

HOCHSCHULE OSNABRÜCK
UNIVERSITY OF APPLIED SCIENCES

KI für Alle

Angewandte KI im Hörsaal für Studierende ohne Programmierkenntnisse

Prof. Dr. Julius Schöning

07.06.2024

Hochschule Osnabrück – Fakultät Ingenieurwiss. und Informatik

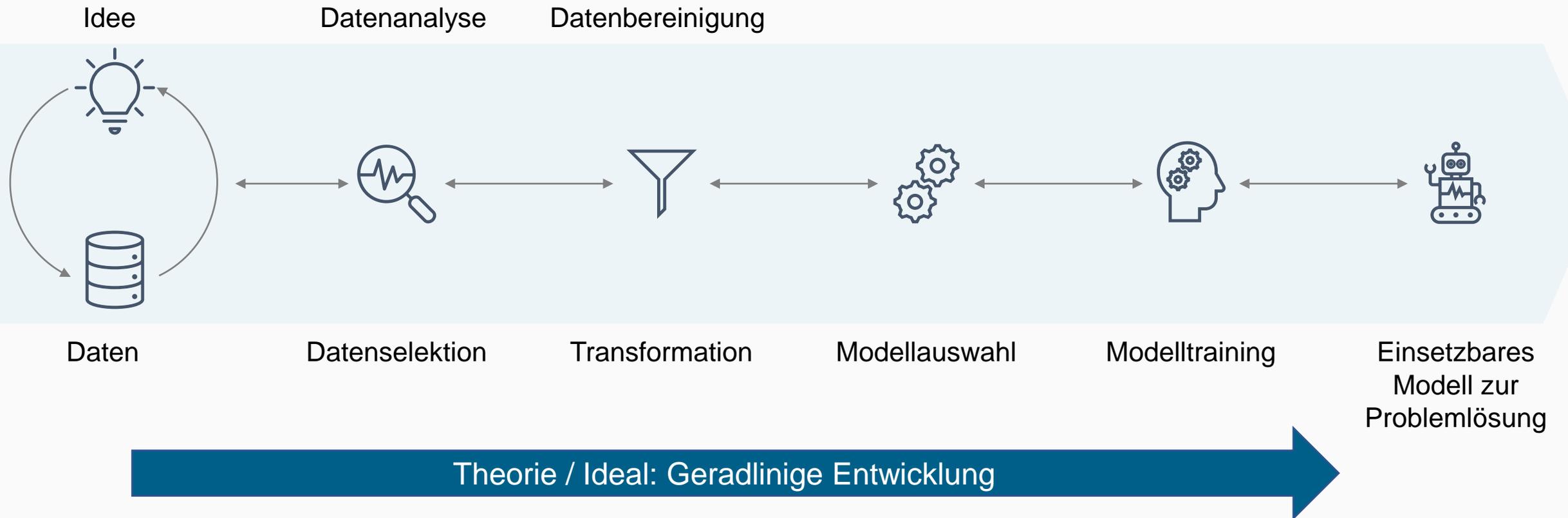
KI für Alle

ChatGPT und die BundeskanzlerInnen

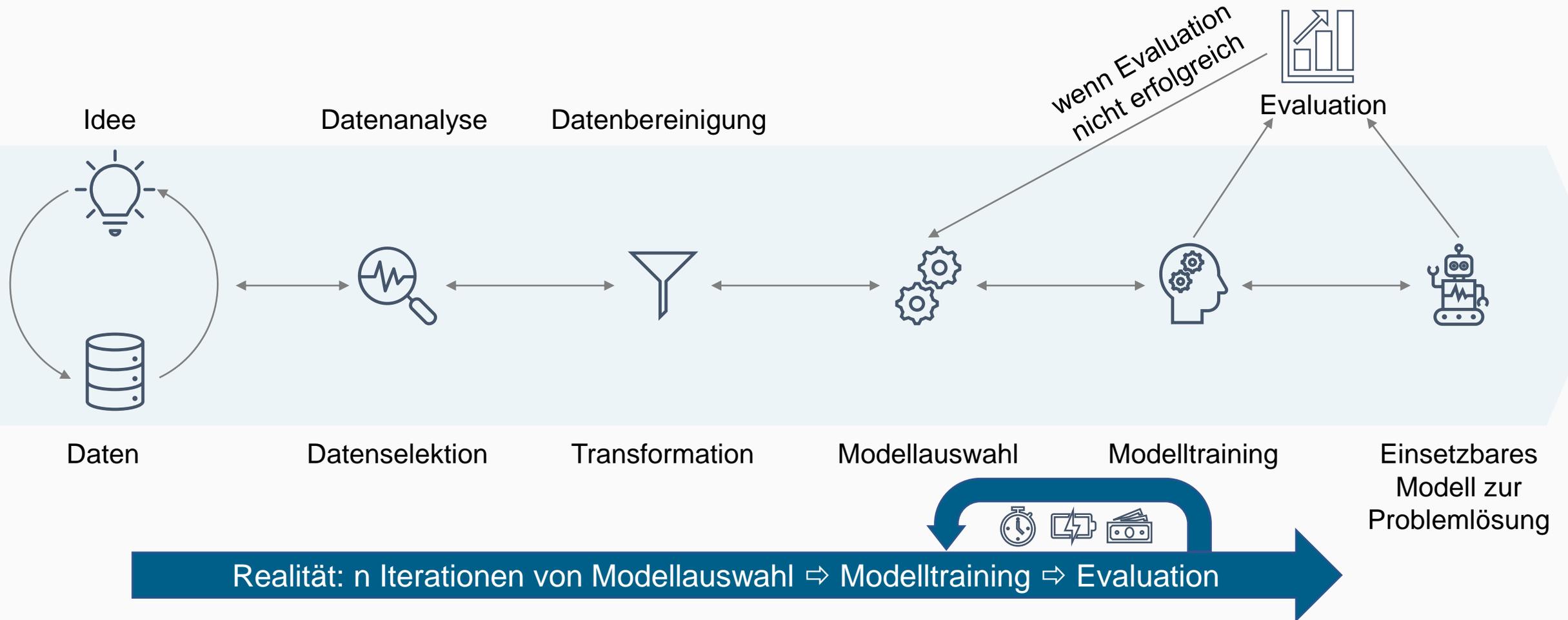
KI für Alle

Angewandte KI in Kürze

Schritte zur erfolgreichen KI-Applikation – Theorie



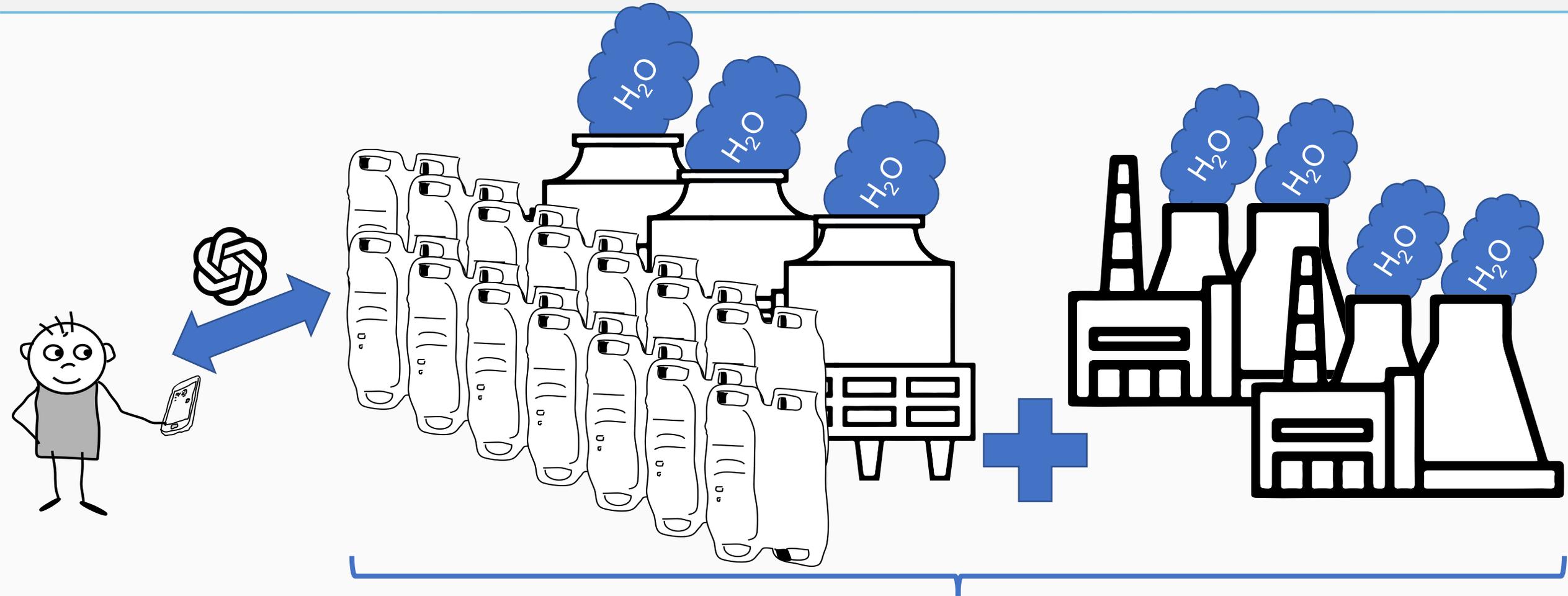
Schritte zur erfolgreichen KI-Applikation – in Realität



KI für Alle

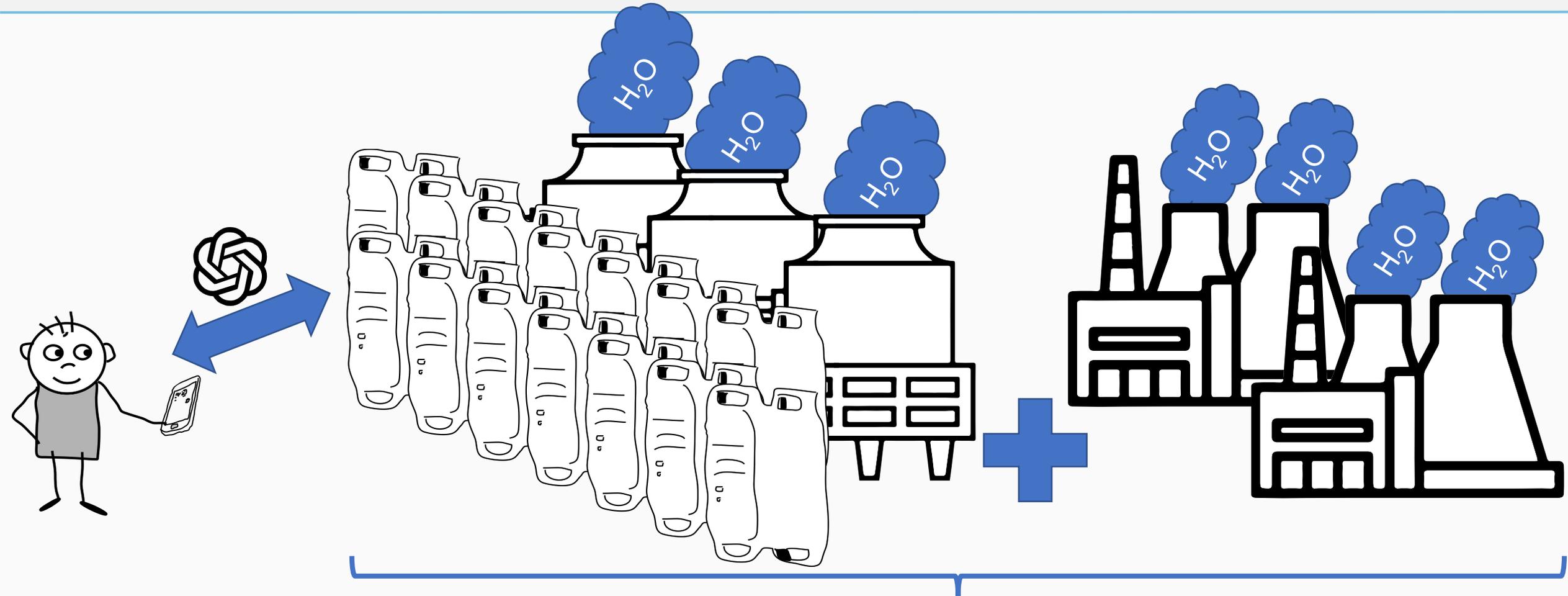
KI und das Problem mit dem „Trinken“

Wasserfußabdruck von GPT-3 beim Training



5,4 Millionen Liter Wasser für das Training von GPT
(zwei olympische Schwimmbecken)

Wasserfußabdruck von GPT-3 bei der Ausführung



16,9 Milliliter Wasser pro Antwort
(ein Schnapsglas Wasser)

KI für Alle

KI ohne Programmierkenntnisse



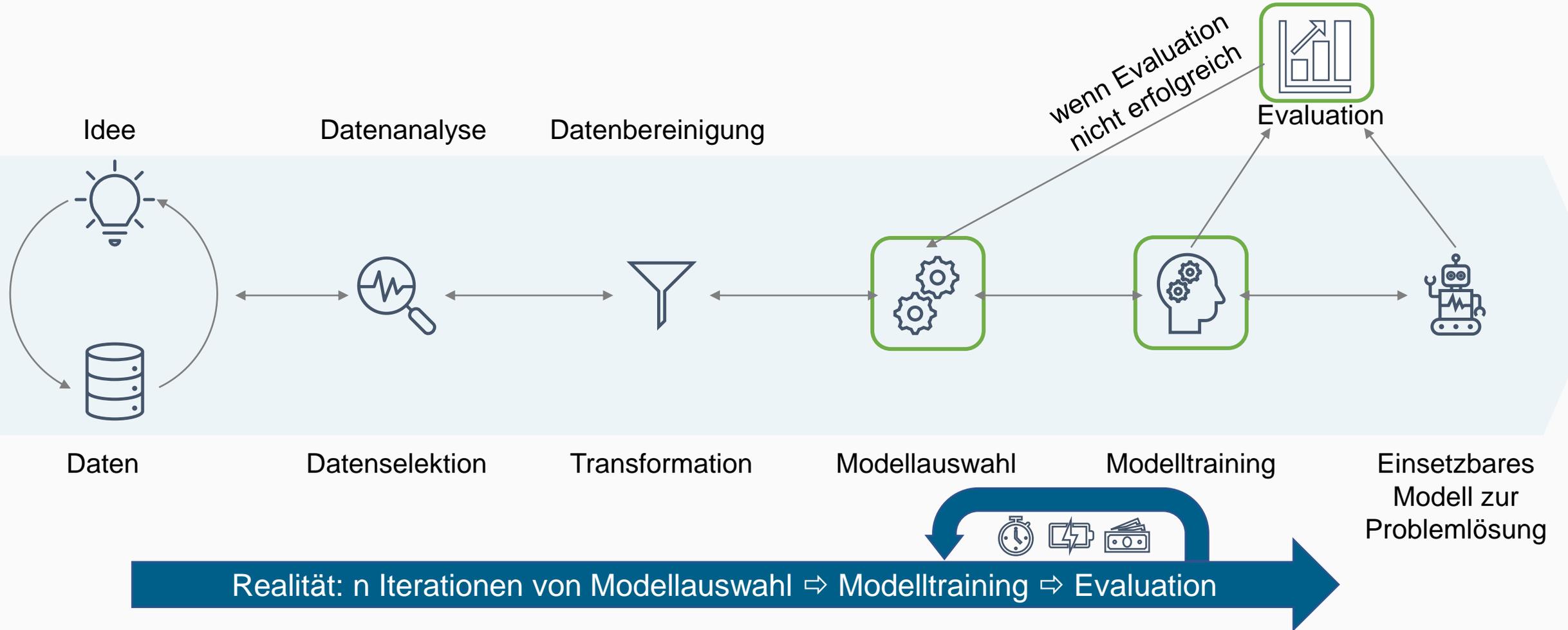
J. Schöning, T. Wawer & K.M. Griese.

*KI für Nicht-Programmierer*innen: Angewandte KI im Hörsaal für Studierende ohne Programmierkenntnisse.*

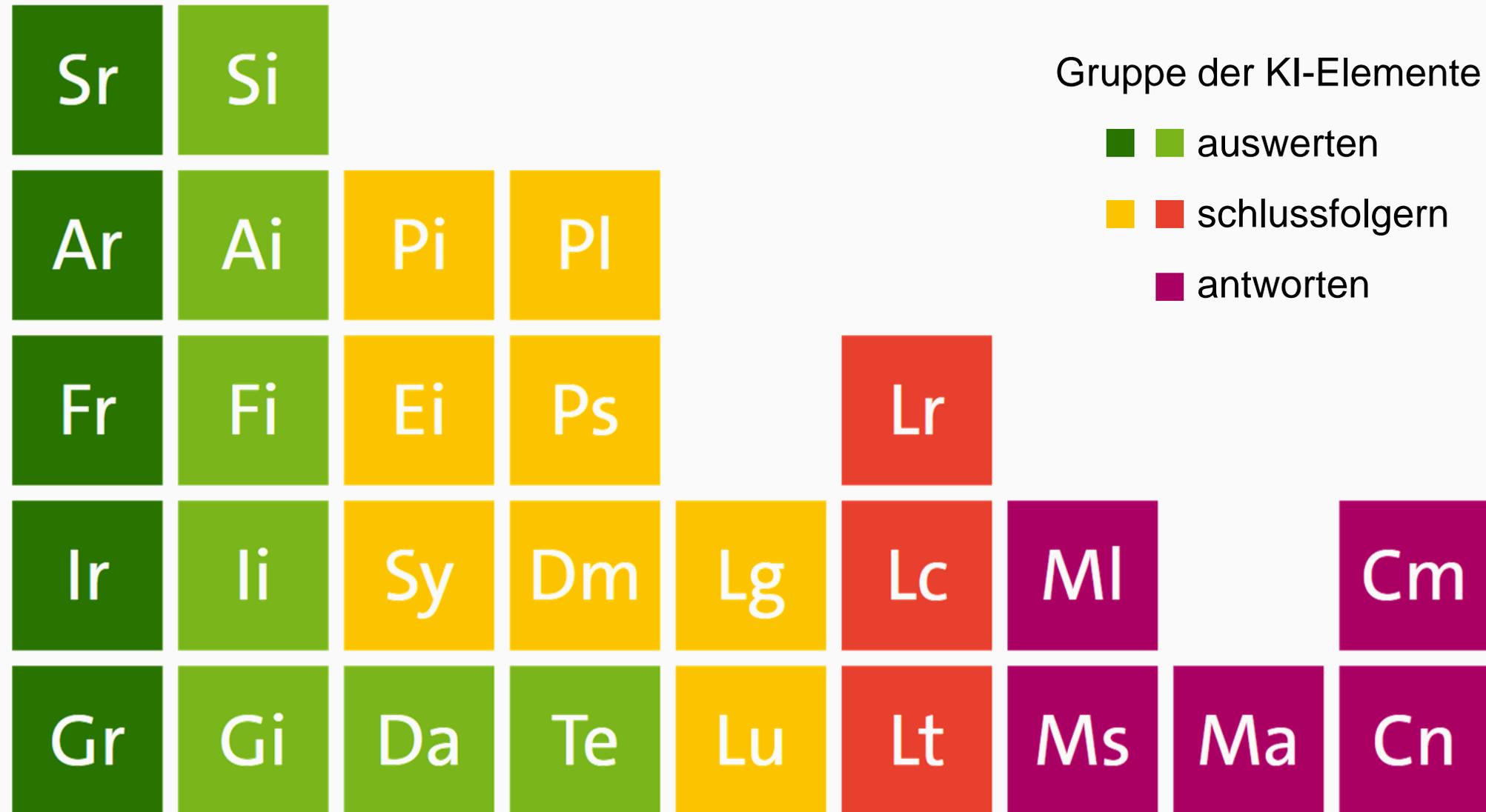
Voneinander Lehren lernen, 5 : 16-27, 2024. **Hochschule Osnabrück.**

https://opus.hs-osnabrueck.de/frontdoor/deliver/index/docId/5286/file/Voneinander_Lehren_lernen_Band_05.pdf#page=16

Schritte in denen üblicherweise programmiert wird



Periodensystem der Künstlichen Intelligenz



KI für Alle

**Generalisiertes didaktisches Planungsraster für
angewandte KI**

Nr.	Unterrichtseinheit	Beispiel Vorlesungstitel	Lernziele	Inhalte	Methoden	Material
1	Anwendungsidee und Datensatz	Ventum Solution GmbH und künstliche Intelligenz	<ul style="list-style-type: none"> • Verständnis über die Möglichkeiten von KI • Schritte der KI-Anwendungspipeline sind bekannt • Verknüpfung zwischen Lehrinhalten (Anwendungsaufgabe) und KI (Datensatz) 	Warum ist KI heute wichtig?	Fallstudie	Fallstudie
				Schilderung realer KI-Anwendungsbeispiele mit Diskussion	Impulsvortrag + Diskussion	Foliensatz + Leitfragen
				In welchen Schritten kann KI angewandt werden?	Vortrag	KI-Anwendungspipeline
				Von der Anwendungsidee zum Datensatz	Kleingruppenarbeit	Beispieldatensatz, Anwendungsidee
2	Datenanalyse und -bereinigung	<i>Big Data</i> für die Ventum Solution GmbH	<ul style="list-style-type: none"> • Bewertung der Qualität von Datensätzen • Auswahl von diskriminierungsfreien Daten • Starten der ausführbaren Webseiten 	Warum sind Daten das neue Gold?	Visuelle Inspektion	Datensatz mit bekannten Fehlern
				Wie gerecht sind Datensätze?	Visuelle Inspektion	Datensatz mit bekannten Ungleichgewichten
				Ausführbare Webseiten / <i>IPython</i>	Praxisarbeit am Computer	Vorgefertigte <i>IPython</i> -Webseiten
3	KI zur Klassifizierung	<i>BirdImageScan</i> – Windkraft im Einklang mit Naturschutz	<ul style="list-style-type: none"> • Übersicht über KI-Modelle zur Klassifizierung • Diskussionen – <ul style="list-style-type: none"> 1) Wann ist die KI „gut genug“ für eine bestimmte Anwendung? 2) Wie teuer ist die Verwendung von KI? 	Wie wird KI trainiert und evaluiert?	Praxisarbeit am Computer	Vorgefertigte <i>IPython</i> -Webseiten
				Welche KI-Modelle zur Klassifizierung gibt es?	Selbstlernaufgabe	Infomaterial, Linksammlung
				Evaluation von KI-Modellen	Kleingruppendiskussion	Vorgefertigte <i>IPython</i> -Webseiten
				Warum kann KI klimaschädlich sein?	Klassenraumdiskussion	„Warmer“ und „lauter“ Hörsaal (durch PC-Lüfter)
4	KI zur Zeitreihenprognose	Wann wird der Strom aus Wind und PV benötigt?	<ul style="list-style-type: none"> • Übersicht über KI-Modelle zur Zeitreihenprognose • Diskussion – Wann ist die KI „gut genug“ für diese Anwendung? 	Wie sieht ein Zeitreihendatensatz aus?	Kleingruppenarbeit	Beispieldatensatz, Anwendungsidee
				Welche KI-Modelle für Zeitreihen gibt es?	Selbstlernaufgabe	Informationsmaterial, Linksammlung
				Welche Faktoren wurden nicht berücksichtigt?	Kleingruppendiskussion	Vorgefertigte <i>IPython</i> -Webseiten

KI für Alle

Unterrichtseinheit 1: Anwendungsidee und Datensatz

Anwendungsbeispiel: Ventum Solution

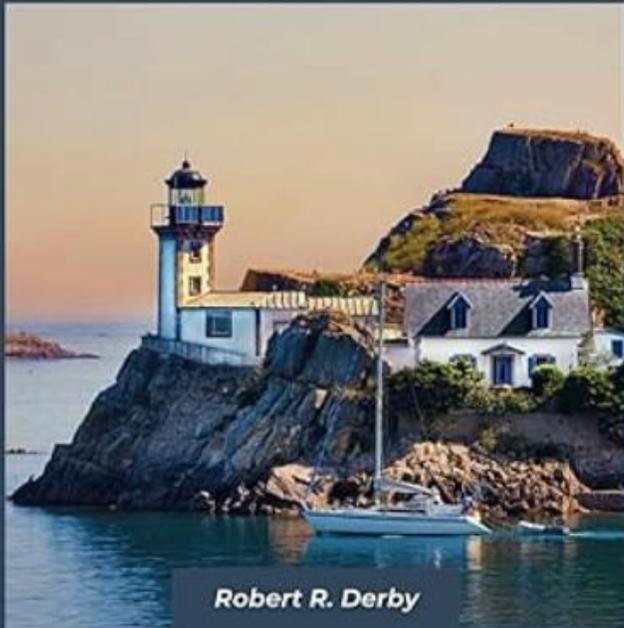
Fallstudie

„Ventum Solutions GmbH“



Anwendungsbeispiel: Büchererstellung

Reiseführer Bretagne und Normandie



Robert R. Derby

Der Ärmelkanal im Norden und der Atlantische Ozean im Westen bilden die herrlichen Meereslandschaften der Region. Das üppige Ackerland, die sanften Hügel und die malerischen Städte der Bretagne bieten eine ländliche Umgebung, die Besucher dazu einlädt, die schöne Umgebung zu erkunden.



Im Gegensatz dazu ist die Normandie für ihr abwechslungsreiches Gelände bekannt, das von riesigen Stränden über üppiges Ackerland bis hin zu hohen Klippen alles umfasst. Auch der Fluss Seine, der durch die herrliche Landschaft fließt, bevor er in den Ärmelkanal mündet, prägt das Gebiet. Die einzigartige Geographie der Normandie bietet eine Vielzahl natürlicher Schätze, wie die berühmten

16

Kreidefelsen von Etretat und die ruhige Ruhe der Halbinsel Cotentin.

Sowohl in der Bretagne als auch in der Normandie gibt es viele natürliche Besonderheiten, die die geografische Vielfalt der Länder unterstreichen. Touristen werden einer Vielzahl spektakulärer Landschaften ausgesetzt, die die Pracht dieser Orte verkörpern, vom berühmten Mont Saint-Michel, einer Inselgemeinde, die von einem mittelalterlichen Kloster gekrönt wird, bis hin zur rauen Schönheit der rosa Granitküste in der Bretagne und den breiten Sandstränden der Normandie.

Beide Bezirke zeichnen sich durch eine herrliche landwirtschaftliche Umgebung aus, mit sanften Weizen-, Mais- und Sonnenblumenfeldern, die sich soweit das Auge reicht erstrecken. Die ländliche Pracht der Region ist geprägt von wunderschönen Bauernhöfen, Apfelplantagen und grasenden Tieren, die einen ruhigen Hintergrund für entspannte Erkundungen und Outdoor-Aktivitäten schaffen.

Aufgrund ihrer geografischen Nähe verfügen die Bretagne und die Normandie über eine Meeresumwelt mit angenehmen Temperaturen und leichten Niederschlägen das ganze Jahr über. Diese angenehme Umgebung trägt zur reichen Vegetation

17

Anwendungsbeispiel: Büchererstellung

Your Text is AI/GPT Generated



Der Ärmelkanal im Norden und der Atlantische Ozean im Westen bilden die herrlichen Meereslandschaften der Region. Das üppige Ackerland, die sanften Hügel und die malerischen Städte der Bretagne bieten eine ländliche Umgebung, die Besucher dazu einlädt, die schöne Umgebung zu erkunden.

Im Gegensatz dazu ist die Normandie für ihr abwechslungsreiches Gelände bekannt, das von riesigen Stränden über üppiges Ackerland bis hin zu hohen Klippen alles umfasst. Auch der Fluss Seine, der durch die herrliche Landschaft fließt, bevor er in den Ärmelkanal mündet, prägt das Gebiet. Die einzigartige Geographie der Normandie bietet eine Vielzahl natürlicher Schätze, wie die berühmten Kreidefelsen von Etretat und die ruhige Ruhe der Halbinsel Cotentin.

Sowohl in der Bretagne als auch in der Normandie gibt es viele natürliche Besonderheiten, die die geografische Vielfalt der Länder unterstreichen. Touristen werden einer Vielzahl spektakulärer Landschaften ausgesetzt, die die Pracht dieser Orte verkörpern, vom berühmten Mont Saint-Michel, einer Inselgemeinde, die von einem mittelalterlichen Kloster gekrönt wird, bis hin zur rauen Schönheit der rosa Granitküste in der Bretagne und den breiten Sandstränden der Normandie.

Beide Bezirke zeichnen sich durch eine herrliche landwirtschaftliche Umgebung aus, mit sanften Weizen-, Mais- und Sonnenblumenfeldern, die sich soweit das Auge reicht erstrecken. Die ländliche Pracht der Region ist geprägt von wunderschönen Bauernhöfen, Apfelplantagen und grasenden Tieren, die einen ruhigen Hintergrund für entspannte Erkundungen und Outdoor-Aktivitäten schaffen.

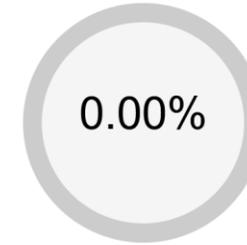
■ Highlighted text is suspected to be most likely generated by AI*

1.603 Characters

219 Words



Ihr Inhalt scheint von GPT-Technologie erstellt worden zu sein.



wurde von einem Menschen verfasst



wurde von einer KI/GPT generiert

■ *Der markierte Text scheint von einem KI-System generiert worden zu sein.

Der Ärmelkanal im Norden und der Atlantische Ozean im Westen bilden die herrlichen Meereslandschaften der Region. Das üppige Ackerland, die sanften Hügel und die malerischen Städte der Bretagne bieten eine ländliche Umgebung, die Besucher dazu einlädt, die schöne Umgebung zu erkunden.

Im Gegensatz dazu ist die Normandie für ihr abwechslungsreiches Gelände bekannt, das von riesigen Stränden über üppiges Ackerland bis hin zu hohen Klippen alles umfasst. Auch der Fluss Seine, der durch die herrliche Landschaft fließt, bevor er in den Ärmelkanal mündet, prägt das Gebiet. Die einzigartige Geographie der Normandie bietet eine Vielzahl natürlicher Schätze, wie die berühmten Kreidefelsen von Etretat und die ruhige Ruhe der Halbinsel Cotentin.

Sowohl in der Bretagne als auch in der Normandie gibt es viele natürliche Besonderheiten, die die geografische Vielfalt der Länder unterstreichen. Touristen werden einer Vielzahl spektakulärer Landschaften ausgesetzt, die die Pracht dieser Orte verkörpern, vom berühmten Mont Saint-Michel, einer Inselgemeinde, die von einem mittelalterlichen Kloster gekrönt wird, bis hin zur rauen Schönheit der rosa Granitküste in der Bretagne und den breiten Sandstränden der Normandie.

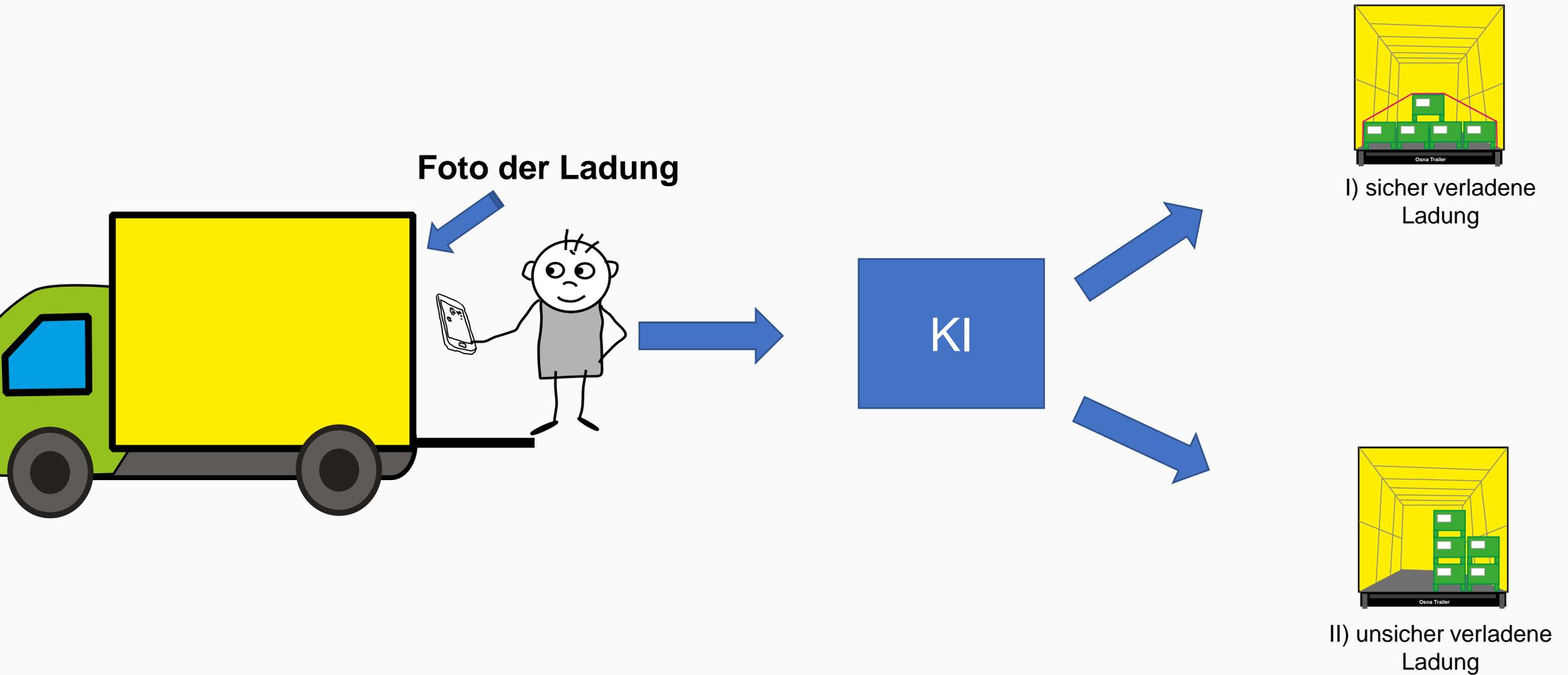
Beide Bezirke zeichnen sich durch eine herrliche landwirtschaftliche Umgebung aus, mit sanften Weizen-, Mais- und Sonnenblumenfeldern, die sich soweit das Auge reicht erstrecken. Die ländliche Pracht der Region ist geprägt von wunderschönen Bauernhöfen, Apfelplantagen und grasenden Tieren, die einen ruhigen Hintergrund für entspannte Erkundungen und Outdoor-Aktivitäten schaffen.

Neuen Text erkennen

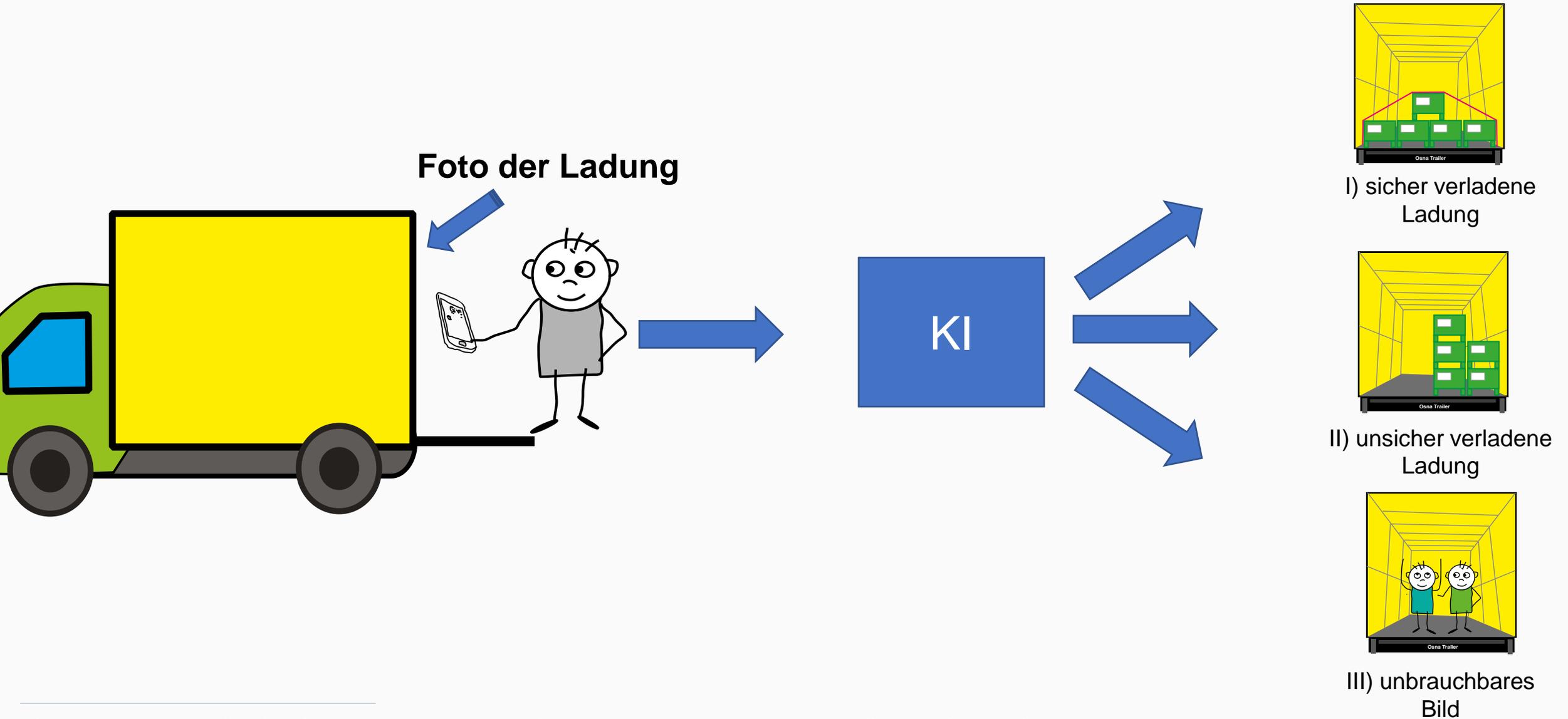
KI für Alle

Unterrichtseinheit 2: Datenanalyse und -bereinigung

Datenanalyse und -bereinigung



Datenanalyse und -bereinigung



Datenanalyse und -bereinigung

Die Erstellung des Datensatzes nimmt in der Realität den größten Teil des Aufwandes - Zeit und Kosten - in Anspruch. Um zu zeigen, dass Ventum Solutions GmbH in der Lage ist zuverlässig Vögel zu detektieren, wird anstatt eigene Aufnahmen ein existierender Datensatz aus der Forschung genutzt. Mit diesem Datensatz wird eine prototypische Implementierung des für das Genehmigungsverfahren benötigten Vogelerkennungsprogrammes erstellt.

Der hier genutzte Datensatz wurde im Rahmen einer Studie [1], [2] in Japan erstellt. Hierfür wurden über drei Tage lang alle zwei Sekunden Aufnahmen vor einer Windkraftanlage gemacht und alle Objekte wie Vögel, Fliegen, Flugzeuge, etc. händisch darauf markiert. Insgesamt entstanden in dieser Studie 15.000 Bilder, auf denen mal Vögel, Flugzeuge, Insekten oder auch nichts zu sehen war.

(a) Klasse Vögel (b) Klasse Verschiedene (c) Klasse Nichts

Abbildung 1: Beispielbilder aus allen drei Klassen

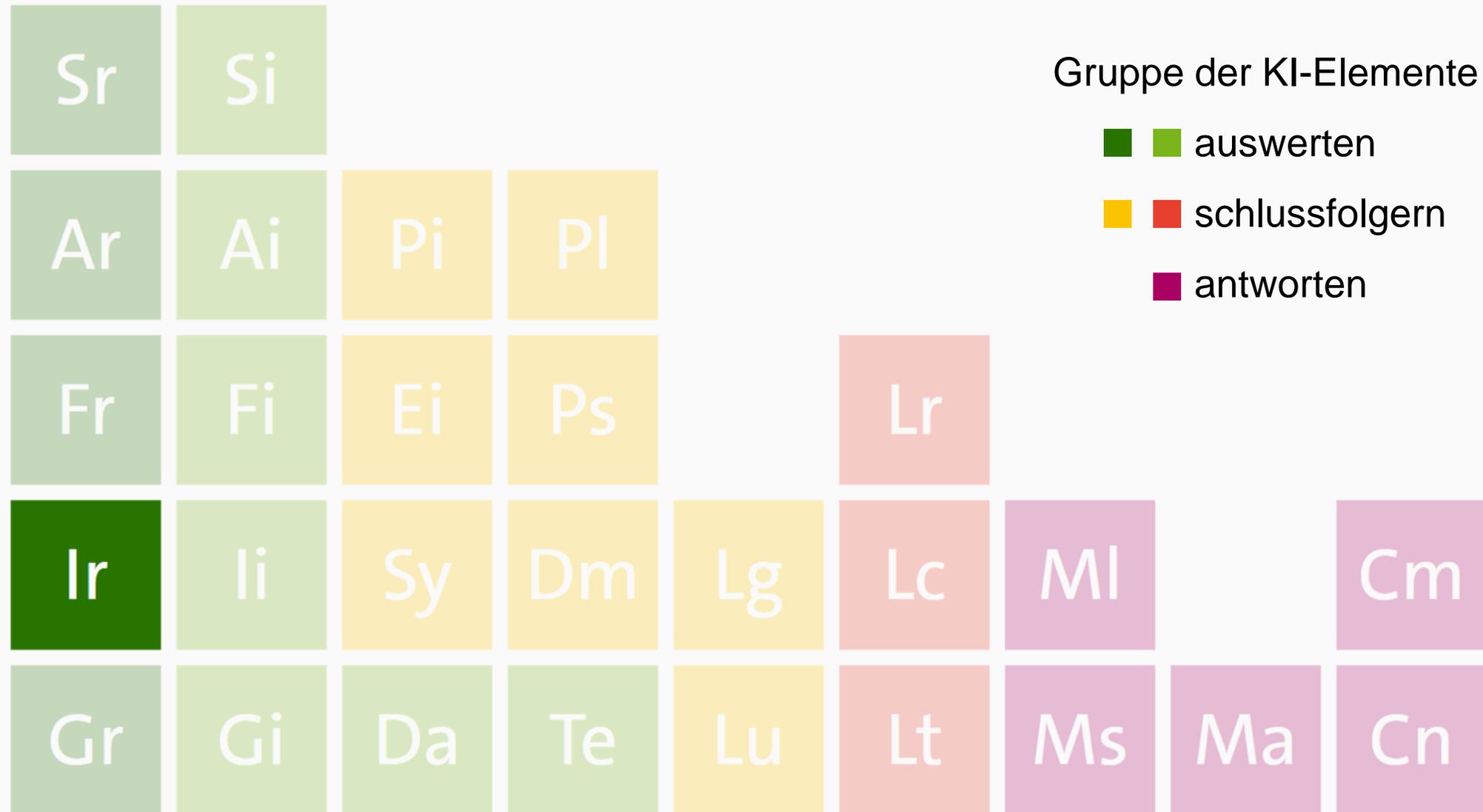
Die von Ventum Solutions GmbH angedachten Windkraftanlagen verfügen über ein "Schnell-Stop-System" und können binnen weniger Sekunden angehalten werden. Aus diesem Grund können die Bilder aus dem Datensatz von Yoshihashi et al. [1], [2] auf einen Bildausschnitt von 3100×1714 Pixel verkleinert werden (Originalgröße von 5616×3744), da schnell reagiert werden kann. Der abgeänderte Datensatz besteht jetzt aus drei Kategorien - auch Klassen genannt. In der ersten Klasse *Vögel*, vgl. Abbildung 1 (a), sind an beliebiger Stelle auf dem Bildausschnitt ein oder mehrere Vögel zu sehen. Die zweite Klasse *Verschiedene*, siehe Abbildung 1 (b), beinhaltet verschiedene "nicht-Vogel" Objekte wie Fliegen oder Flugzeuge. Auf Bildern der dritten Klasse *Nichts*, ist nur die Windkraftanlage zu sehen. Von jeder Klasse gibt es 50 verschiedene Bilder, sodass der aufbereitete Datensatz aus 150 Bildern besteht.

Dieser aufbereitete Datensatz ermöglicht das Entwerfen, Trainieren und Evaluieren eines ANNs zur Bildklassifikation auf einem normalen Computer bzw. Laptop, ohne dafür einen Hochleistungsrechner nutzen zu müssen.

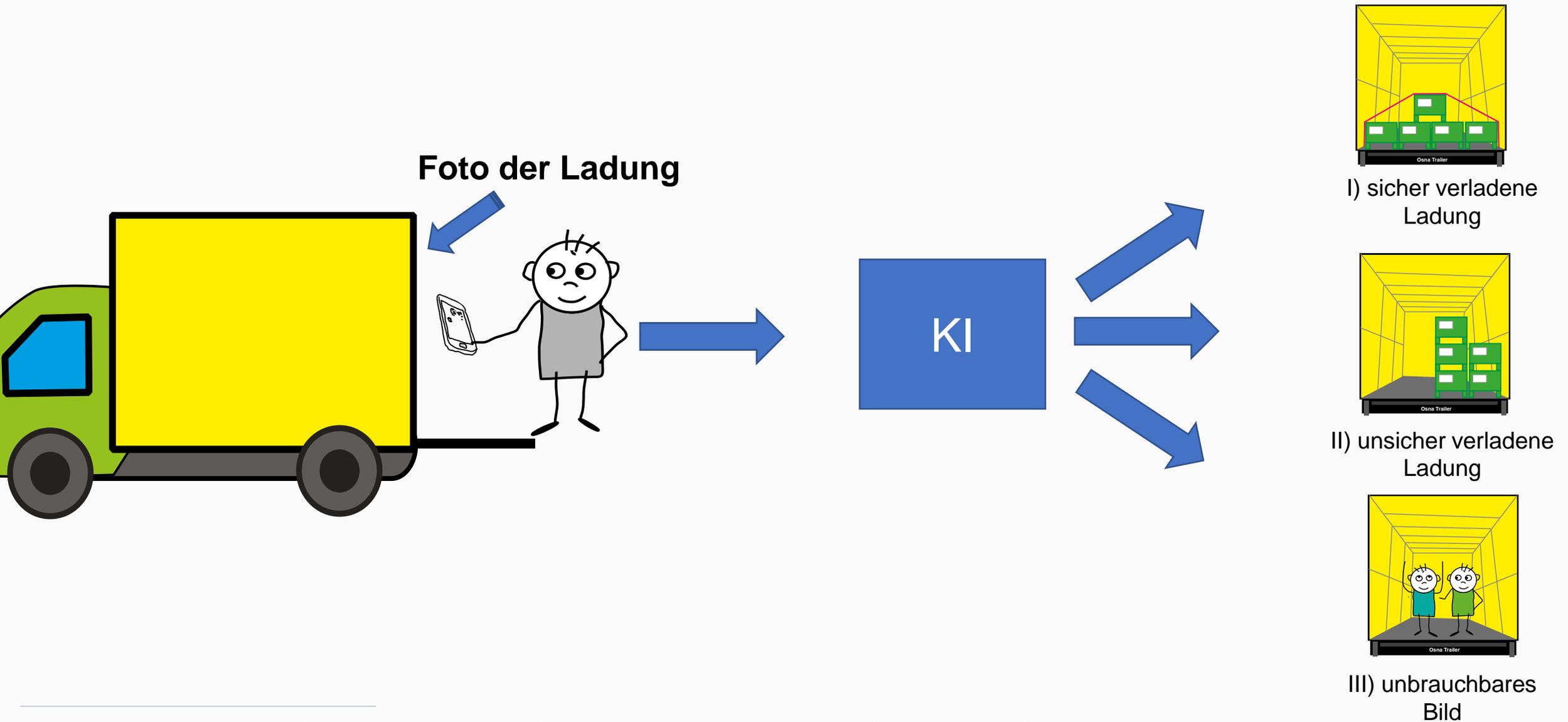
KI für Alle

Unterrichtseinheit 3: KI zur Klassifizierung

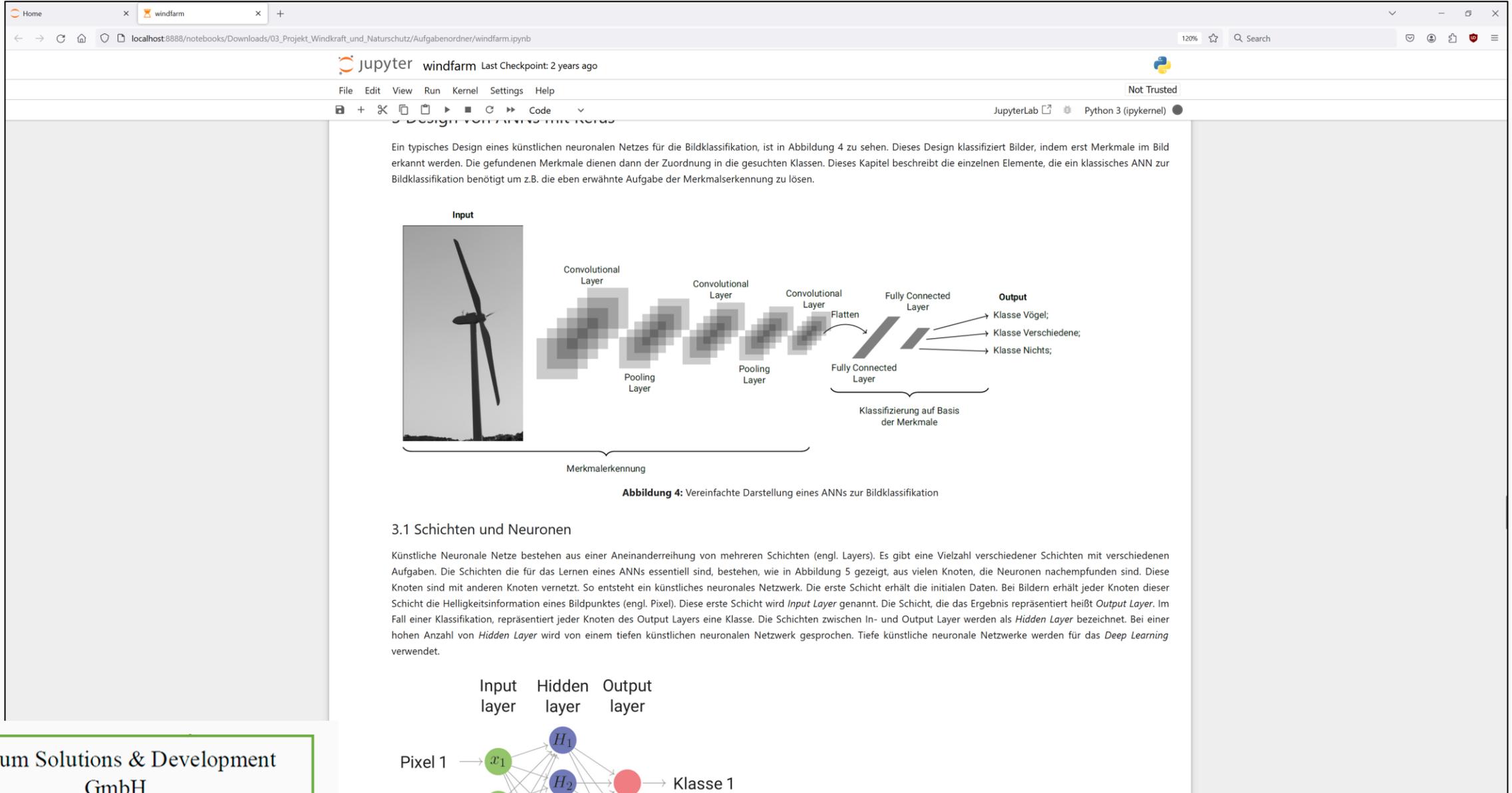
[Ir] Image Recognition – Zuordnung von Objekten



Anwendungsbeispiel: KI-gestützte Bewertung der Ladungssicherheit



Anwendungsbeispiel: KI-gestützte Abschaltung von Windkraftanlagen



The screenshot shows a JupyterLab notebook interface with a browser window at the top displaying the URL `localhost:8888/notebooks/Downloads/03_Projekt_Windkraft_und_Naturschutz/Aufgabenordner/windfarm.ipynb`. The notebook content includes a title "Design von ANNs mit keras", a paragraph explaining the typical design of a CNN for image classification, and a diagram of such a network. The diagram starts with an "Input" image of a wind turbine, followed by a "Merkmalerkennung" (feature recognition) section containing three "Convolutional Layer" and two "Pooling Layer". This is followed by a "Flatten" step, then a "Fully Connected Layer" (labeled "Klassifizierung auf Basis der Merkmale"), and finally an "Output" section with three classes: "Klasse Vögel;", "Klasse Verschiedene;", and "Klasse Nichts;".

Ein typisches Design eines künstlichen neuronalen Netzes für die Bildklassifikation, ist in Abbildung 4 zu sehen. Dieses Design klassifiziert Bilder, indem erst Merkmale im Bild erkannt werden. Die gefundenen Merkmale dienen dann der Zuordnung in die gesuchten Klassen. Dieses Kapitel beschreibt die einzelnen Elemente, die ein klassisches ANN zur Bildklassifikation benötigt um z.B. die eben erwähnte Aufgabe der Merkmalerkennung zu lösen.

Input

Convolutional Layer
Convolutional Layer
Convolutional Layer
Pooling Layer
Pooling Layer
Flatten
Fully Connected Layer
Fully Connected Layer

Output

- Klasse Vögel;
- Klasse Verschiedene;
- Klasse Nichts;

Klassifizierung auf Basis der Merkmale

Merkmalerkennung

Abbildung 4: Vereinfachte Darstellung eines ANNs zur Bildklassifikation

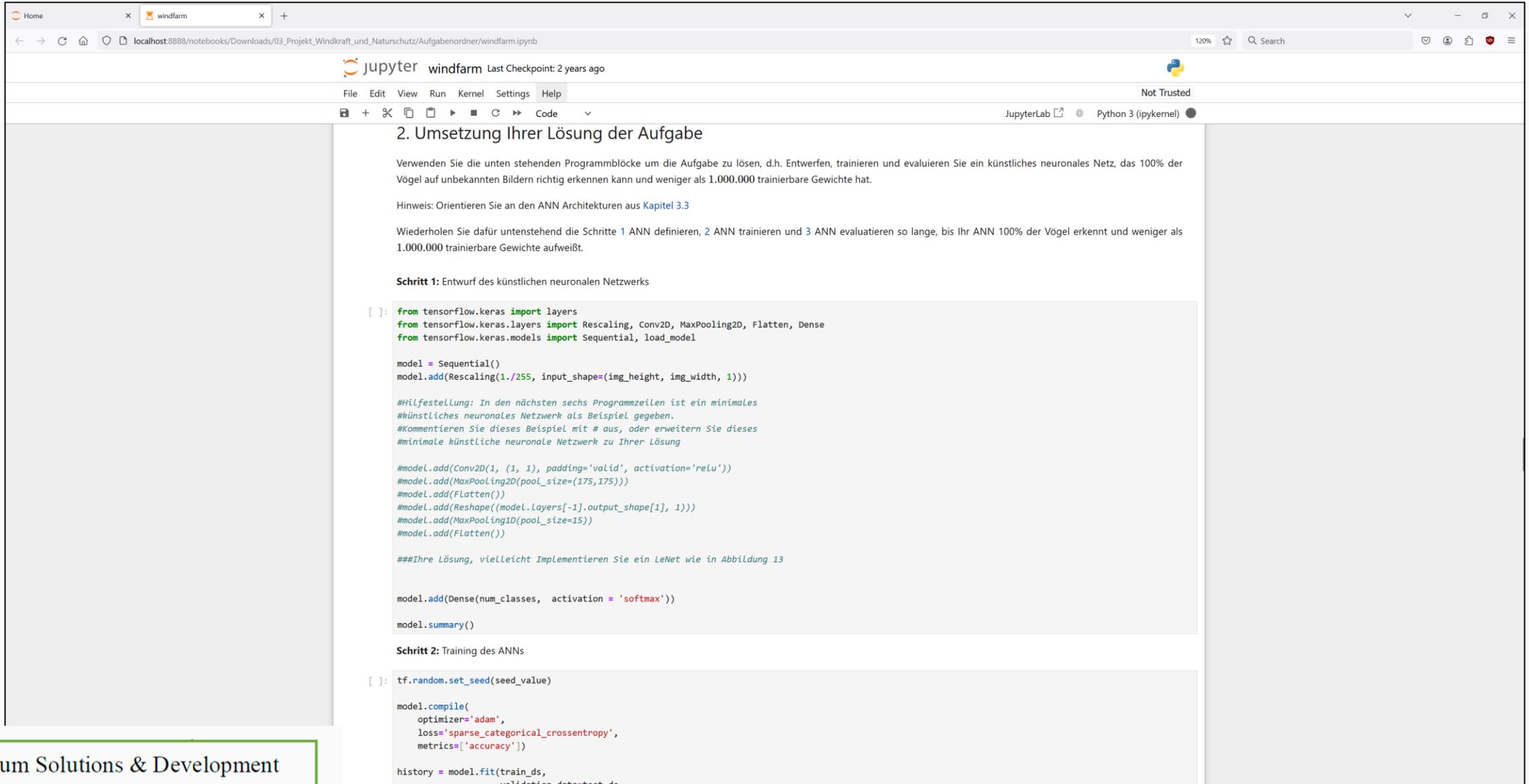
3.1 Schichten und Neuronen

Künstliche Neuronale Netze bestehen aus einer Aneinanderreihung von mehreren Schichten (engl. Layers). Es gibt eine Vielzahl verschiedener Schichten mit verschiedenen Aufgaben. Die Schichten die für das Lernen eines ANNs essentiell sind, bestehen, wie in Abbildung 5 gezeigt, aus vielen Knoten, die Neuronen nachempfunden sind. Diese Knoten sind mit anderen Knoten vernetzt. So entsteht ein künstliches neuronales Netzwerk. Die erste Schicht erhält die initialen Daten. Bei Bildern erhält jeder Knoten dieser Schicht die Helligkeitsinformation eines Bildpunktes (engl. Pixel). Diese erste Schicht wird *Input Layer* genannt. Die Schicht, die das Ergebnis repräsentiert heißt *Output Layer*. Im Fall einer Klassifikation, repräsentiert jeder Knoten des Output Layers eine Klasse. Die Schichten zwischen In- und Output Layer werden als *Hidden Layer* bezeichnet. Bei einer hohen Anzahl von *Hidden Layer* wird von einem tiefen künstlichen neuronalen Netzwerk gesprochen. Tiefe künstliche neuronale Netzwerke werden für das *Deep Learning* verwendet.

Input layer Hidden layer Output layer

Pixel 1 → x_1 → H_1 → H_2 → Klasse 1

Anwendungsbeispiel: KI-gestützte Abschaltung von Windkraftanlagen



2. Umsetzung Ihrer Lösung der Aufgabe

Verwenden Sie die unten stehenden Programmblöcke um die Aufgabe zu lösen, d.h. Entwerfen, trainieren und evaluieren Sie ein künstliches neuronales Netz, das 100% der Vögel auf unbekanntem Bildern richtig erkennen kann und weniger als 1.000.000 trainierbare Gewichte hat.

Hinweis: Orientieren Sie an den ANN Architekturen aus [Kapitel 3.3](#)

Wiederholen Sie dafür untenstehend die Schritte 1 ANN definieren, 2 ANN trainieren und 3 ANN evaluieren so lange, bis Ihr ANN 100% der Vögel erkennt und weniger als 1.000.000 trainierbare Gewichte aufweist.

Schritt 1: Entwurf des künstlichen neuronalen Netzwerks

```
[ ]: from tensorflow.keras import layers
from tensorflow.keras.layers import Rescaling, Conv2D, MaxPooling2D, Flatten, Dense
from tensorflow.keras.models import Sequential, load_model

model = Sequential()
model.add(Rescaling(1./255, input_shape=(img_height, img_width, 1)))

#Hilfestellung: In den nächsten sechs Programmzeilen ist ein minimales
#künstliches neuronales Netzwerk als Beispiel gegeben.
#Kommentieren Sie dieses Beispiel mit # aus, oder erweitern Sie dieses
#minimale künstliche neuronale Netzwerk zu Ihrer Lösung

#model.add(Conv2D(1, (1, 1), padding='valid', activation='relu'))
#model.add(MaxPooling2D(pool_size=(175,175)))
#model.add(Flatten())
#model.add(Reshape((model.Layers[-1].output_shape[1], 1)))
#model.add(MaxPooling1D(pool_size=15))
#model.add(Flatten())

###Ihre Lösung, vielleicht Implementieren Sie ein LeNet wie in Abbildung 13

model.add(Dense(num_classes, activation = 'softmax'))

model.summary()
```

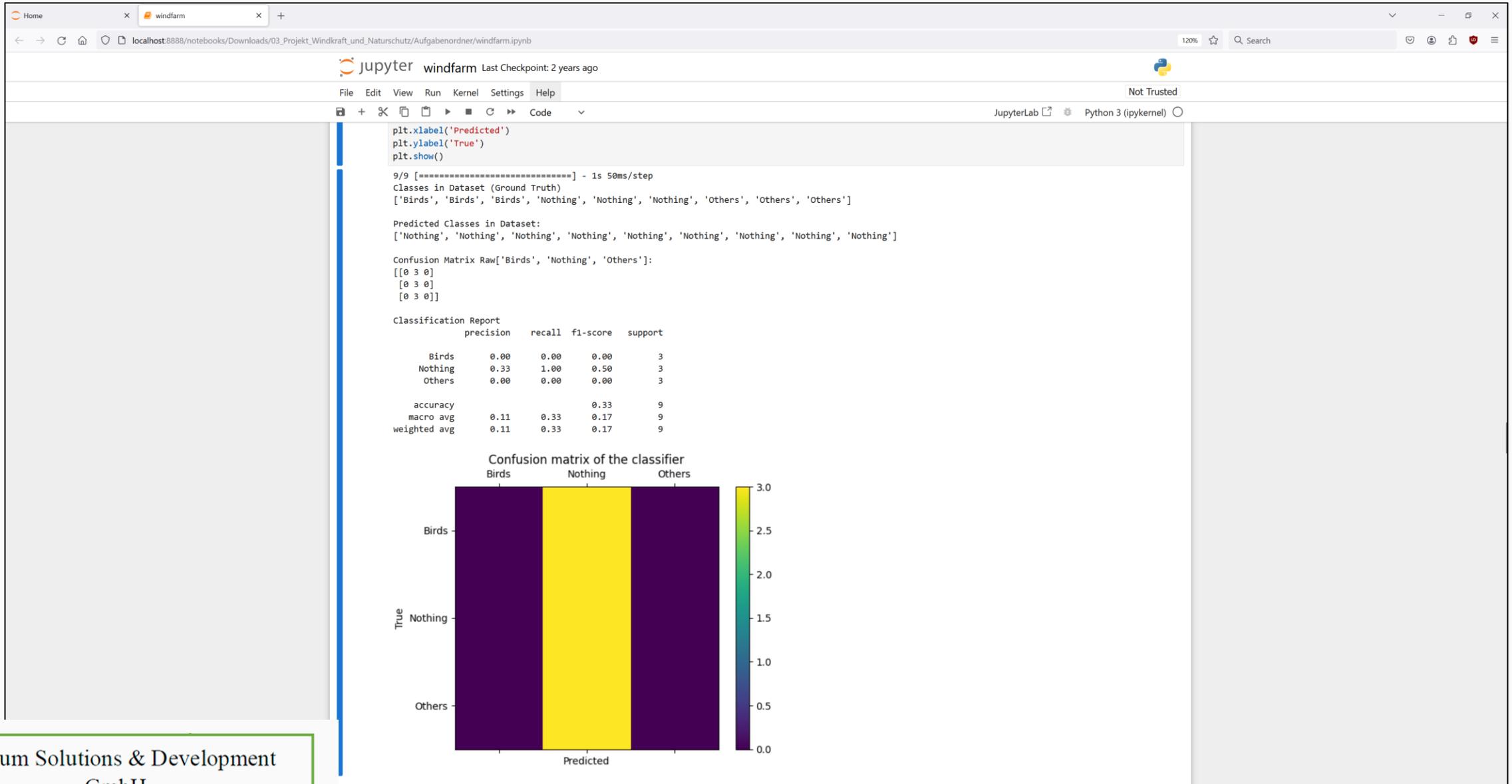
Schritt 2: Training des ANNs

```
[ ]: tf.random.set_seed(seed_value)

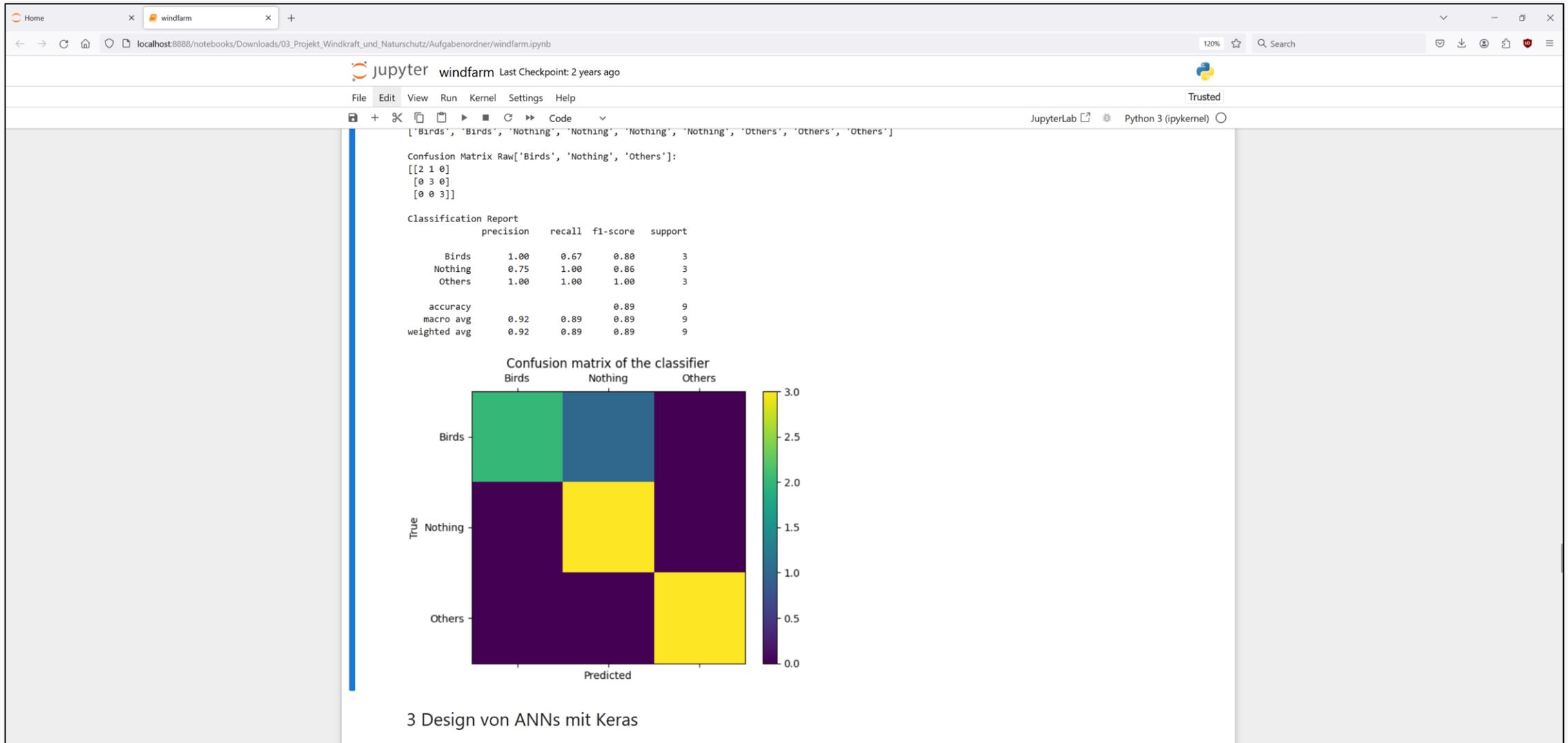
model.compile(
    optimizer='adam',
    loss='sparse_categorical_crossentropy',
    metrics=['accuracy'])

history = model.fit(train_ds,
                    validation_data=test_ds,
```

Anwendungsbeispiel: KI-gestützte Abschaltung von Windkraftanlagen



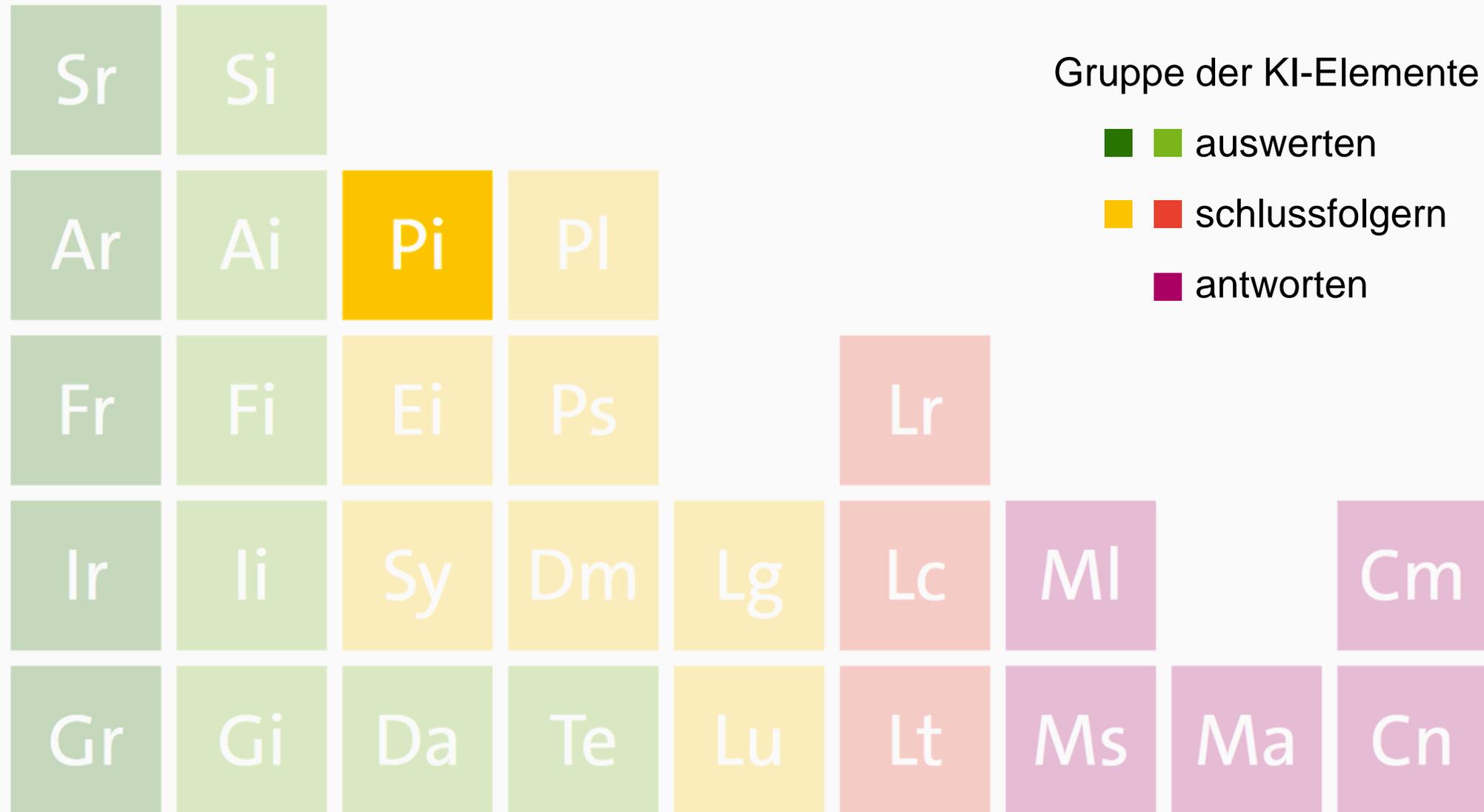
Anwendungsbeispiel: KI-gestützte Abschaltung von Windkraftanlagen



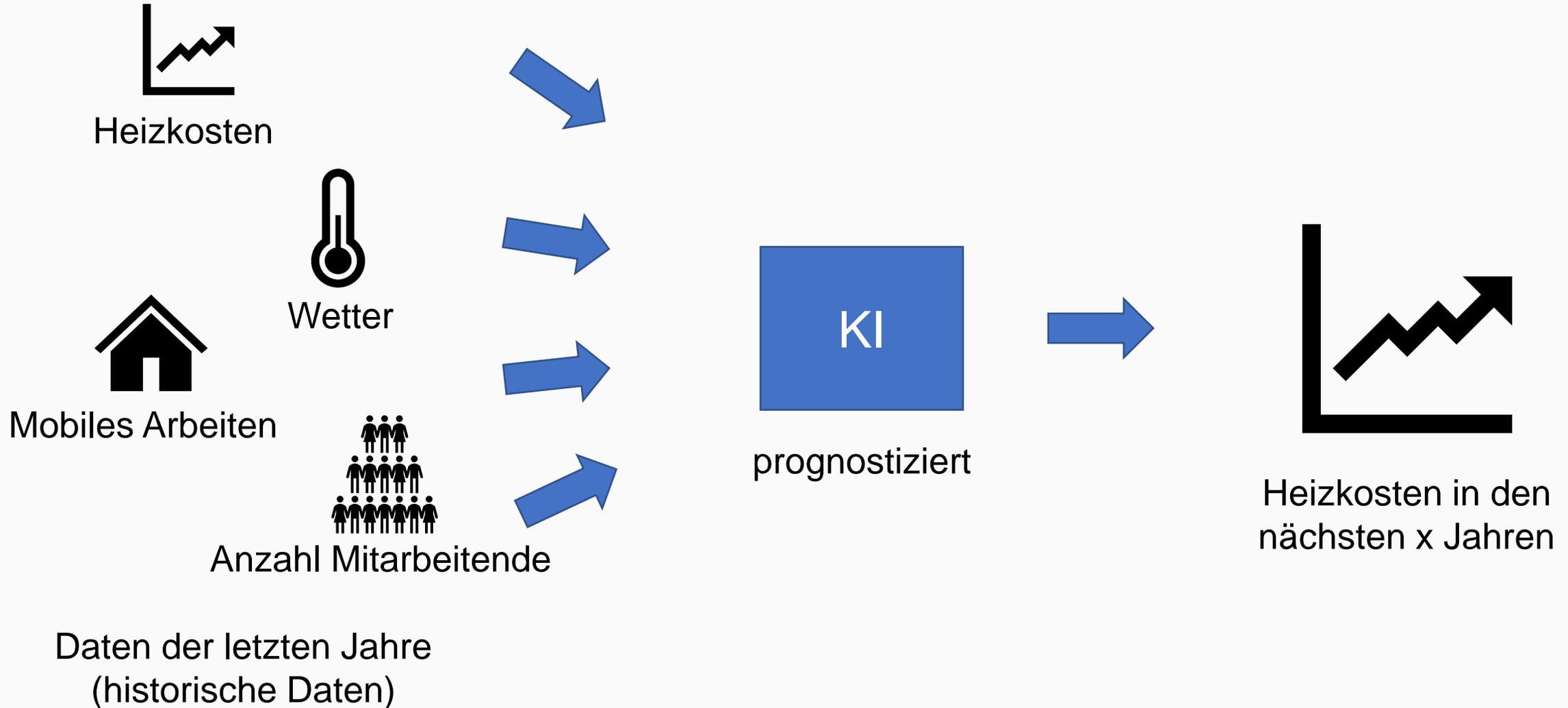
KI für Alle

Unterrichtseinheit 4: KI zur Zeitreihenprognose

[Pi] Predictive Inference – Statistische Prognosemodelle



Anwendungsbeispiel: Statistische Prognosemodelle – Heizkosten



Prädiktion des Stromverbrauches

bei allen rekurrenten Schichten, die nicht die letzte im Netzwerk sind auf True gesetzt werden.

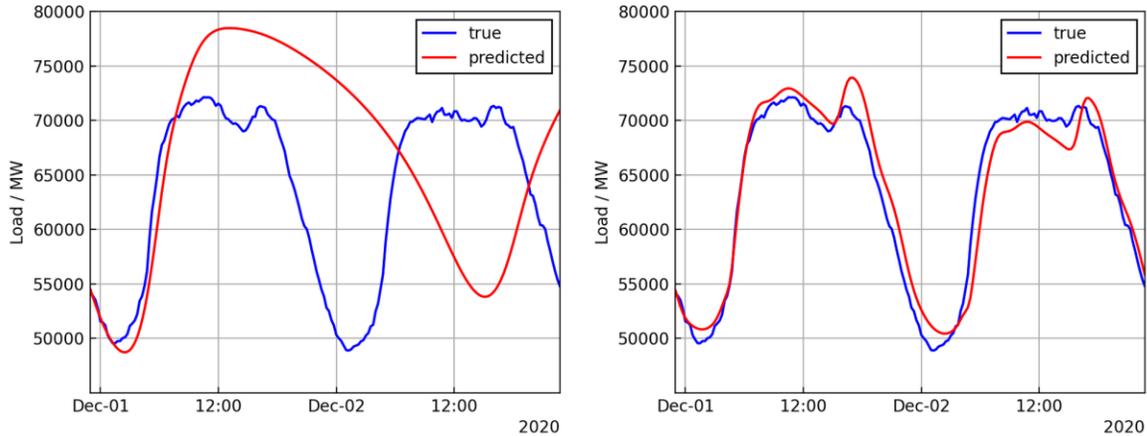
```
model.add(LSTM(32, return_sequences=True))
```

4. Ausblick: Wie Zeitreihenvorhersage richtig gemacht wird

Die in Kapitel 3 beschriebenen LSTM-Zellen sind ein wichtiger Baustein für die Verarbeitung von sequenziellen Daten mit Neuronalen Netzwerken. Es gibt jedoch viele verschiedene Problemstellungen im Bereich der sequenziellen Daten, und für jede Problemstellung gibt es Modellarchitekturen, die sich mehr oder weniger gut eignen.

Wenn Sie in Teilaufgabe 2 Ihr LSTM-Modell dazu benutzen, den Verlauf der nächsten zwei Tage vorherzusagen werden Sie feststellen, dass die Vorhersage nach wenigen Zeitschritten stark von der Wahrheit abweicht, auch wenn Sie zunächst sehr genau ist. Bei genauerem Nachdenken fällt der Grund dafür auf: Gibt es auch nur einen kleinen, aber systematischen Fehler in den Vorhersagen des Modells, so rechnet es für den kommenden Zeitschritt bereits mit falschen Annahmen, wie in Abbildung 6 (a) zu sehen. Dazu kommt, dass auch die internen, rekurrenten Zustände des Netzwerks sich immer weiter von aus dem Training bekannten Situationen entfernen - das Modell entgleist regelrecht.

Die Lösung: Es bedarf einer Architektur, die explizit darauf trainiert werden kann, ganze Sequenzen vorherzusagen. Im Idealfall lernt das Modell dann selbstständig, robuster gegenüber den eigenen Fehlern zu sein, wie das Ergebnis in Abbildung 6 (b) zeigt.



(a) Einfaches LSTM

(b) Encoder-Decoder

Prädiktion des Stromverbrauches

```
[*]: import tensorflow as tf
if len(tf.config.list_physical_devices('GPU'))>0:
    physical_devices = tf.config.list_physical_devices('GPU')
    tf.config.experimental.set_memory_growth(physical_devices[0], enable=True)
|
from subscripts import train_lstm
train_lstm.train_net(build_model, epochs=25)

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
lstm (LSTM)                  (None, 672, 120)           58560
lstm_1 (LSTM)                (None, 672, 60)            43440
lstm_2 (LSTM)                (None, 30)                  10920
dense (Dense)                (None, 1)                   31
-----
Total params: 112,951
Trainable params: 112,951
Non-trainable params: 0

Epoch 1/25
338/Unknown - 904s 3s/step - loss: 0.0051 - mape: 11.9932
```

Falls Sie in Schritt 1 eine valide Modell-Definition geschrieben haben, sollte Ihre Ausgabe grob wie folgt aussehen:

```
Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
lstm (LSTM)                  (None, 672, 32)            4352
lstm_1 (LSTM)                (None, 32)                  8320
dense (Dense)                (None, 1)                   33
-----
Total params: 12,705
Trainable params: 12,705
Non-trainable params: 0

Epoch 1/2
```

Abbildung 3: Beispielhafte Ausgabe eines validen LSTM Modell

Wichtig: Die Schritte 1 und 2 sind als iterativer Prozess zu verstehen. Es ist Teil des Arbeitsablaufs eines jeden Programmierers, festzustellen, dass der soeben geschriebene Code nicht das erwartete Ergebnis, sondern eine Fehlermeldung produziert. In diesem Fall gilt es, die Ursache der Fehlermeldung zu verstehen und zu beheben. Dabei ist es auch nicht ungewöhnlich, dass nur immer noch nicht das gewünschte Ergebnis, sondern eine weitere, neue Fehlermeldung erscheint. Lassen Sie sich davon nicht verunsichern. Oft

KI für Alle

Checkliste – KI in der Fachvorlesung bzw. im Modul

Checkliste – KI in der Fachvorlesung bzw. im Modul

Gib es eine Klassifikationsaufgabe in der Fachdomäne?

ja / nein

Wenn ja:

Ist diese Aufgabe für die Studierenden verständlich?

ja / nein

Hat diese Aufgabe mindestens zwei, aber maximal zehn Klassen?

ja / nein

Gibt es mindestens 100 Beispiele pro Klasse?

ja / nein

Gibt es für diese Aufgabe einen frei zugänglichen Datensatz?

ja / nein

Ist der verfügbare Datensatz kleiner als 4 Gigabyte?

ja / nein

Gib es eine Zeitreihenprognoseaufgabe in der Fachdomäne?

ja / nein

Wenn ja:

Ist diese Aufgabe für die Studierenden verständlich?

ja / nein

Gibt es zusammenhängende Zeitreihen-Daten mit gleichbleibender Abtastrate?

ja / nein

Gibt es für diese Aufgabe einen frei zugänglichen Datensatz?

ja / nein

Ist die Zeitreihe im Datensatz mindestens doppelt so lang wie die Vorhersage?

ja / nein

Ist der verfügbare Datensatz kleiner als 4 Gigabyte?

ja / nein

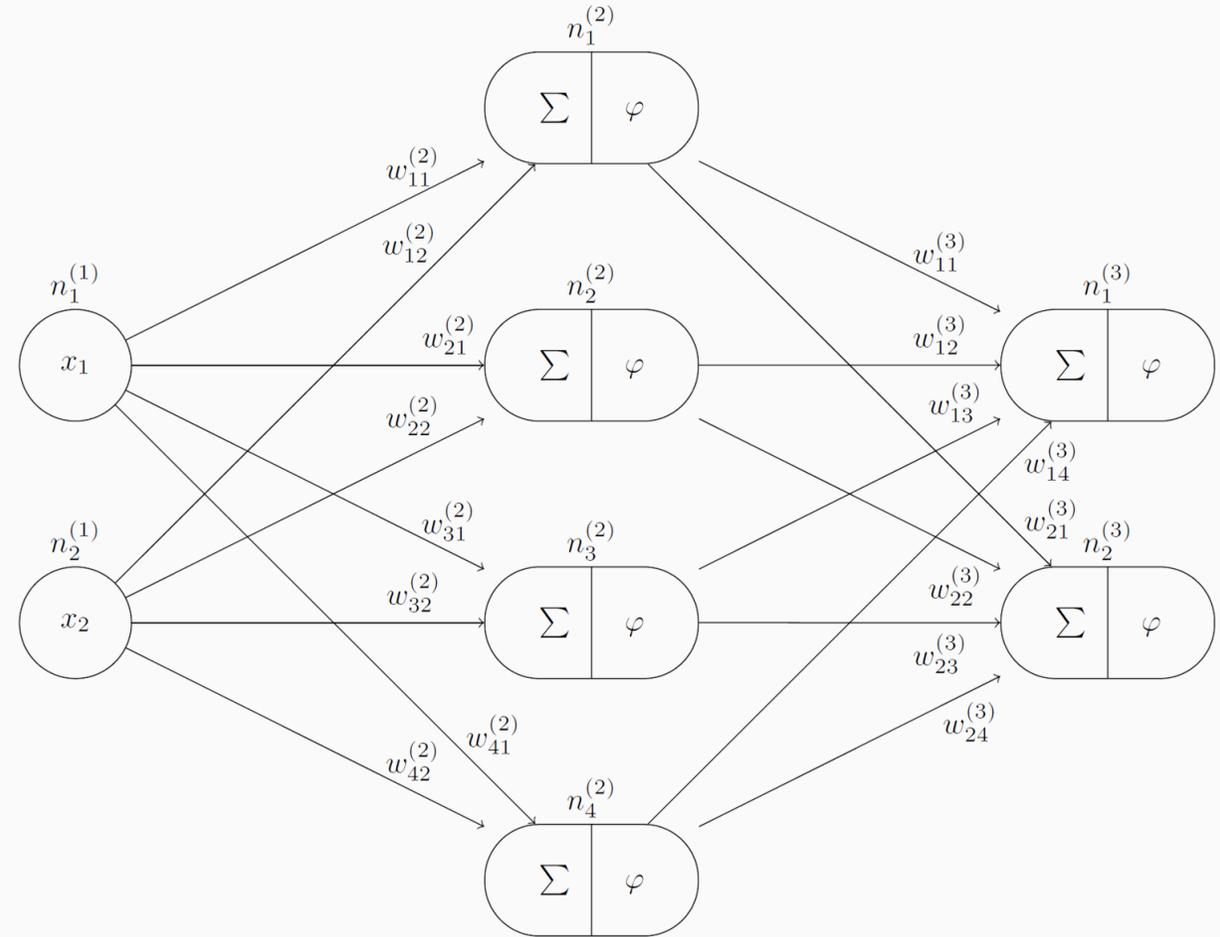
Haben Sie Kontakt zu einer*einem Kolleg*in mit Programmiererfahrung,
die*der beim Erstellen der ausführbaren Webseiten unterstützen kann?

ja / nein

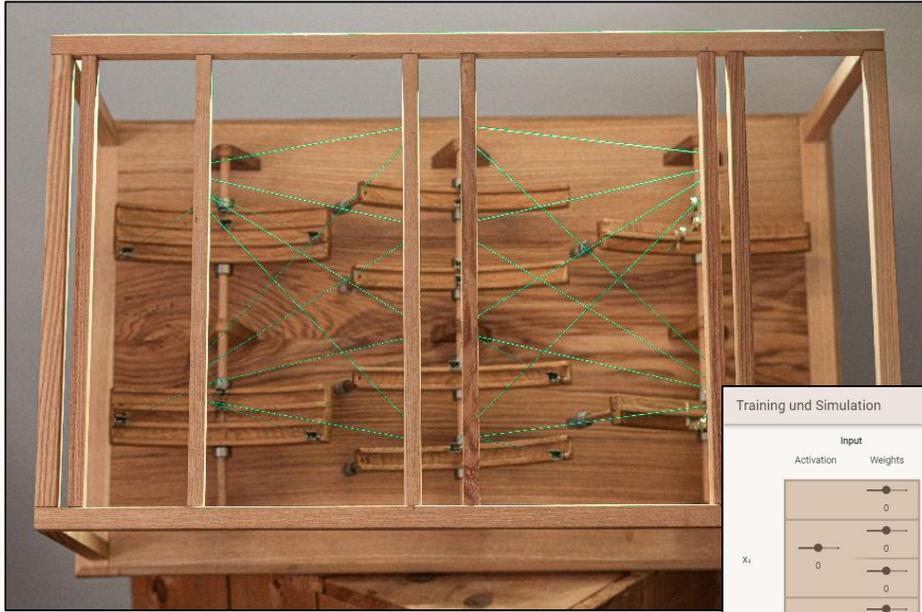
KI für Alle

weitere Ansätze

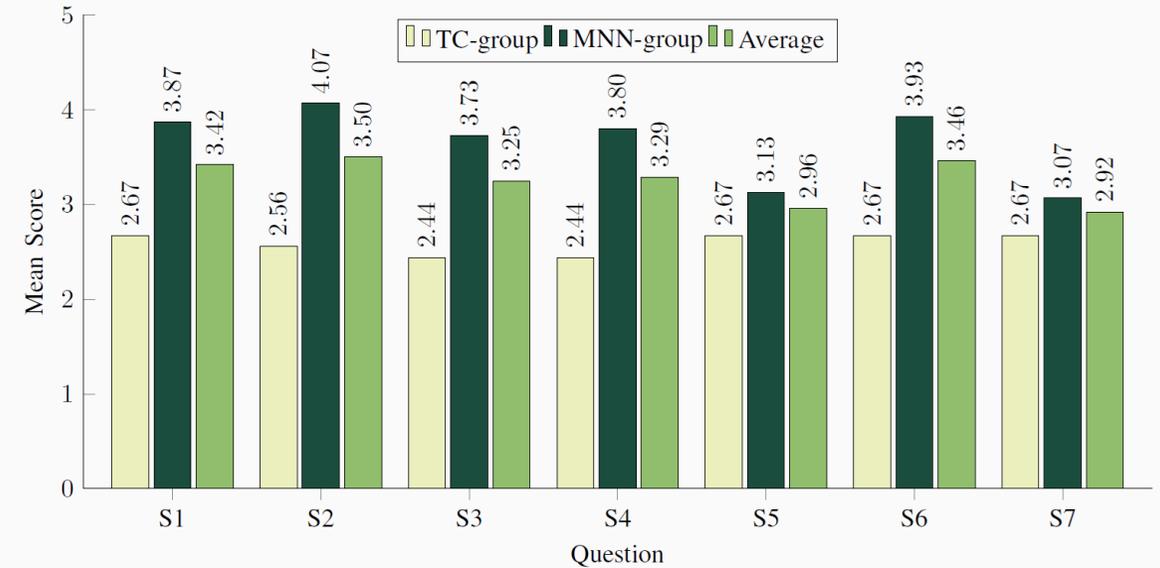
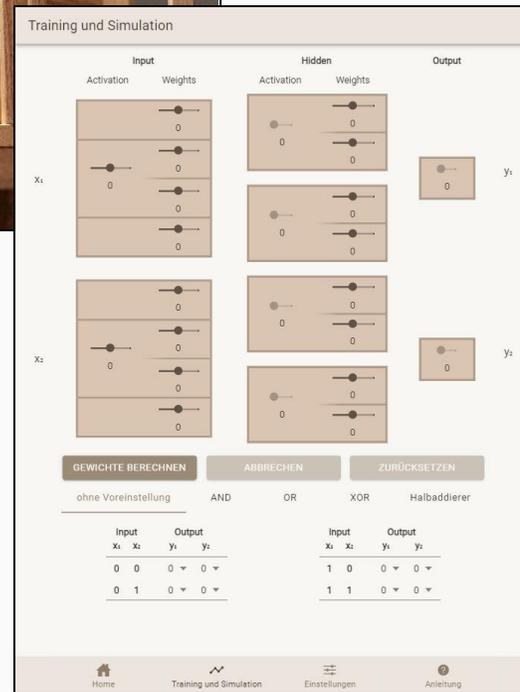
Mechanische Neuronale Netz



Mechanische Neuronale Netz + App



- Digitaler Zwilling auf dem Tablett
- Lehrmaterial
- Live-Schulung





Prof. Dr. Julius Schöning
j.schoening@hs-osnabrueck.de

Veröffentlichungen

 [0000-0003-4921-5179](https://orcid.org/0000-0003-4921-5179)

 [Research Gate](#)

 [Homepage](#)