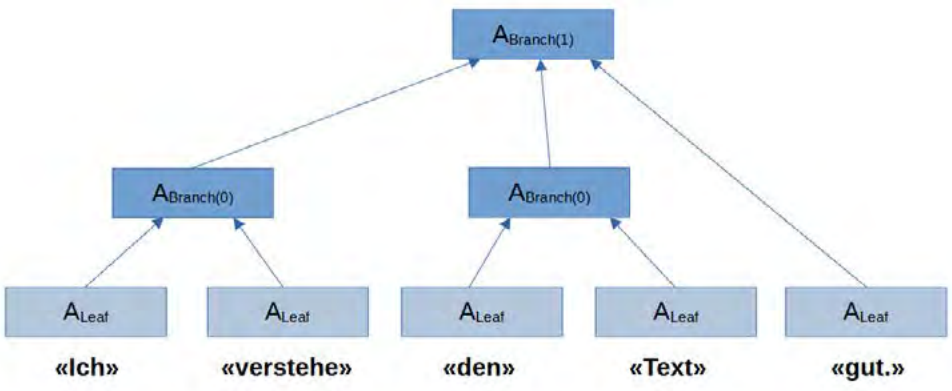

Abbildung 4: Rekursives neurales Netzwerk

Eine weitere wichtige Architektur sind **Convolutional Neural Networks** (CNN), die sich besonders gut für die Verarbeitung von Bildern und Videos eignen. Sie verwenden Matrizen-Kernels, um Muster und Merkmale aus Bilddaten zu extrahieren. Ein Kernel in CNNs ist eine kleine mathematische Matrix, die als «Filter» dient, um bestimmte Merkmale aus den Eingabedaten zu extrahieren. Der Kernel gleitet über das Bild, wobei an jeder Position eine elementweise Multiplikation mit dem entsprechenden Bildausschnitt durchgeführt wird. Die dabei entstehenden Werte werden summiert, um in der resultierenden Feature-Map einen einzelnen Wert zu erzeugen. Durch den Einsatz verschiedener Kernels können unterschiedliche Merkmale wie Kanten, Texturen und Formen erkannt werden. Während des Trainings passen sich die Werte in den Kernels an, um die Erkennung relevanter Muster zu optimieren. Abbildung 5 zeigt eine Klassifikationsaufgabe mittels eines Convolutional Neuronal Networks. Die Kernels sind als keine Quadrate zu sehen, die diese Gebiete für den nächsten Layer zusammenfassen (mitteln). [15]

Ein wesentlicher Fortschritt in der Verarbeitung von Sequenzdaten wurde durch Transformer erreicht. Diese Architektur basiert auf dem Attention-Mechanismus und kann globale Abhängigkeiten in Sequenzen effizient erfassen. Transformer-Modelle haben RNNs im Natural Langauge Processing (NLP) überholt und bilden die Grundlage für moderne Sprachmodelle wie BERT, GPT, T5 und Ollama [8]. Sie werden zunehmend auch in der Bildverarbeitung eingesetzt, z. B. in Vision Transformers. Transformer werden ausführlicher im Kapitel über Generative KI diskutiert.

Ein weiteres Konzept im Deep Learning sind Autoencoder (AE), die unüberwachtes Lernen zur Merkmalsextraktion oder Datenkompression ermöglichen (Abbildung 6). Sie bestehen aus einem Encoder, der die Dimensionen der Daten reduziert, und einem Decoder, der versucht, die ursprüngliche Eingabe wiederherzustellen. Die bekanntesten Varianten sind Variational Auto-Encoders (VAE) und Denoising Auto-Encoders (DAE).

Alle Informationen aus der Eingabeschicht x müssen durch das Nadelöhr in der $P(z|x)$-Schicht. Da die Gewichte bei jedem Durchlauf so angepasst werden, dass die Informationen aus der x Schicht mit der $P(x|z)$ Schicht möglichst übereinstimmen, werden nur die notwendigen Informationen herausgefiltert und durch die Mittelschicht $P(z|x)$ gelassen. In der Quintessenz bedeutet das ein Filterung auf die wichtigen Features beispielsweise eines Bildes (Rand, Form, Farbe).

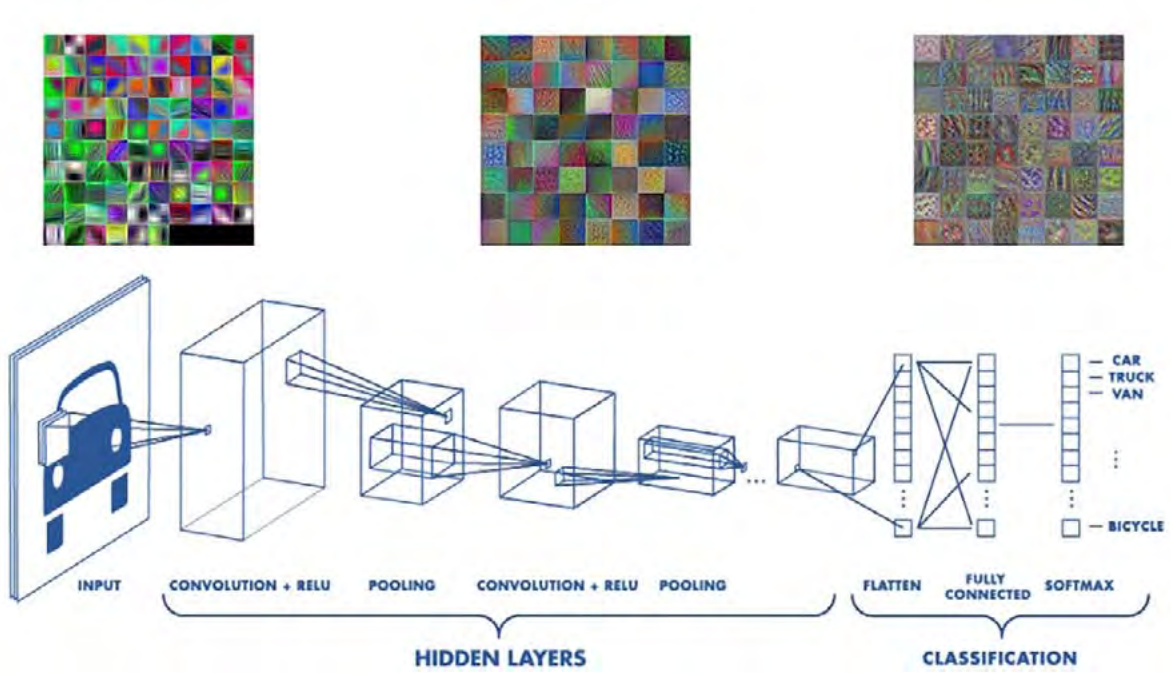

Abbildung 5: Bilderkennung mittels eines Convolutional Neuronal Networks. Die Features werden in den Hidden Layers extrahiert und in den Klassifikationslayern erkannt und einem Ausgangsknoten zugewiesen. (Quelle: [18])

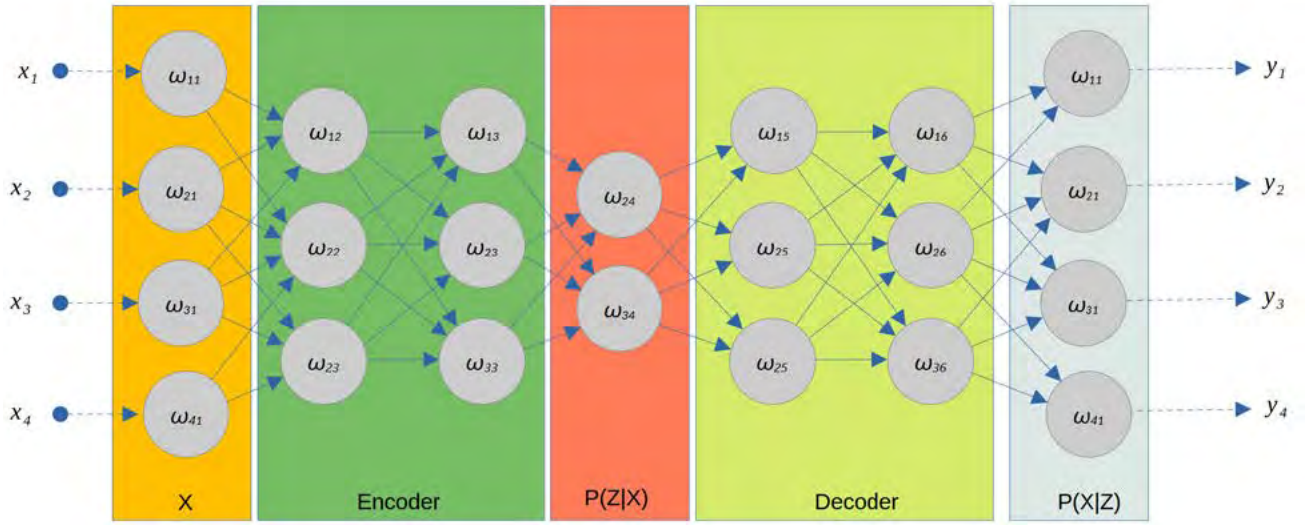

Abbildung 6: Schema eines Autoencoders

Für generative Aufgaben, beispielsweise Bild- oder Filmerzeugung, sind **Generative Adversarial Networks** (GANs) besonders relevant. Diese bestehen aus zwei Netzwerken – einem Generator und einem Diskriminator –, die gegeneinander arbeiten, um realistische Daten zu erzeugen. GANs werden für die Erzeugung von Bildern, Deepfake-Technologien und Stiltransfers verwendet. Bekannte Modelle dieser Kategorie sind DCGAN, StyleGAN und BigGAN.

Schliesslich gibt es noch **Graph Neural Networks** (GNN), die für Daten entwickelt wurden, die in Form von Graphen vorliegen, wie z. B. soziale Netzwerke oder Molekülstrukturen. Sie erweitern neuronale Netze auf nicht-euklidische Datenstrukturen und ermöglichen eine effiziente Verarbeitung solcher Informationen. Die wichtigsten Varianten sind Graph Convolutional Networks (GCN) und Graph Attention Networks (GAT).

Diese Architekturen werden oft kombiniert oder angepasst, um spezifische Probleme zu lösen und neue Anwendungsgebiete zu erschliessen.

Im nächsten Kapitel gehen wir nun auf die Anwendungsbereiche ein, wo viele der beschriebenen Methoden zum Einsatz kommen.

Deep Learning und KMU

Künstliche Intelligenz (KI) und insbesondere Deep Neural Networks (DNNs) bieten auch kleinen und mittleren Unternehmen (KMU) in der Schweiz enorme Chancen, um wettbewerbsfähig zu bleiben und neue Geschäftsmöglichkeiten zu erschliessen.

Dank der Fortschritte in der Rechenleistung von handelsüblichen Computern, Servern, aber auch Dienstleistungen im Cloud-Computing-Bereich und der leichten Verfügbarkeit von grossen Datenmengen können KMU heute auf leistungsstarke KI-Tools zugreifen, die früher grossen Unternehmen vorbehalten waren.

Deep Neural Networks, wie auch prinzipiell Methoden des maschinellen Lernens, ermöglichen es, komplexe Muster in Daten zu erkennen und Vorhersagen zu treffen, die für die Automatisierung von Prozessen, die Personalisierung von Dienstleistungen oder die Optimierung von Geschäftsentscheidungen genutzt werden können.

In der Schweiz, einem Land mit einer starken Innovationskultur und einer diversifizierten Wirtschaft, können KMU diese Technologien nutzen, um die Effizienz zu steigern, Kosten zu senken und neue Märkte zu erschliessen. Ob im Bereich der Fertigung, des Handels, der Finanzdienstleistungen oder der Gesundheitsversorgung – die noch junge Technologie bieten vielfältige Anwendungsmöglichkeiten, die auch für kleinere Unternehmen erschwinglich sind. Durch die Integration von KI können Schweizer KMU nicht nur ihre operative Exzellenz verbessern, sondern auch innovative Produkte und Dienstleistungen entwickeln, die ihnen einen entscheidenden Wettbewerbsvorteil verschaffen.

Anwendungsfelder von Deep Learning für KMU

Um sich den Anwendungsfelder für KMU zu nähern, müssen wir zuerst unabhängig von den Branchensektoren die heute schon möglichen Anwendungsfelder betrachten. Im Folgenden seien in einer nicht abschliessenden Aufzählung mögliche Einsatzgebiete für KMU erörtert.

Predictive Maintenance (Vorausschauende Wartung)

Predictive Maintenance nutzt die Deep-Learning-Technologe insbesondere für Zeitreihenanalysen, um den Zustand von Maschinen und Anlagen in Echtzeit zu überwachen und Ausfälle vorherzusagen. Sensordaten werden analysiert, um Muster zu erkennen, die auf bevorstehende Probleme hinweisen. Dies ermöglicht es, Wartungsarbeiten rechtzeitig durchzuführen, bevor es zu teuren Ausfällen kommt. Die Technologie ist besonders nützlich in der Fertigungsindustrie, der Logistik und der Energieerzeugung. KMU können dadurch Stillstandszeiten reduzieren, die Lebensdauer ihrer Anlagen verlängern und dadurch die Produktivität steigern. Zudem sinken die Kosten für ungeplante Reparaturen und Ersatzteile. Unternehmen, die Predictive Maintenance einsetzen, profitieren von einer höheren Betriebssicherheit und einer besseren Planbarkeit ihrer Ressourcen. Anzufügen ist vielleicht noch, dass Predictive Maintenance auch mit bekannten statistischen Methoden angewendet werden kann. Deep Learning erschliesst aber einen viel breitere und vor allem automatischere Anwendung.

Kundensegmentierung und Personalisierung

Grosse Mengen an Kundendaten zu analysieren und Muster zu identifizieren, die für die Segmentierung und Personalisierung genutzt werden können, ist eine der prominentesten Anwendungen von Deep Learning. Unternehmen können so gezielte Marketingkampagnen erstellen, die auf die Bedürfnisse einzelner Kundengruppen zugeschnitten sind. Diese Technologie ist besonders wertvoll im Einzelhandel, im E-Commerce und im Dienstleistungssektor. Durch personalisierte Angebote und Empfehlungen steigen die Kundenzufriedenheit und die Wahrscheinlichkeit von Wiederholungskäufen. KMU können ihre Marketingeffizienz verbessern und gleichzeitig die Kosten für breit gestreute Werbung reduzieren. Die Analyse von Kaufverhalten und Präferenzen hilft auch bei der Entwicklung neuer Produkte und Dienstleistungen.

Chatbots und virtuelle Assistenten

Chatbots, die im Kern auf Deep-Neuronal-Netzen basieren, können natürliche Sprache verstehen und auf Kundenanfragen in Echtzeit reagieren. Sie eignen sich seit ein paar Jahren auch für den direkten Kundenservice, um häufig gestellte Fragen zu beantworten, Bestellungen aufzunehmen oder Termine zu vereinbaren. Branchen wie Banken, Versicherungen, Reisebüros und der Einzelhandel profitieren besonders von dieser Technologie. Chatbots bieten eine 24/7-Verfügbarkeit und entlasten das Personal von repetitiven Aufgaben. KMU können so die Kundenzufriedenheit steigern und gleichzeitig die Betriebskosten senken. Darüber hinaus können Chatbots Daten aus Kundengesprächen sammeln und analysieren, um wertvolle Einblicke in die Bedürfnisse der Kund:innen zu gewinnen.

Bilderkennung und Qualitätskontrolle

Deep-Learning-Modelle können Bilder analysieren, um Defekte oder Abweichungen in Produkten zu erkennen. Dies ist besonders nützlich in der Fertigungsindustrie, der Lebensmittelproduktion und der Pharmazie. Automatisierte Qualitätskontrollsysteme können Produkte in Echtzeit prüfen und fehlerhafte Teile aussortieren. KMU profitieren von einer höheren Produktqualität und einer Reduzierung manueller Inspektionen, die oft fehleranfällig und zeitaufwendig sind. Die Technologie ermöglicht es auch, Produktionsprozesse kontinuierlich zu optimieren und Ausschuss zu minimieren. Durch den Einsatz von Bilderkennung können Unternehmen ihre Wettbewerbsfähigkeit steigern und die Kundenzufriedenheit erhöhen.

Betrugserkennung

Maschine Learning und insbesondere Deep-Learning-Technologien werden heute schon verwendet, um ungewöhnliche Muster in Transaktionen oder Aktivitäten zu erkennen, die auf Betrug hindeuten. Dies ist besonders relevant für Banken, Versicherungen und E-Commerce-Plattformen. Die Technologie analysiert grosse Datenmengen in Echtzeit und identifiziert verdächtige Aktivitäten, wie z. B. ungewöhnliche Zahlungsströme oder Anmeldeversuche. KMU können so finanzielle Verluste vermeiden und das Vertrauen ihrer Kunden stärken. Die automatische Betrugserkennung reduziert auch den manuellen Aufwand für die Überprüfung von Transaktionen und ermöglicht eine schnellere Reaktion auf potenzielle Bedrohungen. Dies trägt zur Sicherheit und Stabilität des Geschäftsbetriebs bei.

Nachfrageprognosen

Deep-Learning-Modelle können historische Verkaufsdaten analysieren, um die zukünftige Nachfrage vorherzusagen. Dies ist besonders nützlich für Unternehmen im Einzelhandel, der Logistik und der Produktion. Durch genaue Prognosen können KMU ihre Lagerbestände optimieren, Überproduktion vermeiden und Lieferengpässe verhindern. Die Technologie hilft auch bei der Planung von Marketing-

kampagnen und der Steuerung von Produktionsprozessen. Unternehmen können so ihre Ressourcen effizienter einsetzen und ihre Rentabilität steigern. Darüber hinaus ermöglicht die Vorhersage von Nachfragespitzen eine bessere Kundenbindung durch rechtzeitige Lieferungen.

Automatisierte Dokumentenverarbeitung

Deep Learning in Form von Large Language Models kann verwendet werden, um Texte, Rechnungen, Verträge und andere Dokumente automatisch zu verarbeiten und zu analysieren. Dies ist besonders nützlich in der Finanzbranche, im Rechtswesen und im Gesundheitswesen. Die Technologie extrahiert relevante Informationen aus Dokumenten und reduziert den manuellen Aufwand für die Dateneingabe. KMU profitieren von einer schnelleren Bearbeitung von Dokumenten und einer Reduzierung von Fehlern. Automatisierte Systeme können auch Verträge auf Risiken oder Rechnungen auf Unstimmigkeiten überprüfen. Dies spart Zeit und Kosten und verbessert die Effizienz administrativer Prozesse.

Personalisiertes Marketing

Deep Learning ermöglicht es, das Verhalten von Kund:innen zu analysieren und massgeschneiderte Marketingbotschaften zu erstellen. Dies ist besonders effektiv im E-Commerce, im Tourismus und im Mediensektor. Unternehmen können personalisierte Empfehlungen, Angebote und Werbekampagnen erstellen, die auf die individuellen Präferenzen der Kunden zugeschnitten sind. Dies erhöht die Conversion-Rate und die Kundenbindung. KMU können ihre Marketingbudgets effizienter einsetzen und gezielter auf potenzielle Kund:innen zugreifen. Die Analyse von Kundenfeedback und Kaufverhalten hilft auch bei der Entwicklung neuer Produkte und Dienstleistungen.

Spracherkennung und Sprachassistenten

Spracherkennungssysteme können Sprachbefehle verstehen und in Geschäftsprozessen nutzen. Dies ist besonders nützlich in der Telekommunikation, im Gesundheitswesen und im Einzelhandel. Sprachassistenten können verwendet werden, um Bestellungen aufzunehmen, Termine zu vereinbaren oder Informationen abzurufen. KMU profitieren von einer verbesserten Benutzerfreundlichkeit und einer Beschleunigung von Arbeitsabläufen. Die Technologie ermöglicht es auch, barrierefreie Lösungen für Kund:innen mit besonderen Bedürfnissen anzubieten. Sprachassistenten können zudem Daten aus Gesprächen sammeln und analysieren, um wertvolle Einblicke in die Bedürfnisse der Kund:innen zu gewinnen.

Energieoptimierung

Deep Learning kann verwendet werden, um den Energieverbrauch in Gebäuden oder Produktionsanlagen zu analysieren und zu optimieren. Dies ist besonders relevant für Unternehmen in der Fertigungsindustrie, der Immobilienbranche und der Energieerzeugung. Die Technologie identifiziert Einsparpotenziale und ermöglicht eine effizientere Nutzung von Ressourcen. KMU können so ihre Energiekosten senken und ihre Nachhaltigkeitsziele erreichen. Die Analyse von Energieverbrauchsmustern hilft auch bei der Planung von Wartungsarbeiten und der Vermeidung von Energieverschwendung. Dies trägt zur Reduzierung des Umweltfussabdrucks bei und stärkt das Image des Unternehmens.

Im Folgenden wollen wir nun konkret am Beispiel von einigen zufällig ausgewählten Schweizer KMU mögliche Einsatzgebiete anschauen:

Man kann sagen, Deep Learning bietet Schweizer KMU in nahezu allen Branchen enorme Chancen, um Effizienz zu steigern, Kosten zu senken und innovative Lösungen zu entwickeln. In der Fertigungsindustrie, beispielsweise bei Unternehmen wie Bühler, Georg Fischer und Stadler, kann Predictive Maintenance eingesetzt werden, um Maschinenausfälle zu vermeiden, oder für Bilderkennung, um die Qualitätskontrolle zu automatisieren. Dies führt zu höherer Produktivität und reduzierten Stillstandzeiten.

Im Einzelhandel und E-Commerce, etwa bei Digitec Galaxus AG oder lokalen Modehändlern, ermöglichen personalisiertes Marketing und Nachfrageprognosen eine gezielte Kundenansprache und optimierte Lagerbestände, während Chatbots den Kundenservice effizienter gestalten.

In der Finanzdienstleistungsbranche, beispielsweise den Kantonalbanken oder bei Fintech-Startups wie Numbrs, können Deep-Learning-Anwendungen wie Betrugserkennung und automatisierte Dokumentenverarbeitung die Sicherheit und Effizienz von Transaktionen erhöhen.

Für die Logistik- und Transportbranche, etwa bei Kühne + Nagel oder regionalen Transportunternehmen wie Planzer und Galliker, bieten Routenoptimierung und Predictive Maintenance für Fahrzeuge die Möglichkeit, Betriebskosten zu senken und Lieferzeiten zu verbessern. Im Tourismus und Gastgewerbe, beispielsweise bei kleinen Hotels, aber auch bei grösseren Firmen wie Hotelplan, können personalisierte Angebote, Chatbots für Buchungen und Spracherkennung für mehrsprachigen Service die Gästebetreuung verbessern und die Buchungsraten erhöhen.

Im Medien- und Verlagswesen, etwa bei Verlagen wie Tamedia oder Ringier Axel Springer, ermöglicht Deep Learning die automatisierte Inhaltsanalyse und personalisierte Nachrichtenempfehlungen, was die Leserbindung steigert und die Content-Erstellung effizienter macht. In Kreativwirtschaft und Design, beispielsweise bei kleinen Werbeagenturen, unterstützt Bilderkennung Designprozesse, und personalisierte Marketingkampagnen ermöglichen eine gezieltere Kundenansprache.

Fazit

Deep Learning bietet KMU in der Schweiz vielfältige Möglichkeiten, um die Effizienz zu steigern, Kosten zu senken und innovative Lösungen zu entwickeln. Durch den Einsatz dieser Technologien können auch kleinere Unternehmen wettbewerbsfähig bleiben, sich in einem zunehmend digitalen Marktumfeld behaupten und sich als Vorreiter der digitalen Transformation positionieren.

Die oben aufgeführten Anwendungen der Technologie bieten KMU erhebliche Chancen, sich auch im internationalen Wettbewerb zu behaupten. Schweizer Unternehmen profitieren dabei von der hohen Qualität ihrer Produkte und Dienstleistungen, die durch Deep Learning weiter optimiert werden kann.

Die Deep-Learning-Technologie hat erst begonnen ihre Anwendung bei den KMU zu finden. Wer diese historische Gelegenheit als Unternehmen am Schopf packt, wird auch zukünftig im harten Wettbewerb bestehen können.

[1] W. McCullough und W. Pitts, “A logical calculus of the ideas immanent in nervous activity”, *The Bulletin of Mathematical Biophysics*, Bd. 5, Nr. 4, S. 115–133, Dez. 1943.

[2] M. A. Wani, F. A. Bhat, S. Afzal, und A. I. Khan, *Advances in deep learning*. Springer, 2020.

[3] D. Meyer, T. Bonhoeffer, und V. Scheuss, “Balance and Stability of Synaptic Structures during Synaptic Plasticity”, *Neuron*, Bd. 82, Nr. 2, S. 430–443, Apr. 2014, doi: doi.org/10.1016/j.neuron.2014.02.031.

[4] R. G. Morris, “D.O. Hebb: The Organization of Behavior, Wiley: New York; 1949”, *Brain Res Bull*, Bd. 50, Nr. 5–6, S. 437, 1999, doi: doi.org/10.1016/s0361-9230(99)00182-3.

[5] C. M. Bishop und H. Bishop, *Deep Learning – Foundations and Concepts*, 1. Aufl. Calibre, 2023. doi: doi.org/10.1007/978-3-031-45468-4.

[6] S. F. Ahmed u. a., “Deep learning modelling techniques: current progress, applications, advantages, and challenges”, *Artificial Intelligence Review*, Bd. 56, Nr. 11, S. 13521–13617, 2023.

[7] Y. LeCun, L. Bottou, G. B. Orr, und K.-R. Müller, “Efficient backprop”, in *Neural networks: Tricks of the trade*, Springer, 2002, S. 9–50.

[8] M. Treviso *u. a.*, “Efficient Methods for Natural Language Processing: A Survey”. 2023. [Online]. Verfügbar unter: arxiv.org/abs/2209.00099

[9] K. Mehrotra, C. K. Mohan, und S. Ranka, *Elements of artificial neural networks*. Cambridge, MA, USA (Calibre): MIT Press, 1996.

[10] M. A. Wani, F. A. Bhat, S. Afzal, und A. I. Khan, “Introduction to Deep Learning”, in *Advances in Deep Learning*, Singapore: Springer Singapore, 2020, S. 1–11. doi: doi.org/10.1007/978-981-13-6794-6_1.

[11] T. V. Bliss und T. Lomo, “Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path”, *J Physiol*, Bd. 232, Nr. 2, S. 331–356, Juli 1973, doi: doi.org/10.1113/jphysiol.1973.sp010273.

[12] W. Maass, “Networks of spiking neurons: The third generation of neural network models”, *Neural Networks*, Bd. 10, Nr. 9, S. 1659–1671, 1997, doi: doi.org/10.1016/S0893-6080(97)00011-7.

[13] K. Yamazaki, V.-K. Vo-Ho, D. Bulsara, und N. Le, “Spiking Neural Networks and Their Applications: A Review”, *Brain Sciences*, Bd. 12, Nr. 7, S. 863, Juni 2022, doi: doi.org/10.3390/brainsci12070863.

[14] M. A. Wani, F. A. Bhat, S. Afzal, und A. I. Khan, “Supervised Deep Learning Architectures”, in *Advances in Deep Learning*, Singapore: Springer Singapore, 2020, S. 53–75. doi: doi.org/10.1007/978-981-13-6794-6_4.

[15] M. A. Wani, F. A. Bhat, S. Afzal, und A. I. Khan, “Supervised Deep Learning in Face Recognition”, in *Advances in Deep Learning*, Singapore: Springer Singapore, 2020, S. 95–110. doi: doi.org/10.1007/978-981-13-6794-6_6.

[16] J. V. Neumann, “The General and Logical Theory of Automata”, Cerebral mechanisms in behavior; the Hixon Symposium, S. 1–41.

[17] P. J. Werbos, *The roots of backpropagation: From ordered derivatives to neural networks and political forecasting*. Nashville, TN: John Wiley & Sons, 1994.

[18] Bild: medium.com/@krushnakr9/deep-learning-convolution-neural-network-69fc0a588507