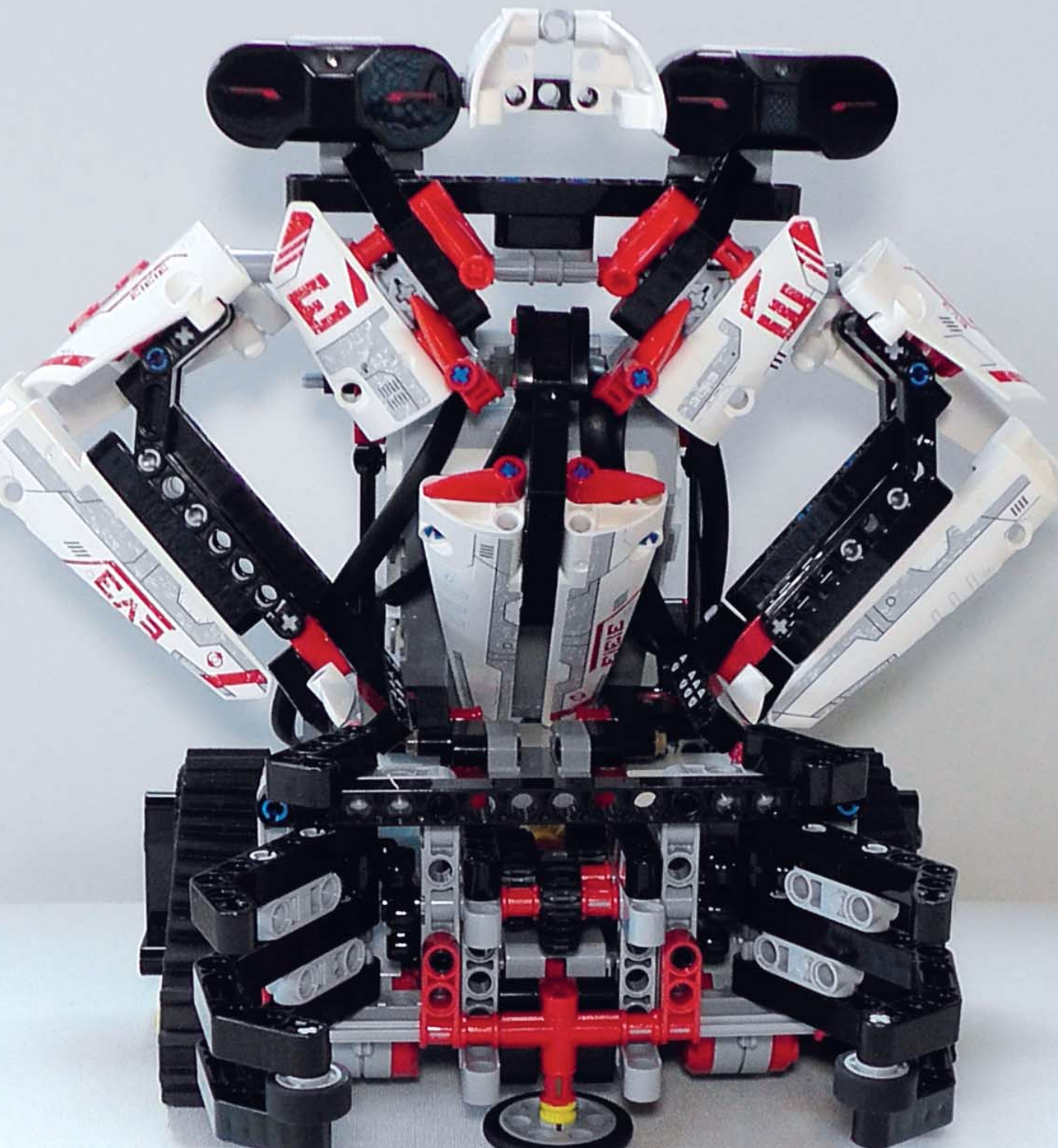
**Laborversuch: Kennenlernen von KI**



Inhaltsverzeichnis

[1 Einleitung 3](#_Toc159411212)

[2 Kontext Künstliche Intelligenz 3](#_Toc159411213)

[3 GesturBot 5](#_Toc159411214)

[3.1 Allgemeine Beschreibung GesturBot 6](#_Toc159411215)

[3.2 Aufbau des Roboters 6](#_Toc159411216)

[3.3 Datensatz erstellen 7](#_Toc159411217)

[3.4 Modell trainieren 7](#_Toc159411218)

[3.5 Modell im Roboter verwenden 7](#_Toc159411219)

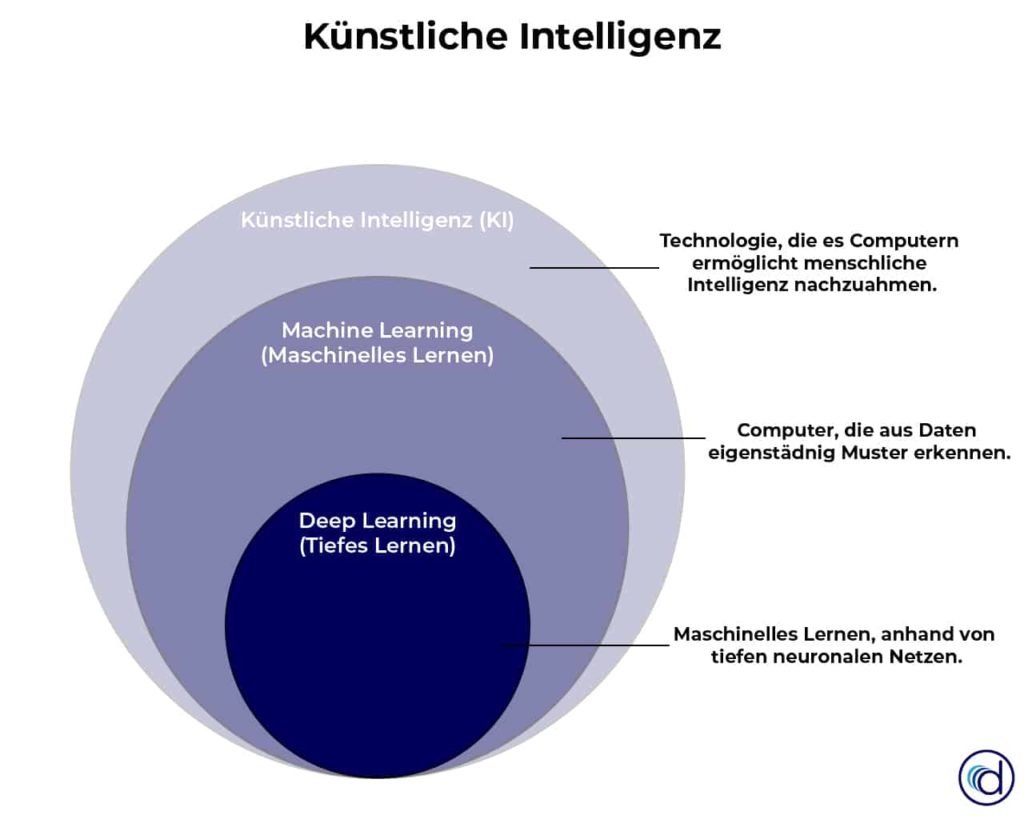
[4 Quellen 7](#_Toc159411220)

# Einleitung

Künstliche Intelligenz spielt in der heutigen Welt eine immer größere Rolle. Sei es im Autonomen Fahren, in Computerspielen oder in der Satzvervollständigung auf WhatsApp - das Anwendungsgebiet von KI wird immer größer und oftmals wird der Einsatz von KI gar nicht bemerkt. Da es für die meisten Personen relativ schwer ist, sich etwas unter dem Begriff Künstliche Intelligenz vorzustellen, wird in diesem Laborpraktikum versucht, erste Einblicke in die Künstliche Intelligenz, insbesondere dem Machine Learning zu geben.

# Kontext Künstliche Intelligenz

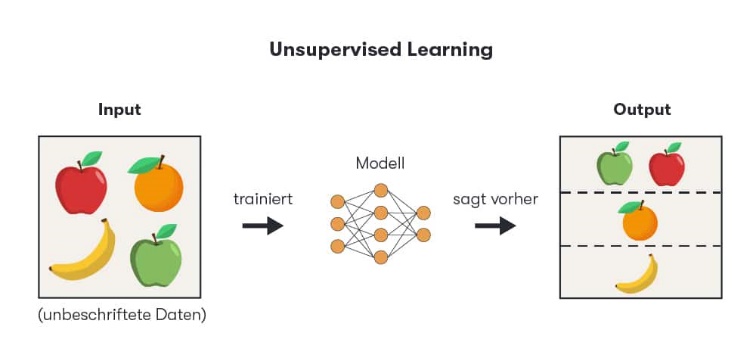
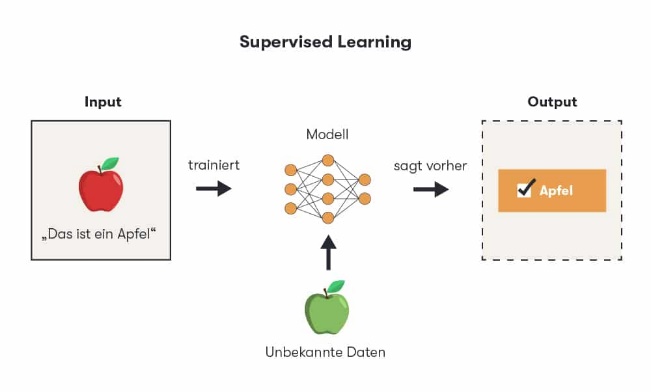
Der Begriff Künstliche Intelligenz – in Englisch Artificial Intelligence, kurz AI - beschreibt den Versuch, menschliches Lernen und Denken auf Computer zu übertragen und ihm so eine gewisse Intelligenz zu verleihen. Genauer gesagt versucht sie Informationen aus Eingabedaten zu erkennen.

Ein Teilgebiet der Künstlichen Intelligenz ist das Machine Learning wie in der unteren Abbildung zu sehen ist. Es beschäftigt sich mit der Entwicklung von Algorithmen und Modellen, die es Computern ermöglichen, aus Daten zu lernen und selbstständig Muster zu erkennen ohne explizit programmiert werden zu müssen. 

Gliederung KI: <https://datasolut.com/was-ist-kuenstliche-intelligenz> (20.02.2024)

In den letzten Jahren machte dieser Zweig besonders große Fortschritte aufgrund von immer höheren Rechenleistungen, mehr Speicherplatz und allgemein die Verfügbarkeit von großen Datenmengen. Diese drei Faktoren sind die Grundvoraussetzungen für Machine Learning.

Das Gebiet Machine Learning lässt sich weiter unterteilen: Es beinhaltet die drei Teilbereiche Supervised Learning, auf Deutsch Überwachtes Lernen, Unsupervised Learning, übersetzt Unüberwachtes Lernen und Reinforcement Learning, nämlich Bestärkendes Lernen. Die Unterschiede der einzelnen Methoden lassen sich in den beiden Abbildungen unten gut sehen:



Unsupervised Learning verwendet nicht gelabelte Daten um Muster und Ähnlichkeiten in Daten zu erfassen und diese in Gruppen zusammenzufassen. Diesen Vorgang nennt man Clustering. Nicht gelabelte Daten bedeutet, dass es keine Ausgabevariablen gibt, die dem Modell während des Training zur Verfügung gestellt werden.

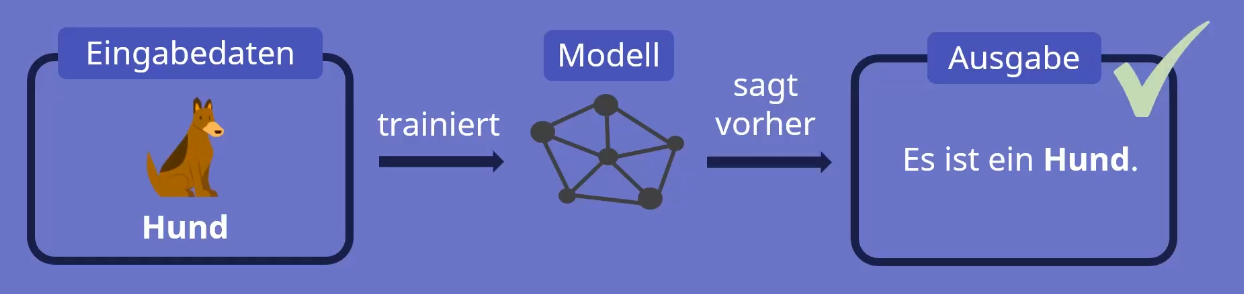
Ein Beispiel des weit verbreiteten Algorithmus kMeans findet sich in untenstehender Abbildung.



kMeans Clustering: <https://www.kaggle.com/code/heeraldedhia/kmeans-clustering-for-customer-data> (20.02.2024)

Reinforcement Learning unterscheidet sich etwas von den beiden anderen Teilgebieten. Es verwendet anders wie die beiden anderen Ansätze keinen statischen Datensatz. Hier soll ein sogenannter Agent, meist eine Software/Algorithmus, durch Interaktion mit einer dynamischen Umgebung lernen, bestimmte Aktionen auszuführen. Der Agent geht dabei nach dem Trial-and-Error Verfahren vor und bekommt Rückmeldung in Form von Belohnungen von der Umgebung, je nachdem wie erfolgreich seine ausgeführte Aktion war. Das Ziel ist die Maximierung der Belohnungen über die Zeit hinweg. Die beiden nachfolgenden Links führen zu zwei Beispielvideos zu Reinforcement Learning: [Hide and Seek](https://www.youtube.com/watch?v=kopoLzvh5jY) oder [Laufen Lernen](https://www.youtube.com/watch?v=L_4BPjLBF4E).

Dieser Versuch beschäftigt sich jedoch mit dem Supervised Learning was besonders bekannt ist durch den Begriff der Neuronalen Netze oder Deep Learning. Das Modell beim Supervised Learning wird dabei mit gelabelten Trainingsdaten trainiert. Es soll die Beziehungen zwischen den Eingabedaten und zugehörigen Ausgabedaten erlernen, um später Vorhersagen auf neue, nicht gelabelte Daten treffen zu können. Diesen Vorgang nennt man Klassifikation.



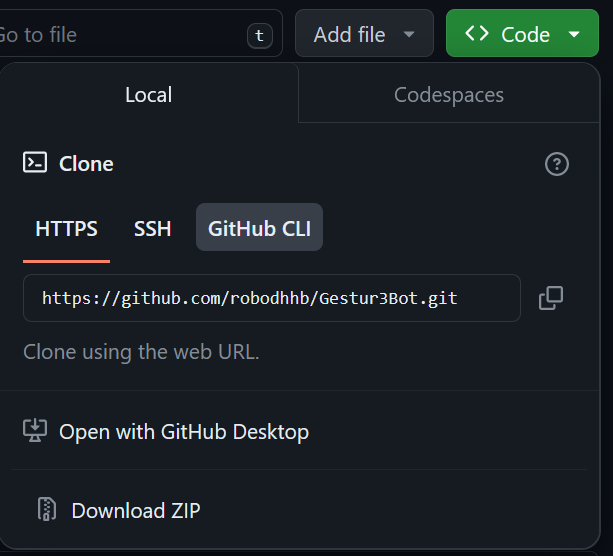
Klassifikation: <https://studyflix.de/informatik/machine-learning-4356> (20.02.2024)

# GesturBot

In diesem Versuch soll eine leicht modifizierte Version des Lego Mindstorms Gripp3r Roboter durch Gesten gesteuert werden können. Die Grundidee dafür kommt von Detlef Heinze, der diesen Versuch mit der internen Entwicklungsumgebung von Lego Mindstorms durchgeführt hat. Dies ist der Link zu seinem GitHub Repository: [Heinze GitHub](https://github.com/robodhhb/Gestur3Bot). Wir verwenden jedoch ausschließlich MATLAB für diesen Versuch, da dies relevanter für das Studium ist und zudem deutlich handlicher als die mitgelieferte Programmierumgebung von Lego.

Hier ist der Link zu unserem GitHub Repository: XXXXXXXXX

Am besten ist es einfach, den Master Branch zu klonen, so befinden sich bereits alle wichtigen Unterlagen wie die .m Files in dem Ordner. Dafür befindet sich rechts oben ein grüner Button mit der Aufschrift *Code*. Diesen einfach drücken und danach „*Download ZIP*“ wählen. Nun müsst ihr nur noch das ZIP-Archiv in einen Ordner eurer Wahl entpacken.

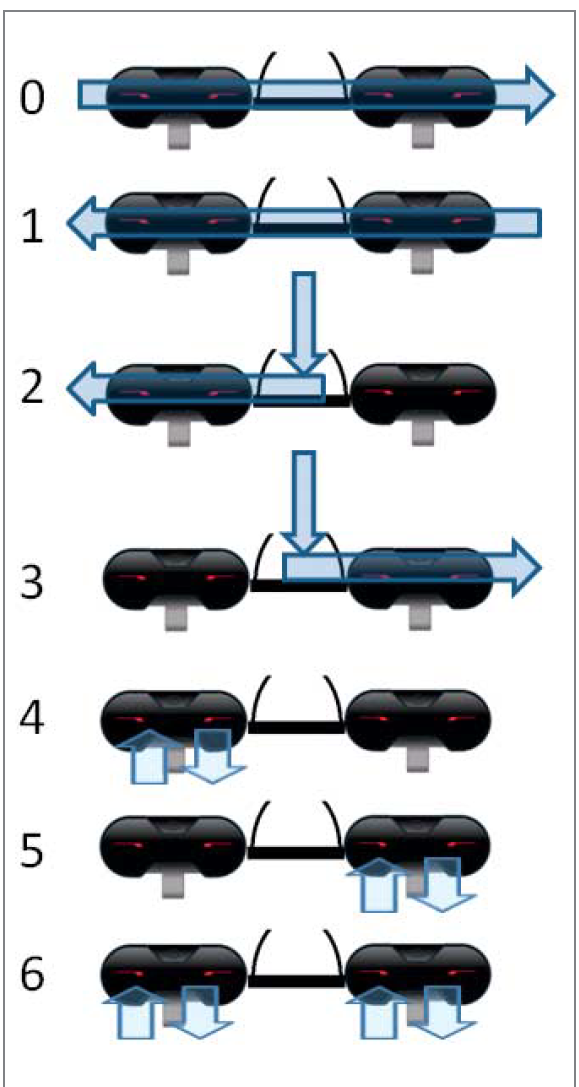


1.

2.

## Allgemeine Beschreibung GesturBot

Ziel des Versuchs ist es, den Roboter mit Handgesten zu steuern. Dabei sind prinzipiell nachfolgende sieben Gesten möglich, wir verwenden jedoch nur drei der Gesten. Einerseits müssen dann nicht so viele Trainingsdaten aufgenommen werden, andererseits ist die Klassifikation zuverlässiger.

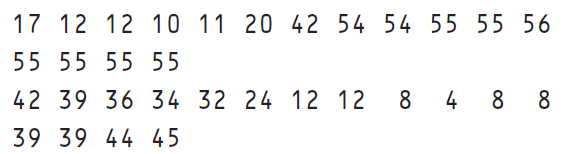


Die 7 Gesten des GesturBots

Wir empfehlen, die Gesten 0, 1 und 6 zu verwenden. Geste 0 wird zum Vorwärtsfahren verwendet, Geste 1 zum Rückwärtsfahren und Geste 6 zum Stoppen. Diese Gesten liefern relativ unterschiedliche Abstandswerte, was die Klassifikation vereinfachen sollte, weil das Modell dann die Muster zwischen Abstand und zusammenhängender Klasse besser versteht. Um die Geste 6 zu erzeugen müssen beide Hände zuerst auf den jeweiligen IR-Sensor zubewegt werden und dann zurückgezogen werden. Entsprechend funktionieren die Gesten 4 und 5, jedoch nur mit einer Hand auf einen der beiden Sensoren.

Für die Erkennung der Gesten sind die beiden IR-Sensoren zuständig. Sie messen den Abstand zwischen einem Objekt, zum Beispiel eurer Hand, und dem Sensor, indem sie die Intensität des reflektierten Infrarotlichts des Objektes messen. Damit dies gut funktioniert sind die Gesten mit der Faust oder mit der flachen Hand auszuführen. Außerdem hat der GesturBot eine Nase zwischen den IR-Sensoren, die verhindern soll, dass zurückgestrahltes Licht des einen Sensors auf den anderen Sensor fällt. Zudem müssen die Gesten mit mittlerer Geschwindigkeit ausgeführt werden damit die IR-Sensoren die Abstandswerte überhaupt detektieren können. Ein Beispielvideo findet sich im Ordner *Lab Description*.

Jede Geste besteht aus einem Vektor mit 32 Sensorwerten. Von jedem Sensor kommen dabei 16 Werte. Im der Abbildung unten findet sich eine Beispielmessung der Geste 0. Die erste Zeile kommt vom linken Sensor, die zweite Reihe vom rechten IR-Sensor, jeweils vom Benutzer aus gesehen.



Beispiel Geste 0

Den Weg der Hand kann man hier sehr gut beobachten: Die Zeit läuft von links nach rechts.

## Aufbau des Roboters

Dieser Abschnitt ist nur relevant falls der Roboter nicht aufgebaut ist. Ansonsten könnt ihr direkt mit Abschnitt 3.2 Datensatz erstellen weitermachen.

Als aller erstes muss der Roboter leicht modifiziert werden, indem ein zweiter IR-Sensor angebracht wird. Hier findet ihr die Bauanleitungen, davon wird die für den Gripp3r-Roboter benötigt: [Anleitungen](https://www.lego.com/de-de/themes/mindstorms/buildarobot). Alternativ haben wir euch die Anleitung auch in unserem GitHub Repository abgelegt.

Mission 1 und Mission 2 sind so durchzuführen wie in der Lego Anleitung beschrieben. Mission 3 wird jetzt bis zu Schritt 36 durchgeführt, danach müsst ihr euch an den drei Bildern im Ordner *Robot Build* orientieren.

Zudem muss der Tastsensor an Port 2 und nicht wie in der Anleitung beschrieben an Port 1 angeschlossen werden.

## Datensatz erstellen

## Modell trainieren

## Modell im Roboter verwenden

# Quellen

<https://www.iks.fraunhofer.de/de/themen/kuenstliche-intelligenz.html#:~:text=K%C3%BCnstliche%20Intelligenz%20(KI)%20ist%20ein,durch%20maschinelles%20Lernen%20erzeugt%20werden>. (20.02.2024)