Grundlagen

Joern Ploennigs · AI4SC

> Success in creating AI would be the biggest event in human history. Unfortunately, it might also be the last.
>
> — Stephen Hawking

Machine Learning

Maschinelles Lernen (ML) ist ein Teilbereich der Künstlichen Intelligenz, der sich auf die Entwicklung von Algorithmen und Techniken konzentriert, die es Computern ermöglichen, aus Erfahrung zu lernen, ohne explizit programmiert zu werden. ML-Algorithmen analysieren Daten, erkennen Muster und treffen aufgrund dieser Muster Vorhersagen oder treffen Entscheidungen. Während KI ein breiteres Konzept ist, das sich auf die Idee von künstlicher Intelligenz im Allgemeinen bezieht, konzentriert sich maschinelles Lernen speziell darauf, wie Computer lernen können, aus Daten zu lernen und Probleme zu lösen. Wir definieren es als:

📘 Definition: Maschinelles-Lernen

Maschinelles Lernen (ML) ist ein Teilgebiet der Künstlichen Intelligenz, dass sich mit der Entwicklung und Untersuchung statistischer Algorithmen befasst, die aus Daten lernen und auf ungesehene Daten verallgemeinern und so Aufgaben ohne explizite Anweisungen ausführen können.

Künstliche Intelligenz (KI) ist ein übergeordneter Begriff, der sich auf das Streben bezieht, computergestützte Systeme zu entwickeln, die in der Lage sind, Aufgaben zu erfüllen, die normalerweise menschliche Intelligenz erfordern. KI umfasst eine Vielzahl von Techniken, darunter maschinelles Lernen (ML), aber auch andere Ansätze wie Expertensysteme, Logikprogrammierung und neuronale Netze. Der Begriff KI bezieht sich auf das allgemeine Konzept von computergestützter Intelligenz, während maschinelles Lernen eine spezifische Methode innerhalb des KI-Feldes ist. Wir definieren es wie folgt:

📘 Definition: Künstliche-Intelligenz

Künstliche Intelligenz (KI) ist ein Forschungsgebiet in der Informatik, das Methoden und Software entwickelt und untersucht, die es Maschinen ermöglichen, ihre Umgebung wahrzunehmen, zu verstehen, zu lernen und daraus Handlungen oder Entscheidungen abzuleiten, um definierte Ziele zu erreichen, die traditionell menschliche Intelligenz erfordern. Solche Maschinen werden als KIs bezeichnet.

Vorgehen

Die meisten Machine-Learning-Projekte laufen nach dem gleichen Vorgehensschema ab. Hierbei wird nicht strikt sequentiell gearbeitet, sondern meist Agil, wobei die Modelle und die zugrundeliegenden Daten iterativ verbessert werden. Dadurch können insbesondere die ersten Schritte, wie das Data Wrangling, einen sehr großen Teil des Zeitaufwandes ausmachen.

Abbildung 1: Typische Vorgehensweise im Maschinellem Lernen

Datenexploration (Data Exploration): In diesem Schritt werden die Daten eingehend untersucht, um ein besseres Verständnis für ihre Struktur, Eigenschaften und Muster zu erhalten. Häufige Aufgaben umfassen die Analyse von Verteilungen, Korrelationen zwischen Variablen, das Identifizieren von Ausreißern und das Visualisieren der Daten mittels Diagramme oder Grafiken. Ziel ist es, Einblicke in die Daten zu gewinnen, die für die Modellierung relevant sind, sowie potenzielle Probleme oder Herausforderungen zu identifizieren, die angegangen werden müssen.

Datenbereinigung (Data Wrangling): In diesem Schritt werden die Daten bereinigt und vorverarbeitet, um sicherzustellen, dass sie für das Modelltraining geeignet sind. Dies beinhaltet Aufgaben wie das Entfernen fehlender oder unvollständiger Daten, das Behandeln von Ausreißern, das Skalieren von Merkmalen und das Codieren kategorialer Variablen. Ziel ist es, qualitativ hochwertige Daten bereitzustellen, die frei von Störungen oder Verzerrungen sind und eine reibungslose Modellierung ermöglichen.

Modell- und Merkmalsauswahl (Model and Feature Selection): In diesem Schritt werden geeignete Modelle ausgewählt, die für das spezifische Problem am besten geeignet sind, sowie relevante Merkmale, die zur Vorhersage beitragen. Dies kann durch die Bewertung verschiedener Modelle und Merkmalskombinationen anhand von Leistungsmetriken oder durch den Einsatz von Techniken wie Feature Importance oder Cross-Validation erfolgen. Ziel ist es, das beste Modell und die am besten geeignete Merkmale auszuwählen, um genaue und robuste Vorhersagen zu ermöglichen.

Modelltraining (Model Training): In diesem Schritt wird das ausgewählte Modell auf den Trainingsdaten trainiert, um die Beziehung zwischen den Eingangsmerkmalen und den Zielvariablen zu lernen. Das Modell wird durch Anpassen seiner Parameter an die Trainingsdaten verbessert, wobei ein Optimierungsalgorithmus wie Gradientenabstieg verwendet wird. Ziel ist es, ein Modell zu entwickeln, das die Trainingsdaten gut generalisiert und in der Lage ist, genaue Vorhersagen für neue, unbekannte Daten zu treffen.

Modellvalidierung (Model Validation): Nach dem Training wird das Modell auf Validierungs- oder Testdaten bewertet, um seine Leistung zu überprüfen und zu beurteilen, wie gut es auf neue Daten generalisiert. Die Vorhersagen des Modells werden mit den tatsächlichen Werten verglichen, um die Genauigkeit und Zuverlässigkeit des Modells zu überprüfen. Ziel ist es, sicherzustellen, dass das Modell robust ist und konsistente Leistung über verschiedene Datensätze und Umgebungen bietet.

Modellanwendung (Model Scoring): Nach erfolgreicher Validierung wird das trainierte Modell in eine Produktionsumgebung implementiert, wo es zur Vorhersage auf Echtzeitdaten angewendet wird. Dies kann die Integration des Modells in Softwareanwendungen, Webdienste oder andere Systeme umfassen. Ziel ist es, das Modell in einer Umgebung bereitzustellen, in der es praktische Anwendungen hat und kontinuierlich genutzt werden kann.

Machine Learning Typen

Die folgenden ML-Typen sind keine strikt getrennten Klassen. Viele moderne Verfahren sind Mischformen oder hängen in ihrer Einordnung davon ab, welches Trainingssignal und welche Daten im konkreten Anwendungsfall verwendet werden.

Supervised Learning (Überwachtes Lernen)

Supervised Learning (überwachtes Lernen) ist eine Klasse von Algorithmen im maschinellen Lernen, bei dem ein Modell anhand von vorgegebenen Beispieldaten trainiert wird. Dafür werden die Beispieldaten mit den gewünschten Zielwerten (Labels) versehen. Ziel besteht darin, dass das Modell eine Zuordnung zwischen Eingangsdaten und Zielwerten herzustellen lernt, um Vorhersagen für neue, unbekannte Daten treffen zu können.

Die Algorithmen eignen sich insbesondere für Daten und Anwendungsfälle, bei denen man die Zielwerte kennt und gelabelte Daten zum Trainieren hat.

Die Abbildung unten illustriert das am Beispiel von Bildern von Orangen und Äpfeln. Damit das Modell die Kategorien lernen kann, müssen wir jedes Bild mit der entsprechenden Kategorie (Orange, Apfel) labeln und können, dann ein Modell trainieren, welches dann für neue Bilder diese Kategorien unterscheiden kann.

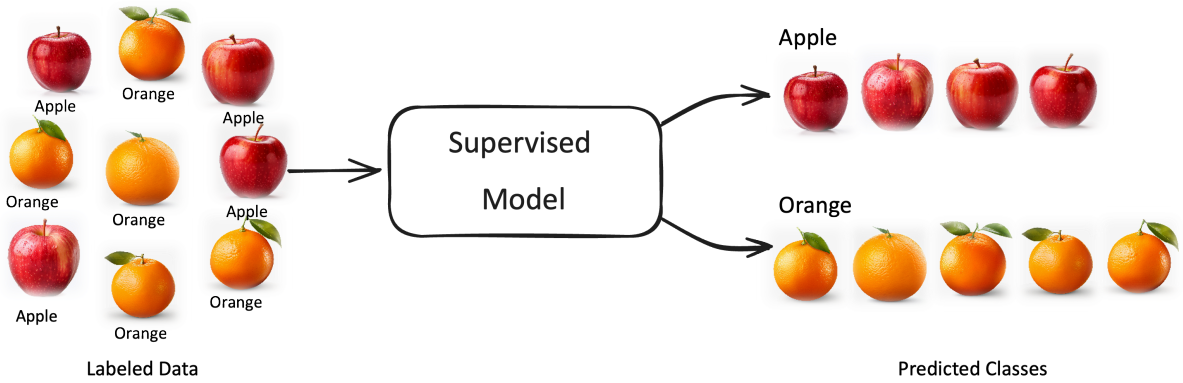

Abbildung 2: Illustration des Supervised Learning (Überwachendes Lernen)

Typische Anwendungen von Supervised Learning umfassen die Bilderkennung, bei der Algorithmen trainiert werden, um Bilder zu klassifizieren oder Objekte zu erkennen, die Sprachverarbeitung, bei der Modelle trainiert werden, um Spracheingaben in Text zu transkribieren oder menschliche Sprachbefehle zu verstehen, und die medizinische Diagnose, wo Modelle verwendet werden, um Krankheiten anhand von Patientendaten zu identifizieren. Supervised Learning ist ein grundlegendes Werkzeug in vielen Bereichen, in denen Vorhersagen oder Klassifikationen basierend auf vorhandenen Daten getroffen werden müssen. Durch die Nutzung von Supervised Learning können komplexe Muster in den Daten erkannt werden, um wertvolle Erkenntnisse zu gewinnen und fundierte Entscheidungen zu treffen.

Im Bauwesen ist eine Anwenungen von Supervised Learning zur Klassifikation z.B. die Erkennung von Rissen in Betonstrukturen anhand von Bilddaten (Riss / Kein Riss). Die Regression wird z.B. verwendet zur Vorhersage der Druckfestigkeit von Beton in Abhängigkeit von Mischung und Aushärtungszeit.

Unsupervised Learning (Unüberwachtes Lernen)

Unsupervised Learning (Unüberwachtes Lernen) ist ein Ansatz im maschinellen Lernen, bei dem ein Algorithmus ein Modell aus Datenmengen lernt, ohne dass diese vorher mit den erwarteten Zielwerten (Labels) versehen wurden. Anders als beim überwachten Lernen gibt es somit keine vorgegebenen korrekten Zielwerte, auf die das Modell trainiert werden soll. Stattdessen versucht das Modell, Muster oder Strukturen in den Daten zu identifizieren, um Einblicke oder Erkenntnisse zu gewinnen. Der Nachteil hierbei ist, dass Modell Strukturen unterscheiden mag, die nicht relevant sind und die resultierenden Kategorien nicht semantisch benannt sind.

Die Algorithmen eignen sich insbesondere für Daten und Anwendungsfälle, bei denen man die Struktur nicht kennt oder gelabelte Daten zum Trainieren fehlen.

Die Abbildung unten illustriert das am vorherigen Beispiel von Orangen und Äpfeln. Anders als beim Supervised Learning haben wir den Bildern keine Label zugeordnet. Das Modell lernt dennoch die Bilder in zwei unbenannte Gruppen (Cluster) zu unterscheiden (anhand Farbe und Form).

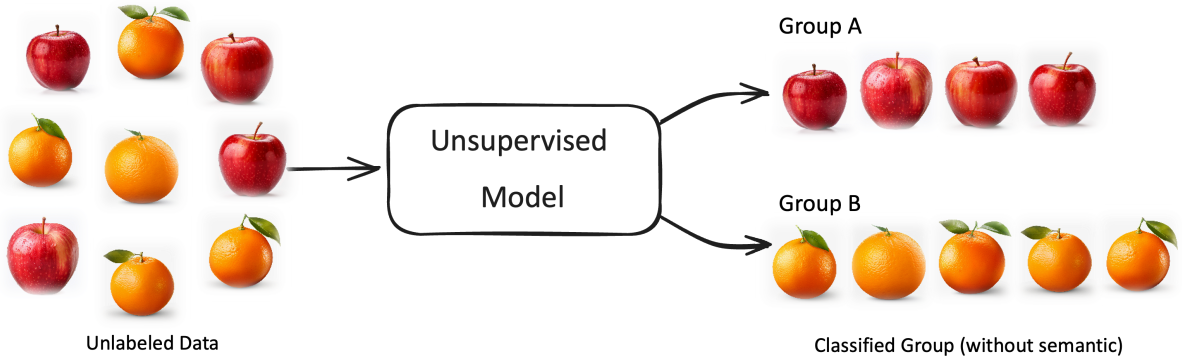

Abbildung 3: Illustration des Unsupervised Learning (Unüberwachendes Lernen)

Typische Anwendungen von Unsupervised Learning umfassen die Clusteranalyse, bei der ähnliche Datenpunkte gruppiert werden, die Dimensionsreduktion, um die Komplexität von Datensätzen zu verringern, und die Anomalieerkennung, um ungewöhnliche Muster oder Ausreißer in den Daten zu identifizieren. Unüberwachtes Lernen wird in einer Vielzahl von Bereichen eingesetzt, darunter Datamining, Bildverarbeitung, Sprachverarbeitung und vieles mehr, wo das Ziel darin besteht, verborgene Strukturen oder Muster in den Daten zu finden, ohne dass vorheriges Wissen über die Ergebnisse vorhanden ist.

Im Bauwesen wird Clustering z.B. zur Gruppierung von Vibrationsdaten von Brücken zur Erkennung ungewöhnlicher Muster oder möglicher Schäden genutzt.

Self-supervised Learning (Selbstüberwachtes Lernen)

Das selbstüberwachte Lernen ist ein relativ neue Gruppe an Modellen, die zwischen dem überwachten und unüberwachten Lernen einzuordnen sind, da sie Ideen aus beiden kombinieren. Beim selbstüberwachten Lernen werden die Trainingsziele automatisch aus den vorhandenen Daten selbst abgeleitet, indem zum Beispiel Daten weggelassen (maskiert) werden, variiert werden, oder durch andere Modelle erzeugt werden. Dadurch lassen sich große Mengen ungelabelter Daten zum Vortraining nutzen, bevor ein Modell gegebenenfalls mit gelabelten Daten nachtrainiert wird.

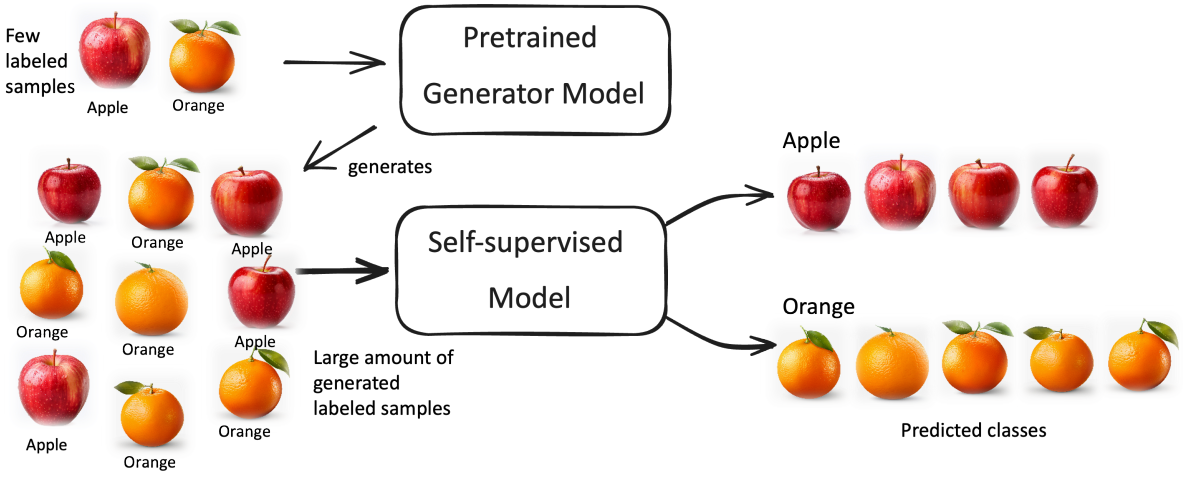

Abbildung 4: Illustration des Selfsupervised Learning (Selbstüberwachendes Lernen)

Selbstüberwachtes Lernen wird heute oft beim Training großer tiefer Neuronaler Netzwerke eingesetzt, die sehr viel Trainingsdaten erfordern. Das umfasst zum Beispiel große Sprach- oder Bildmodelle (LLM/LVM) wie ChatGPT oder Verfahren in der Spracherkennung.

Die Schwierigkeit liegt darin, geeignete Vortrainingsaufgaben zu definieren, aus denen das Modell nützliche Strukturen lernen kann. Dafür werden meist große Mengen ungelabelter Daten und erhebliche Rechenressourcen benötigt. Der Ansatz braucht also nicht wirklich wenig Trainingsdaten, sondern vor allem weniger manuell gelabelte Daten.

Reinforcement Learning (Verstärkendes Lernen)

Reinforcement Learning (Verstärkendes Lernen) ist eine Klasse von Algorithmen des maschinellen Lernens, bei dem ein Model durch Trial-and-Error-Interaktion mit seiner Umgebung lernt, wie es seine Handlungen so anpassen kann, dass er eine langfristige Belohnung maximiert. Im Gegensatz zum überwachten Lernen, bei dem das Model mit gelabelten Daten trainiert wird, oder zum unüberwachten Lernen, bei dem keine Labels vorhanden sind, erhält ein Model im Reinforcement Learning Feedback in Form von Belohnungen oder Bestrafungen für die von ihm getroffenen Handlungen. Das Ziel des Models besteht darin, eine Handlungsstrategie zu entwickeln, die es ihm ermöglicht, über die Zeit hinweg die Gesamtbelohnung zu maximieren.

Die Algorithmen eignen sich insbesondere für komplexe, interaktive Prozesse, bei denen eine eindeutige Lösung nicht kennt (supervised) oder in den Daten identifizieren kann (unsupervised), aber bewerten (und belohnen) kann, welche Entscheidungen tendenziell besser oder schlechter sind.

Ein solches Modell lernt Orangen von Äpfeln zu unterscheiden, indem es eine zweite Bewertungsinstanz um Feedback fragt, ob ein neues Bild zu der Klasse der Äpfel oder Orangen gehört.

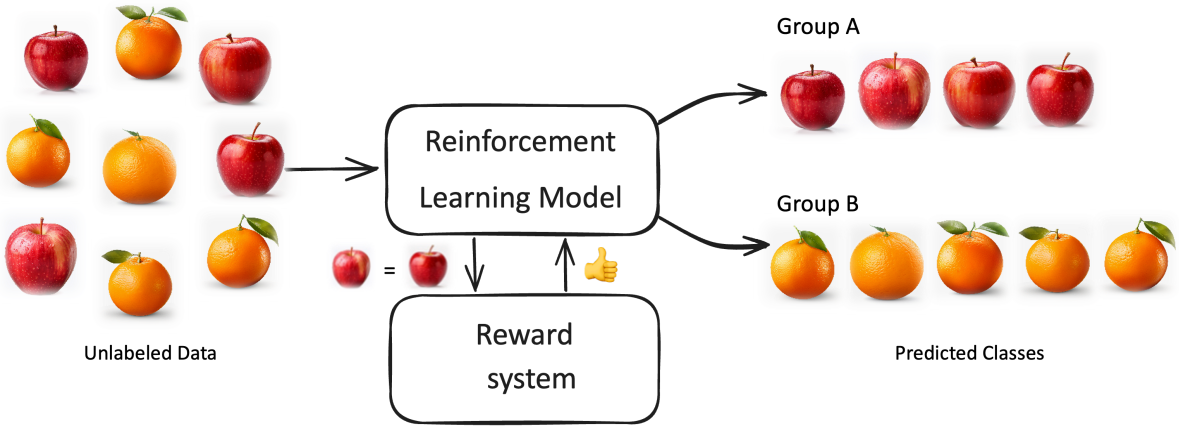

Abbildung 5: Illustration des Reinforcement Learning (Verstärkendes Lernen)

Typische Anwendungen von Reinforcement Learning umfassen die Entwicklung von autonomen Robotern, die lernen, durch physische Umgebungen zu navigieren, das Spielen von Brettspielen wie Schach oder Go sowie die Steuerung von virtuellen Agenten in Videospielen. Reinforcement Learning hat das Potenzial, komplexe Probleme zu lösen, bei denen die richtigen Entscheidungen stark von der aktuellen Situation abhängen und die optimale Strategie durch Erfahrung erlernt werden muss.

Im Bauingenieurwesen kann Reinforcement Learning z.B. zur Optimierung der Fahrwege von autonomen Baumaschinen auf der Baustelle genutzt werden, basierend auf Belohnung für Effizienz und Sicherheit.

Datenklassen

Strukturierte Daten sind Daten, die in einem festgelegten Format vorliegen, das es einfach macht, sie zu organisieren, abzurufen und zu analysieren. Diese Daten bestehen aus klar definierten

Feldern mit spezifischen Datentypen und Relationen zwischen den Feldern. Ein typisches Beispiel für strukturierte Daten sind Tabellen in relationalen Datenbanken, wo jede Zeile einen Datensatz repräsentiert und jede Spalte ein bestimmtes Merkmal darstellt. Beispiele für strukturierte Daten sind Kundendatenbanken, Finanzdaten und Inventarverzeichnisse.

- Tabellen: Diese Daten sind in tabellarischer Form organisiert. (CSV-Dateien, SQL-Datenbanktabellen oder Excel-Tabellen). Typische Beispiele sind: Statistiken, Baustoffkataloge, Projektdaten
- Zeitreihen: Diese Daten sind zeitlich geordnet und bestehen aus einer Abfolge von Zeitpunkten und den jeweiligen Werten. Typische Beispiele sind: Aktienkurse, Wetterdaten, Energiedaten, Sensordaten

Unstrukturierte Daten hingegen sind Informationen, die keinem bestimmten Format oder Schema folgen und daher schwer zu organisieren und zu analysieren sind. Diese Daten können Texte, Bilder, Audio- und Videodateien, E-Mails oder Social-Media-Beiträge umfassen, die in natürlicher Sprache verfasst sind und keine klaren Strukturen aufweisen. Unstrukturierte Daten machen einen Großteil der Daten im Internet aus und stellen eine Herausforderung für die Datenanalyse dar, da sie komplexe Verarbeitungstechniken erfordern, um Bedeutung und Muster zu extrahieren.

- Text: Textdaten sind unstrukturierte, da die Struktur komplex und die Bedeutung sehr Kontextspezifisch ist. Typische Beispiele sind: E-Mails, Dokumente, Internettexte, Bauberichten, Vorschriften
- Bilder: Bilder werden typischerweise in Form von Pixelwerte als mehrdimensionale Matrix gerastert. Typische Beispiele sind: Fotos, Mängelbelege, Diagramme
- Video: Videos werden als Bildsequenz interpretiert. Typische Beispiele sind: Filme, Robotervision, Überwachungskameras
- Audio: Diese Daten enthalten Klangwellenformen und können auch als Zeitreihe gesehen werden. Typische Beispiele sind: Telefonate, Musik, Audioüberwachung, Straßenlärmmonitoring

Semi-strukturierte Daten liegen zwischen strukturierten und unstrukturierten Daten und enthalten teilweise festgelegte Strukturen. Sie können zum Beispiel in Form von JSON- oder XML-Dokumenten vorliegen, die bestimmte Strukturmerkmale aufweisen, aber dennoch Flexibilität bieten, um zusätzliche Informationen hinzuzufügen. Semi-strukturierte Daten können auch in NoSQL-Datenbanken gefunden werden, die es ermöglichen, Daten ohne festes Schema zu speichern. Ein weiteres Beispiel sind E-Mails, die strukturierte Felder wie Absender, Betreff und Datum haben, aber auch unstrukturierte Textkörper enthalten können.

- CAD/BIM: CAD-Daten können als semi-strukturierte Daten betrachtet werden, da die Basisstruktur (z.B. IFC) bekannt ist, aber unstrukturierte Elemente wie Text enthält. Typische Beispiele sind: Bauplänen, Ansichten
- XML/JSON: XML/JSON-Dateien haben zwar eine Objektstruktur, sie ist aber flexibler sind als tabellarische Daten. Typische Beispiele sind: Webseiten in HTML, Internetabfragen

Bibliographie