

Image credit: [Volvo cars](https://www.volvocars.com)

Künstliche Intelligenz (KI) in SCHULEN

Projekt: Calliope-Rennspiel

Wissen zu Künstlicher Intelligenz spielerisch in die Schulen bringen

- SuS-fokussierte Präsentation für Workshops -

Ferenc Hechler, Mirko Jelinek, Christian Schiller, Dirk Wolters



LIFE IS FOR SHARING.

[AI@SCHOOL](https://ai@school.de) – [Autonomes Fahren mit dem Calliope mini](https://ai@school.de) - Seite 1



Dieses Material steht unter der Creative-Commons-Lizenz Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International. Um eine Kopie dieser Lizenz zu sehen, besuchen Sie <http://creativecommons.org/licenses/by-sa/4.0/>

Unser Gehirn



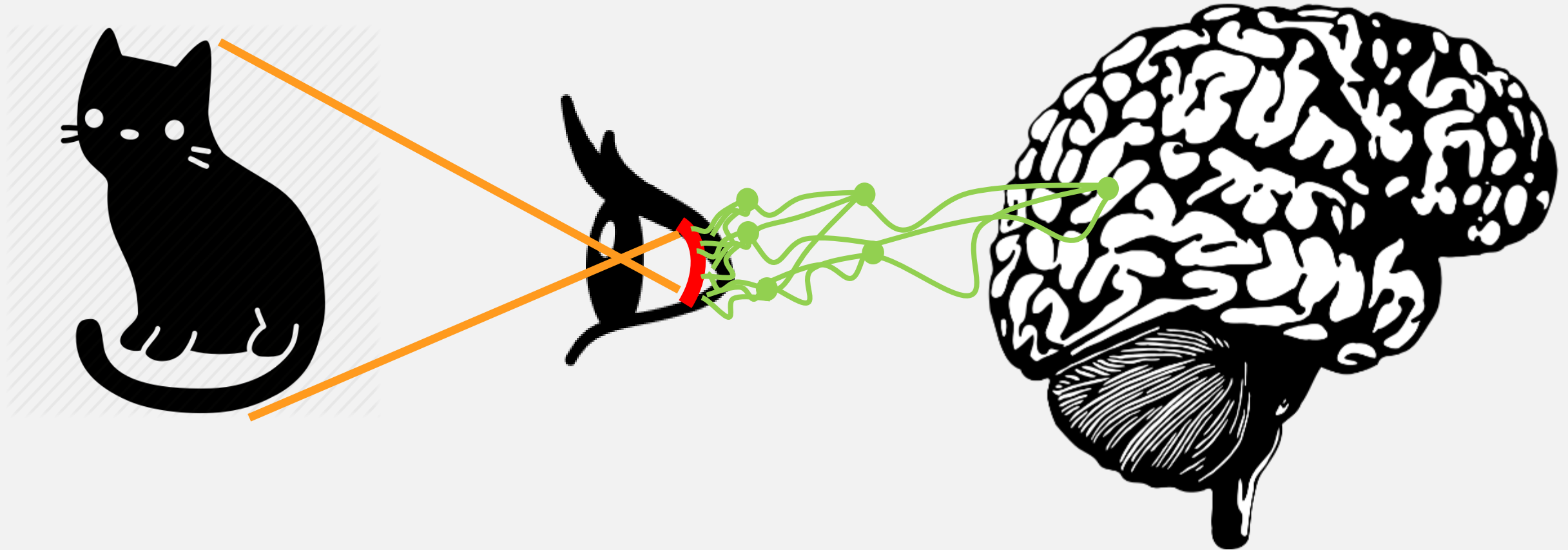
LIFE IS FOR SHARING.

[AI@SCHOOL](#) – [Autonomes Fahren mit dem Calliope mini](#) - Seite 2

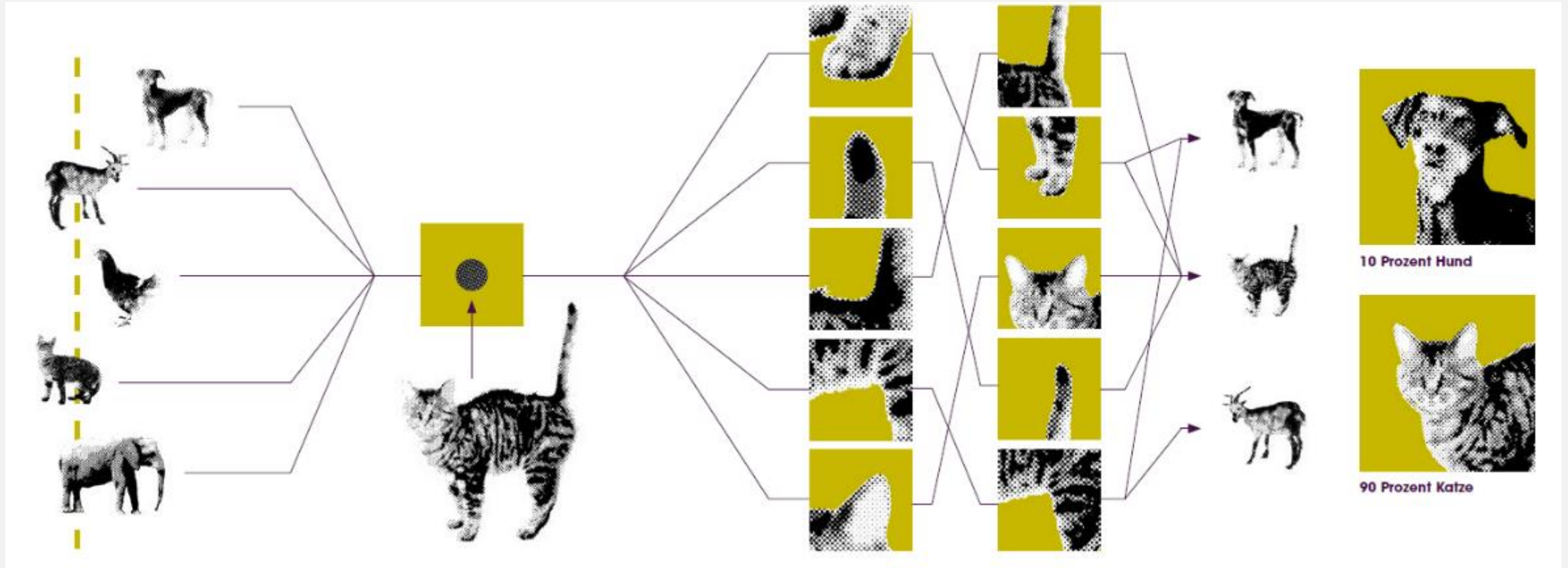


Dieses Material steht unter der
Creative-Commons-Lizenz Namensnennung - Weitergabe
unter gleichen Bedingungen 4.0 International.
Um eine Kopie dieser Lizenz zu sehen, besuchen Sie
<http://creativecommons.org/licenses/by-sa/4.0/>

Wie funktioniert unser Gehirn?



Wie erkennen neuronale Netze Katzen?



Quelle: Bundesministerium für Bildung und Forschung

Beispiele für künstliche Intelligenz



Ein klassisches Programm



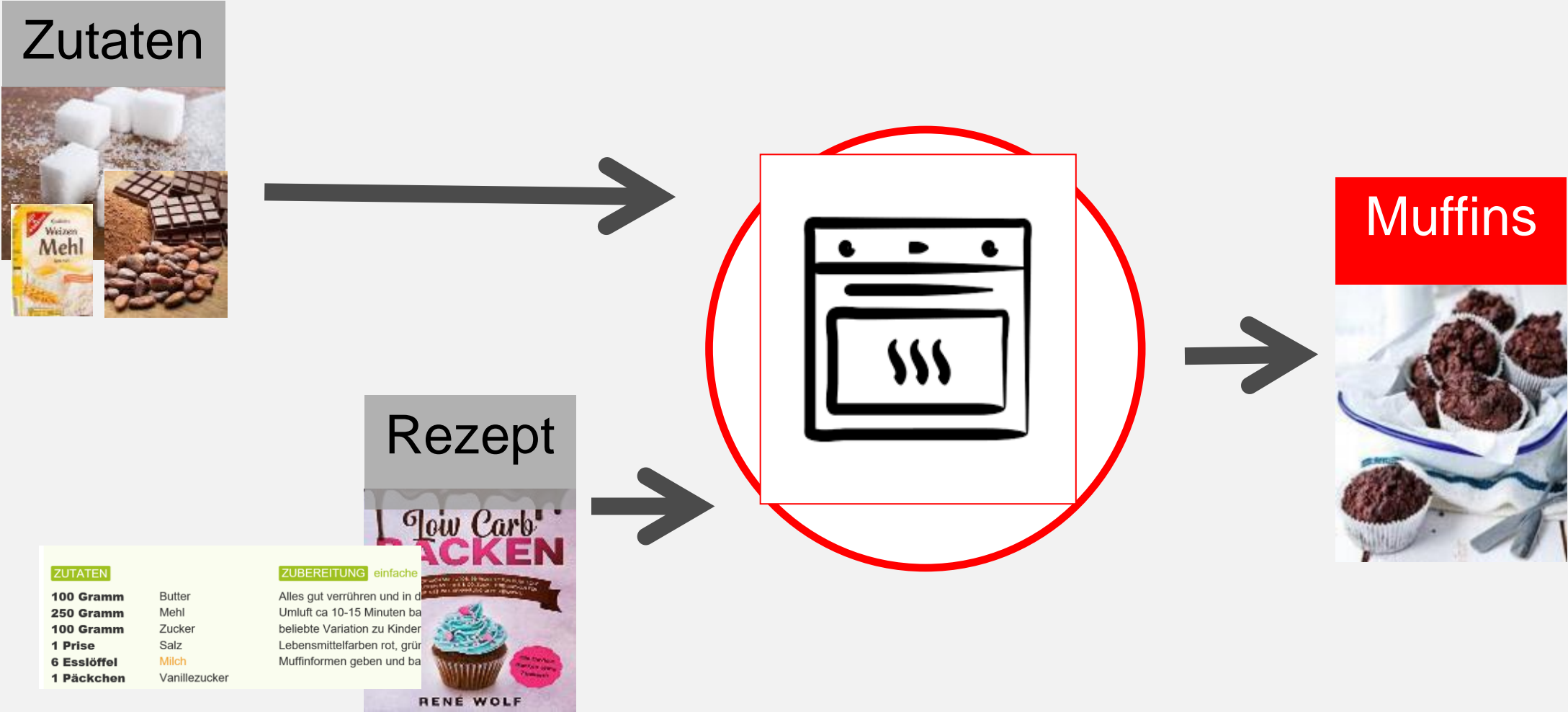
LIFE IS FOR SHARING.

[AI@SCHOOL](#) – [Autonomes Fahren mit dem Calliope mini](#) - Seite 6

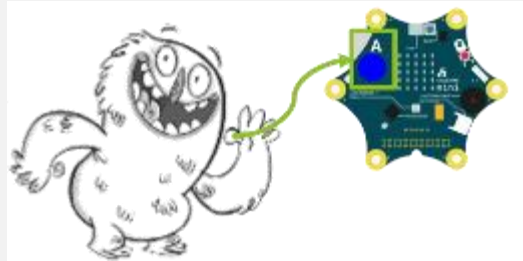


Dieses Material steht unter der Creative-Commons-Lizenz Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International. Um eine Kopie dieser Lizenz zu sehen, besuchen Sie <http://creativecommons.org/licenses/by-sa/4.0/>

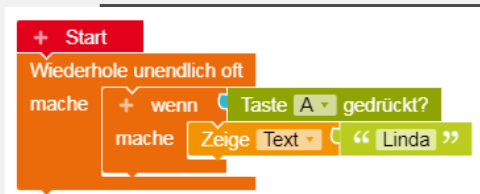
Wie würden wir backen?



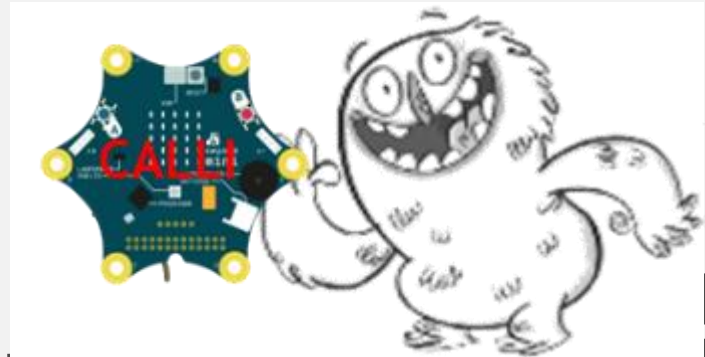
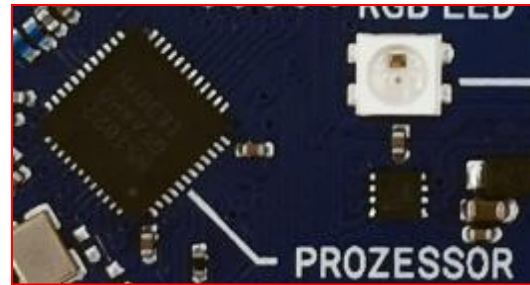
Wie funktioniert Programmierung?



Reale-Daten

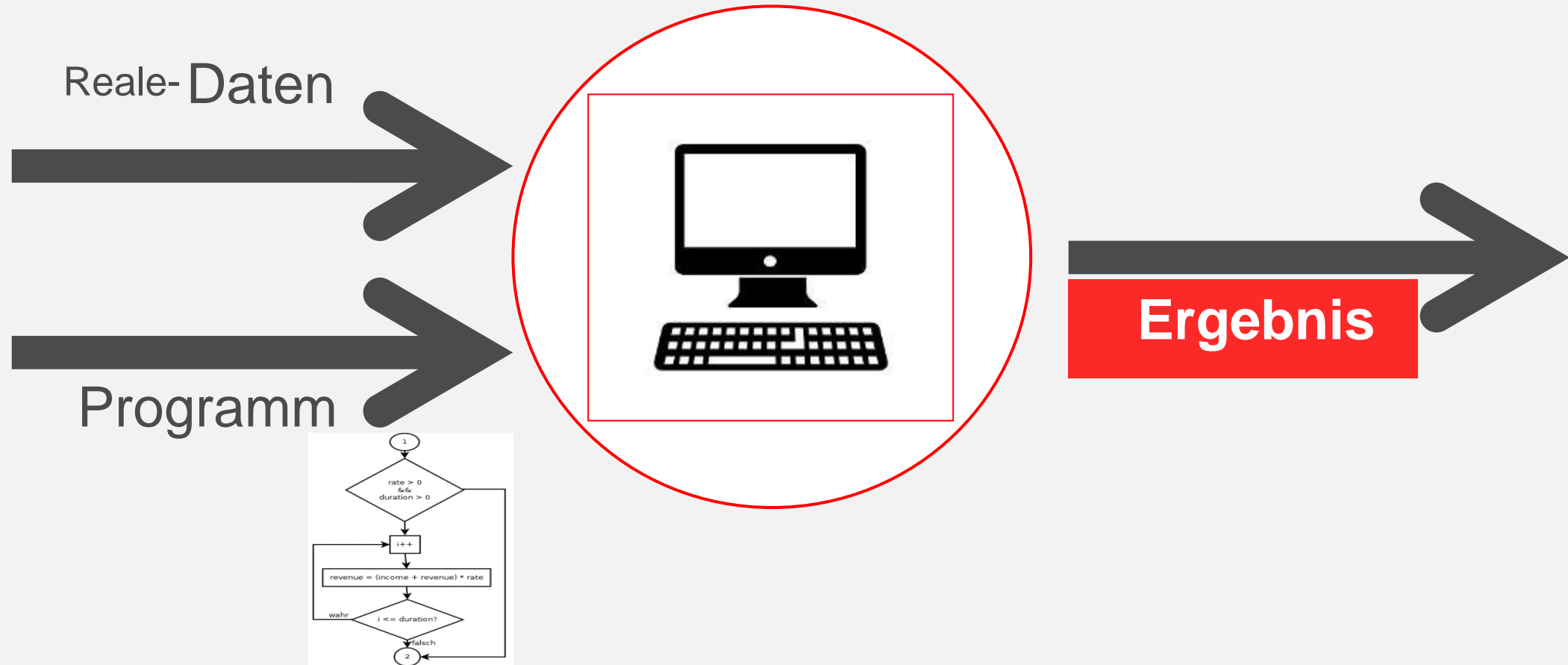


Programm

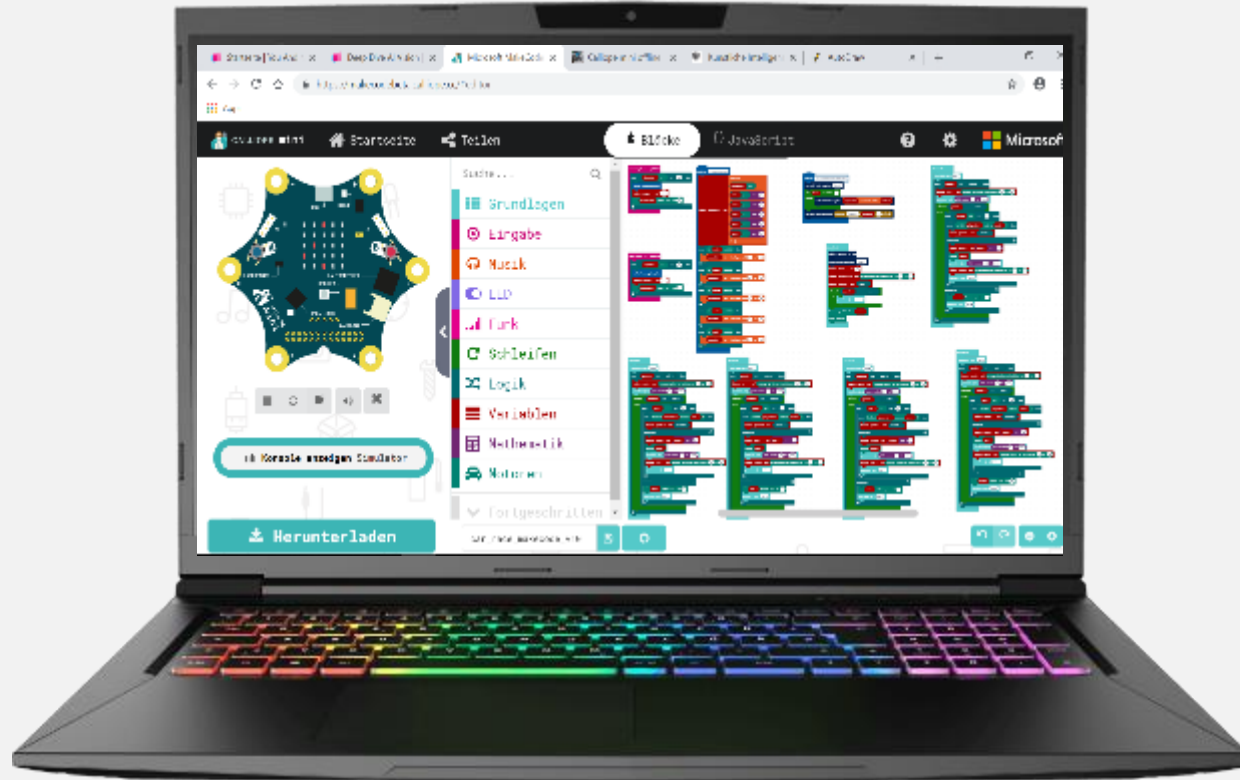


Ergebnis

Wie funktioniert Programmierung?



Was benötigen wir dafür?



Link zum Editor (Makecode): <https://makecode.calliope.cc>

Link zum Editor (Open Roberta): <https://www.open-roberta.org>



LIFE IS FOR SHARING.

AI@SCHOOL – Autonomes Fahren mit dem Calliope mini - Seite 10

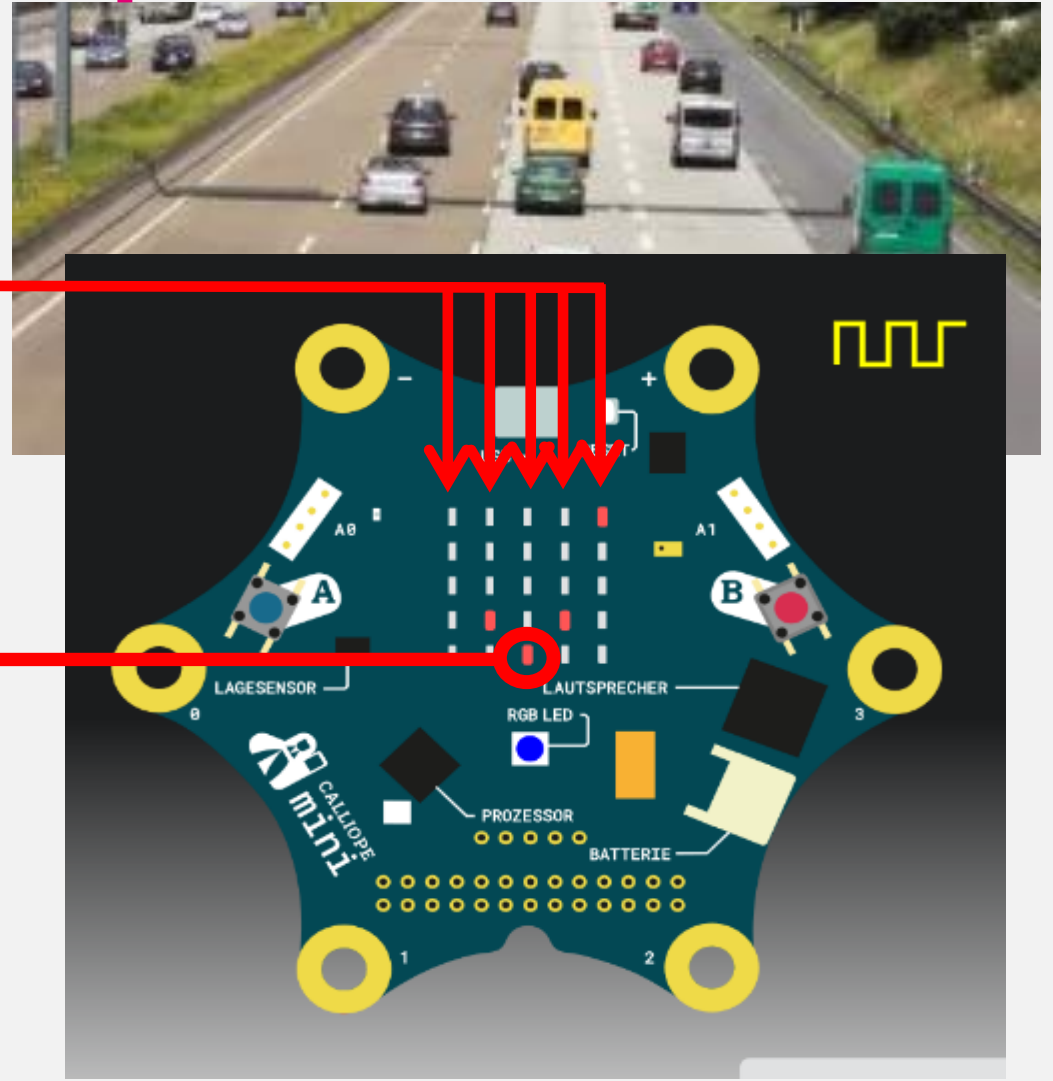


Dieses Material steht unter der Creative-Commons-Lizenz Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International. Um eine Kopie dieser Lizenz zu sehen, besuchen Sie <http://creativecommons.org/licenses/by-sa/4.0/>

Wie funktioniert das Rennspiel?


5 Fahrbahnen mit Autos
die überholt werden müssen

Eigenes Auto




Wie ist das Spiel aufgebaut*? – Die Fahrbahnen





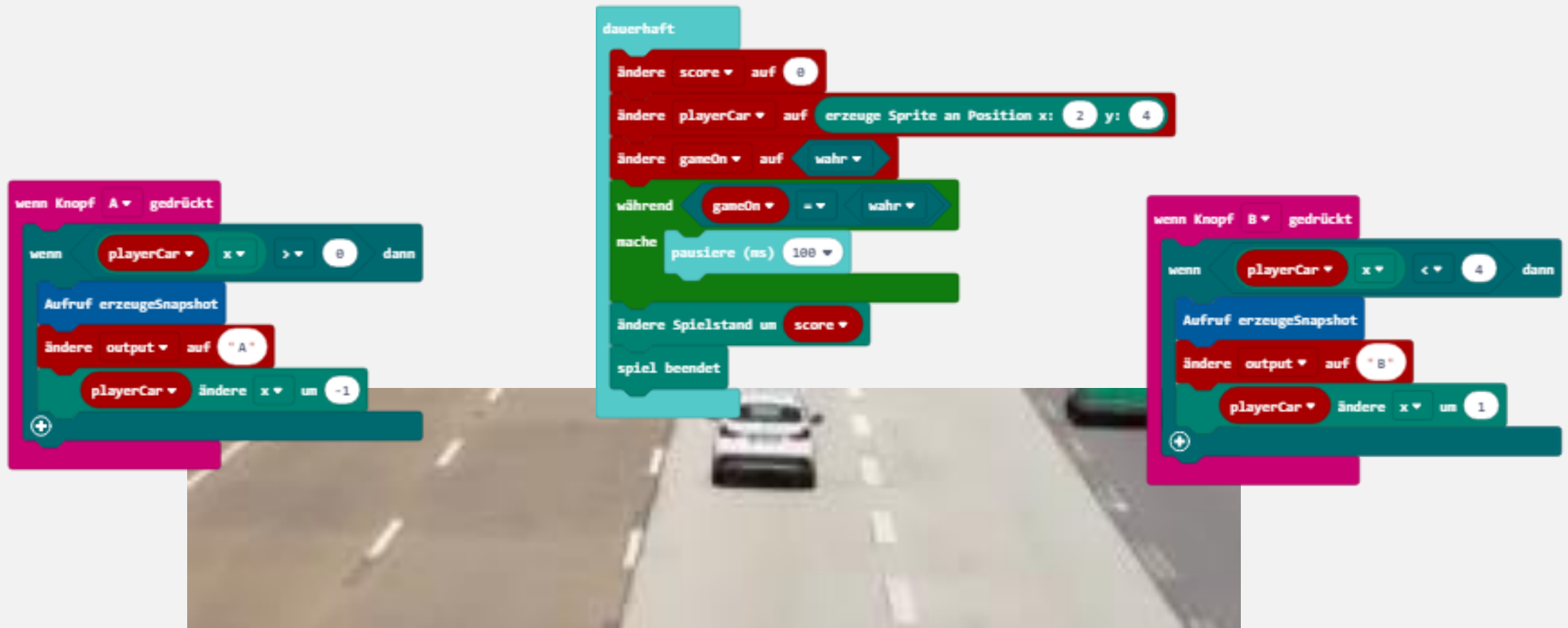








Wie ist das Spiel aufgebaut*? – Eigenes Auto

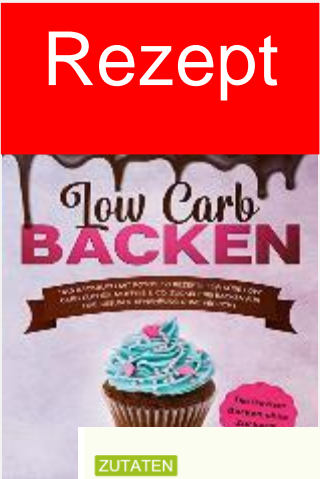
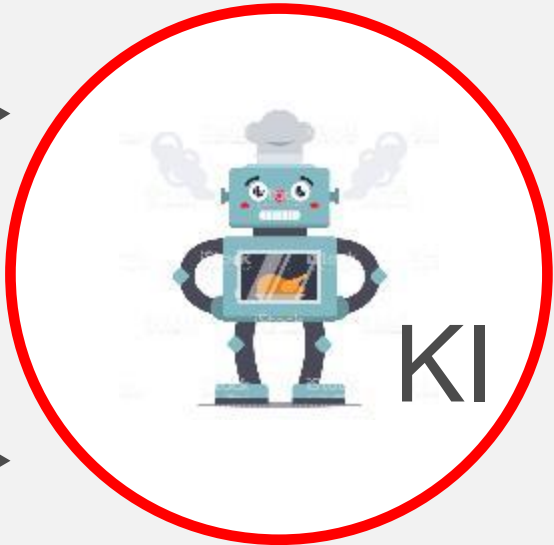


Maschinelles Lernen - KI



Wie würde die Künstliche Intelligenz backen?

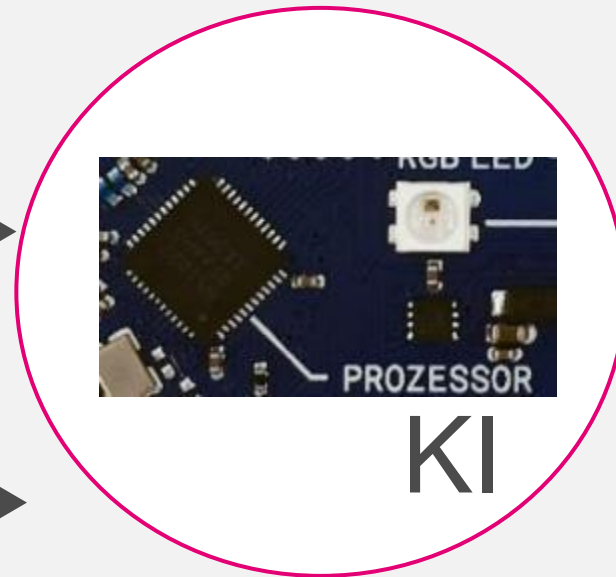
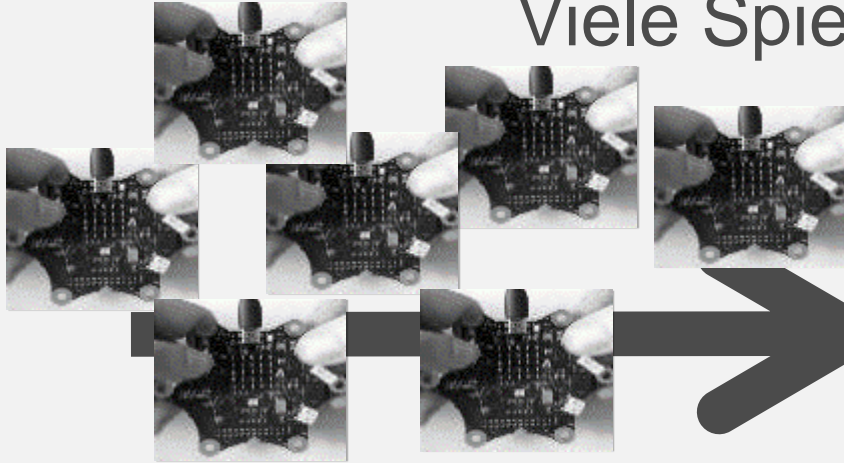
viele Muffins



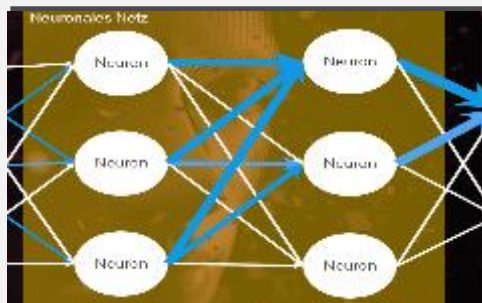
ZUTATEN		ZUBEREITUNG einfache
100 Gramm	Butter	Alles gut verrühren und in d Umluft ca 10-15 Minuten ba beliebte Variation zu Kinder Lebensmittelfarben rot, grün Muffinformen geben und ba
250 Gramm	Mehl	
100 Gramm	Zucker	
1 Prise	Salz	
6 Esslöffel	Milch	
1 Päckchen	Vanillezucker	

Wie funktioniert maschinelles Lernen?

Viele Spieldaten



KI

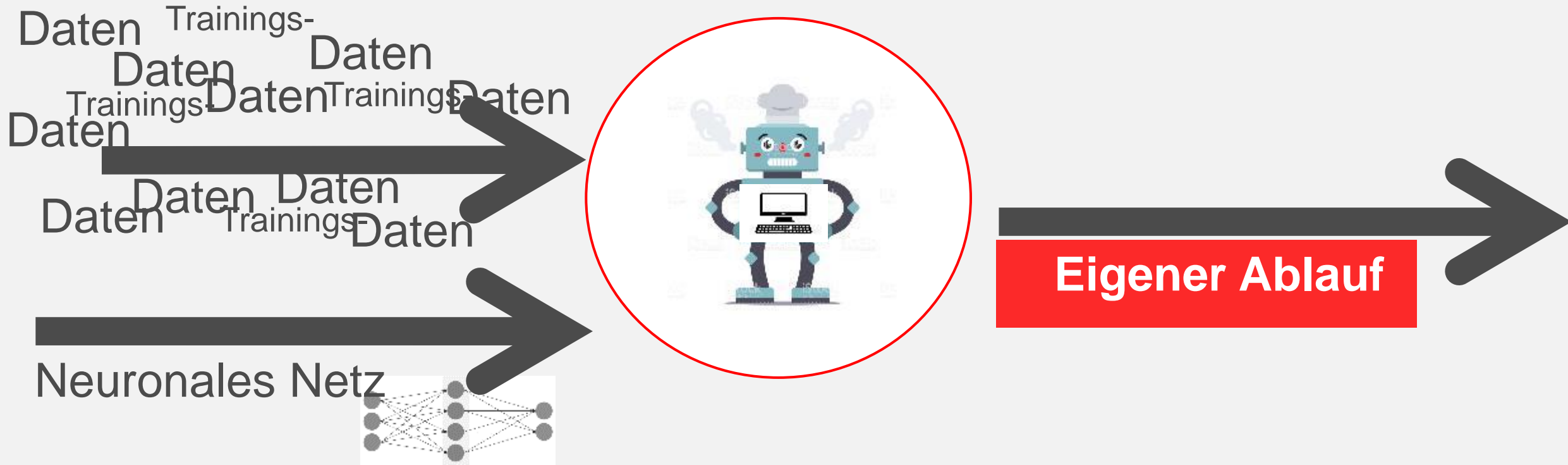


Neuronales Netz



Spielt selber

Wie funktioniert maschinelles Lernen?

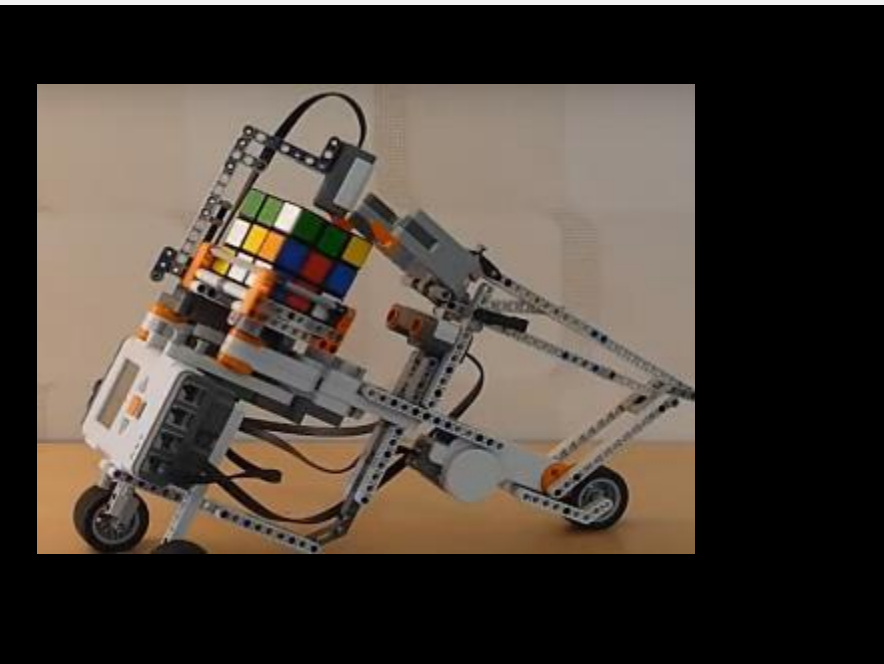
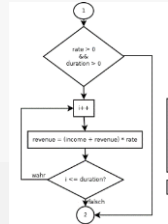


Wo liegt der Unterschied?

Lösen durch Probieren
vom Menschen
(20 - 26 Schritte)

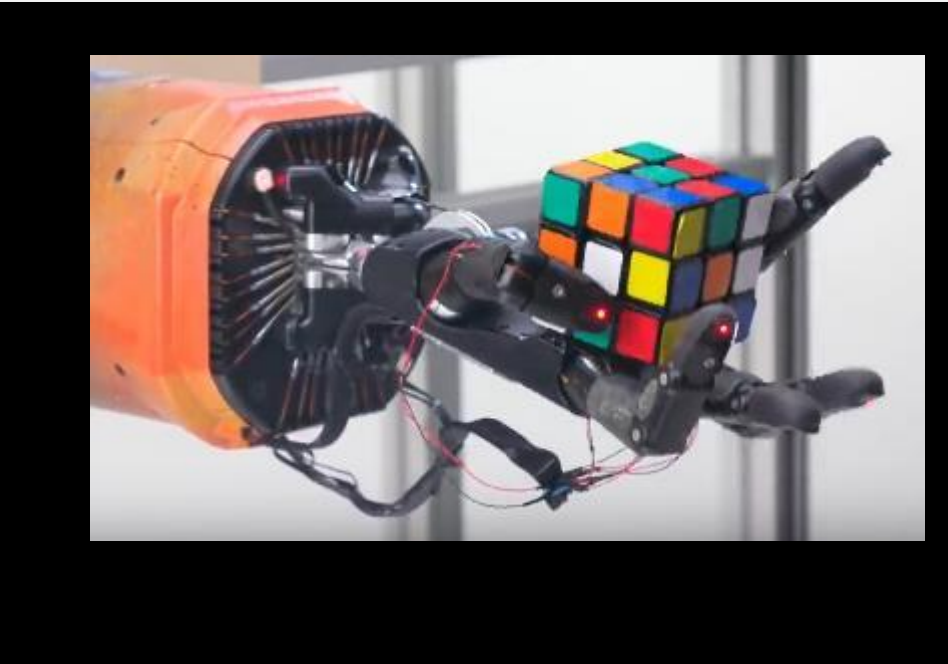
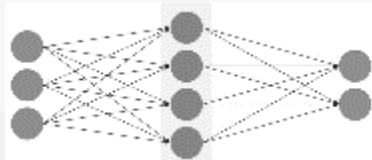


Lösen durch ein fertiges Programm
auf dem Computer
(60 Schritte)



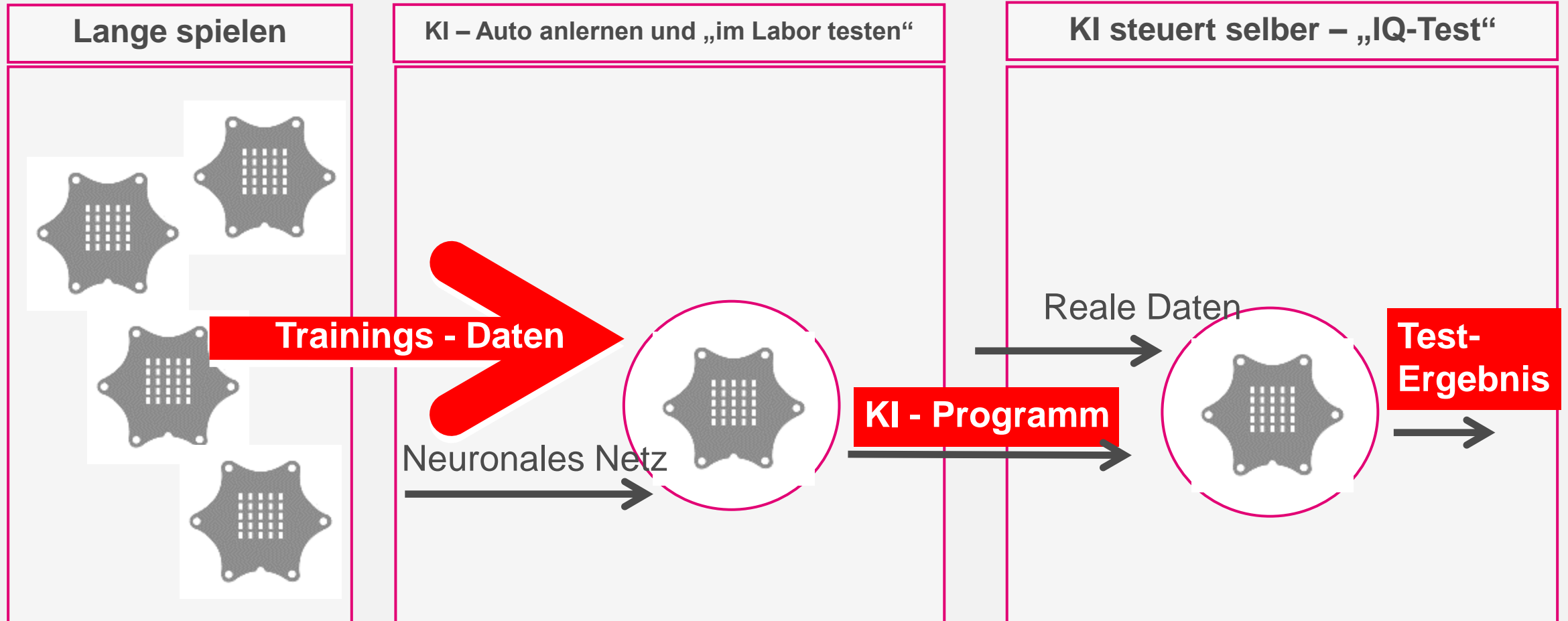
Quelle: <https://www.youtube.com/watch?v=w3f-WyDqOUw>

Lösen durch Trainieren
des Computers
(30 Schritte)



Quelle: <https://youtu.be/kVmp0uGtShk>

Wie lernen wir nun unseren Calliope an?



Welche Spielsituationen gibt es?

L
I
N
K
S

eigenes Auto	Fahrbahn 1	Fahrbahn 2	Fahrbahn 3	Fahrbahn 4	Fahrbahn 5	Taster A/B/nichts
2	1	0	3	5	2	A

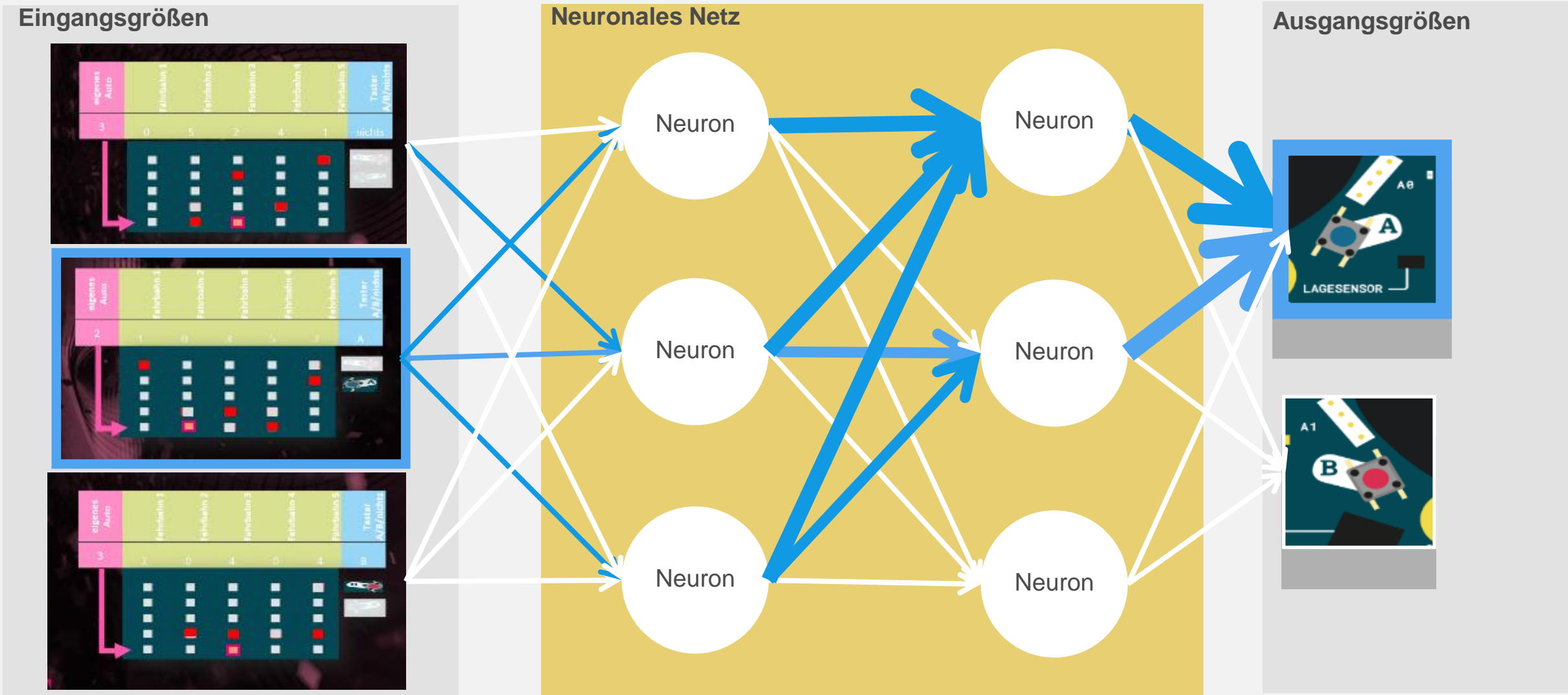
R
E
C
H
T
S

eigenes Auto	Fahrbahn 1	Fahrbahn 2	Fahrbahn 3	Fahrbahn 4	Fahrbahn 5	Taster A/B/nichts
3	2	0	4	0	4	B

N
I
C
H
T
S

eigenes Auto	Fahrbahn 1	Fahrbahn 2	Fahrbahn 3	Fahrbahn 4	Fahrbahn 5	Taster A/B/nichts
3	0	5	2	4	1	nichts

Welche Funktion übernimmt das Neuron?

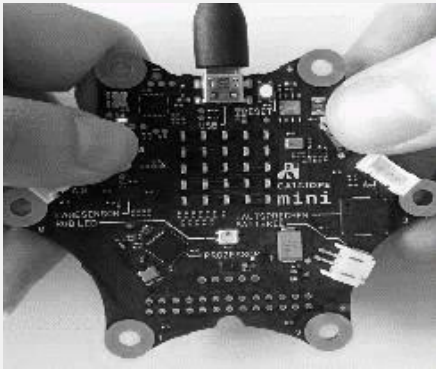


Welches sind die drei Schritte des Anlernens?

1

Lange Spielen um
viele Trainingsdaten
zu erzeugen.

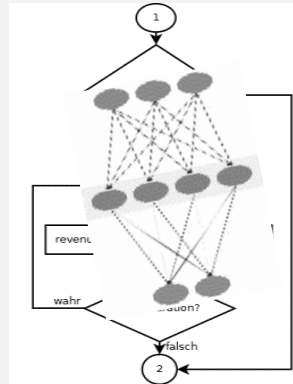
Python-Skript
`ki-datenlogger.py`



2

Eigenes Auto durch
das Neuronale Netz
Anlernen und Testen.

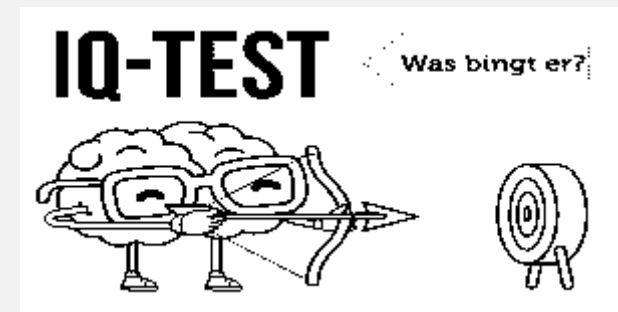
Basisversion: Python-Skript
`ki-trainieren-sklearn.py`
Expertenversion: Orange3
`orange-workflow`



3

künstlich intelligenter
Calliope übernimmt
die Steuerung und
muss „IQ-Test“
bestehen.

Rennspiel-Simulation auf Desktop-PC
`ki-rennspiel.py`
„Hirntransplantation“ ;-) auf Calliope
`iq-test-erstellen.py`

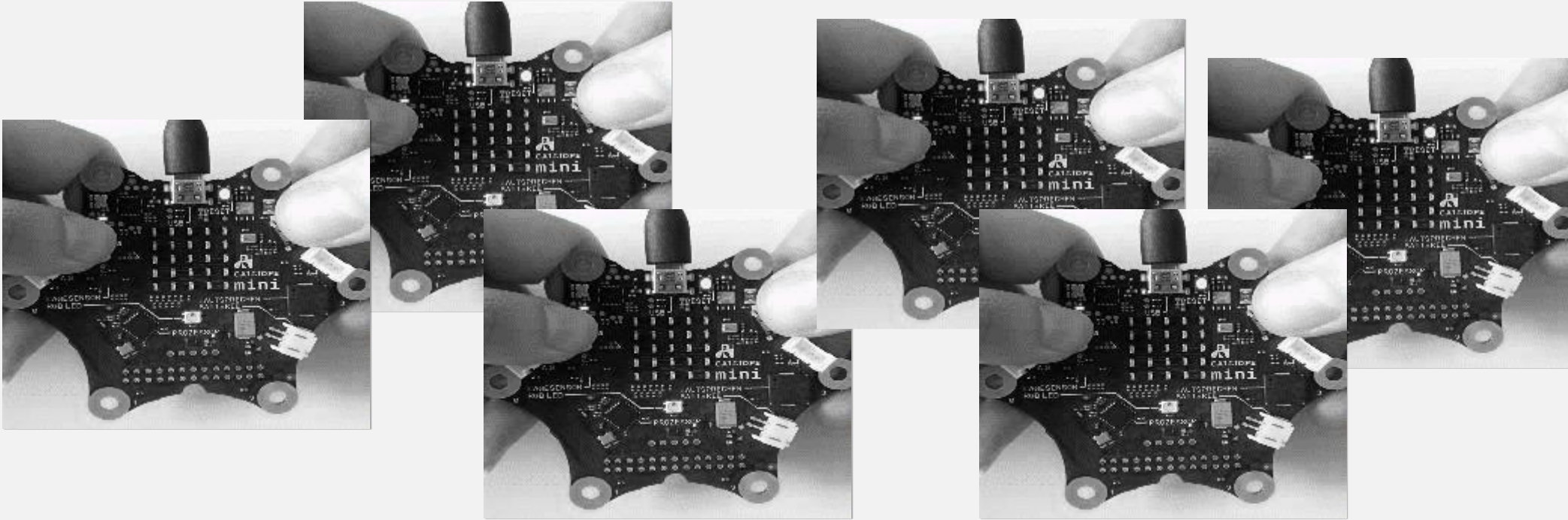


Trainingsphase

ki-datenlogger.py

1

Lange Spielen um viele Trainingsdaten zu erzeugen.



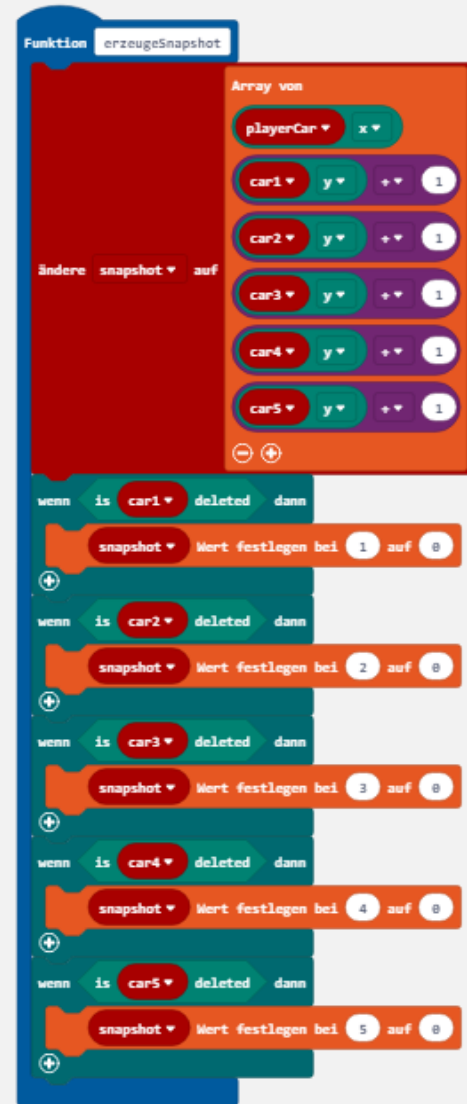
LIFE IS FOR SHARING.

[AI@SCHOOL](#) – [Autonomes Fahren mit dem Calliope mini](#) - Seite 23

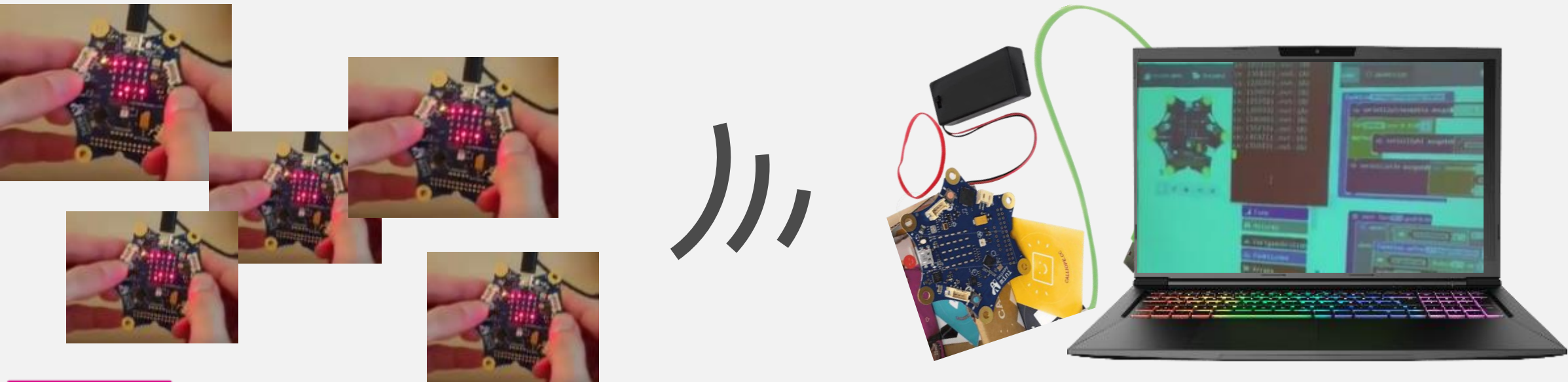


Dieses Material steht unter der Creative-Commons-Lizenz Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International. Um eine Kopie dieser Lizenz zu sehen, besuchen Sie <http://creativecommons.org/licenses/by-sa/4.0/>

Wie ist das Spiel aufgebaut*? – Trainings-Daten



Wie ist das Spiel aufgebaut*? - Daten sammeln



```
when internet connection received
  when received packet signal strength >= 50 then
    when button pressed then
      add to list: button pressed
    else
      add to list: button not pressed
  end when
end when

when start button pressed then
  set output string to: 1
  set serial output to: 2
end when

function create training dataset
  set output string to: 1
  for index from 0 to 5
    when button pressed then
      add to list: output string + button pressed + index
    else
      add to list: output string + button not pressed + index
    end when
  end for
  set output string to: 1
  output: 1
end function
```

Für Einzelspielmodus wird der Trainingsdatensatz direkt per USB übertragen

```
when start button pressed then
  set output string to: 1
  set serial output to: 2
  send data: 1 via USB
end when

when button pressed then
  send data: 1 via USB
end when
```

```

(ki-calliope-rennspiel-basis) goethe@ubuntu:~/github/telekom/ki-in-schulen/Calliope-Rennspiel/Python$ python ki-datenlogger.py /dev/ttyACM0
Nutze COM-Port: /dev/ttyACM0
1,5,4,3,0,0,x
Nutzdaten erkannt - Übernahme in DataFrame
1,0,5,4,0,0,x
Nutzdaten erkannt - Übernahme in DataFrame
Rennspiel-Funkgruppe1
Nutzdaten erkannt - Übernahme in DataFrame
-----
Nutzdaten erkannt - Übernahme in DataFrame
2,2,1,1,1,1,x
Nutzdaten erkannt - Übernahme in DataFrame
2,3,1,1,1,1,x
Nutzdaten erkannt - Übernahme in DataFrame
2,4,1,1,1,1,x
...
Tastendruck erkannt! Beende Datensammlung und speichere CSV.
Bitte ggf. nochmal Calliope resettet, um Vorgang abzuschließen.
Oder 15 Sekunden warten, dann geschieht dies automatisch.
Rennspiel-Funkgruppe1
Nutzdaten erkannt - Übernahme in DataFrame
  PlayerPos Car1Pos Car2Pos Car3Pos Car4Pos Car5Pos Action
0          1         5         4         3         0         0      x
1          1         0         5         4         0         0      x
2          R         n         s         i         l         F      n
3          -         -         -         -         -         -      -
4          2         2         1         1         1         1      x
5          2         3         1         1         1         1      x
6          2         4         1         1         1         1      x
7          2         5         1         1         1         1      x
8          2         0         1         1         2         1      x
9          2         0         1         1         2         1      A
10         1         0         3         1         4         1      x
11         1         0         4         1         5         1      x
12         1         0         5         2         0         2      x
13         R         n         s         i         l         F      n
Trainingsdaten gespeichert in Datei: ./csv-rohdaten/ki-rennspiel-log-20210316142033.csv

```

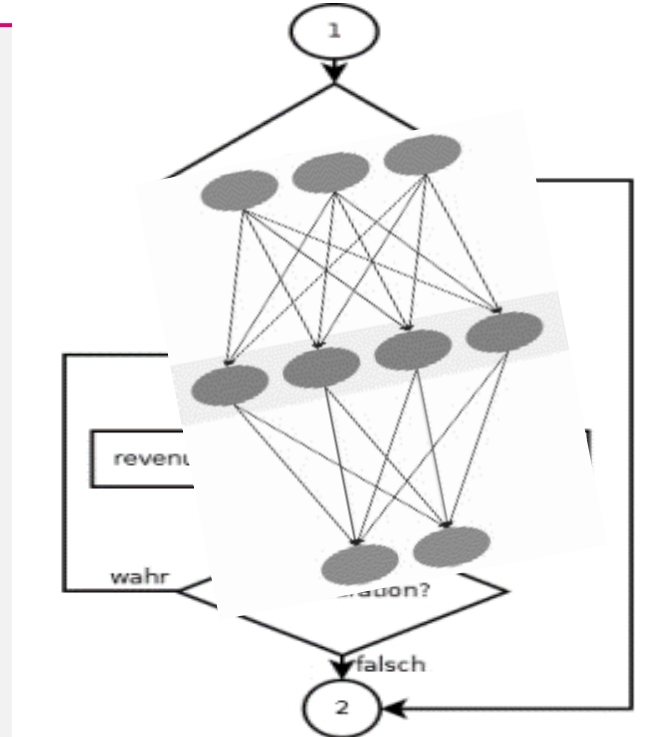
Beliebiger Tastendruck beendet Datensammlung

Trainingsdaten zur Nutzung in Schritt 2
„KI anlernen“ mit **ki-trainieren-sklearn.py**



2

Eigenes Auto durch das Neuronale Netz Anlernen und Testen.



Von **ki-datenlogger.py**
erzeugte Trainingsdaten

```
(ki-calliope-rennspiel-basis) goethe@ubuntu:~/github/telekom/ki-in-schulen/Calliope-Rennspiel/Python$ python ki-trainieren-sklearn.py csv-rohdaten/120minutes-1.csv 7,5,3
Hidden Layers: [7, 5, 3]
Rohdaten Liste der Aktionen (sollten nur x,A,B sein!): ['x' 'B' 'A']
Rohdaten Anzahl x: 6892
Rohdaten Anzahl A: 1092
Rohdaten Anzahl B: 1047
Beginne Trainingsvorgang.
Training beendet.
Erreichte Modellgenauigkeit (Trainingsdaten): 0.8318106312292359
Erreichte Modellgenauigkeit (Testdaten) : 0.8334255672385169
Pickle-Datei des trainierten ML-Modells gespeichert.
Dateiname: ./modelle/sklearn-py-modell-20210316141323.pkcls
JSON-Datei des trainierten ML-Modells gespeichert.
Dateiname: ./modelle/sklearn-py-modell-20210316141323.json
```

Angelerntes neuronales Netzwerk
zur Nutzung in Schritt 3 „IQ-Test“ mit **ki-rennspiel.py**

Angelerntes neuronales Netzwerk
zur Nutzung in Schritt 3 „IQ-Test“ mit **iq-test-erstellen.py**



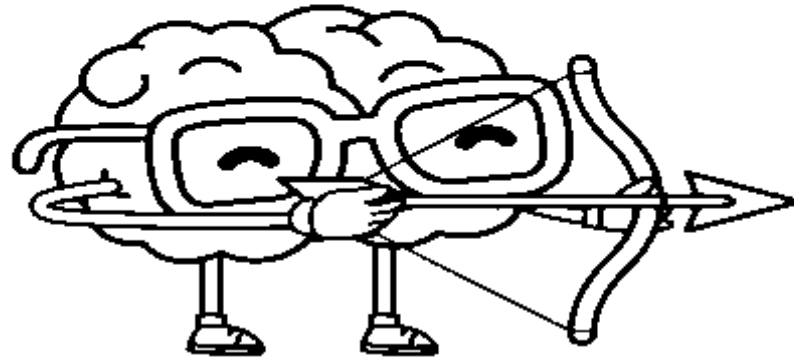
3

Rennspiel-Simulation auf Desktop-PC

künstlich intelligenter Calliope übernimmt die Steuerung und muss „IQ-Test“ bestehen.

IQ-TEST

Was bingt er?

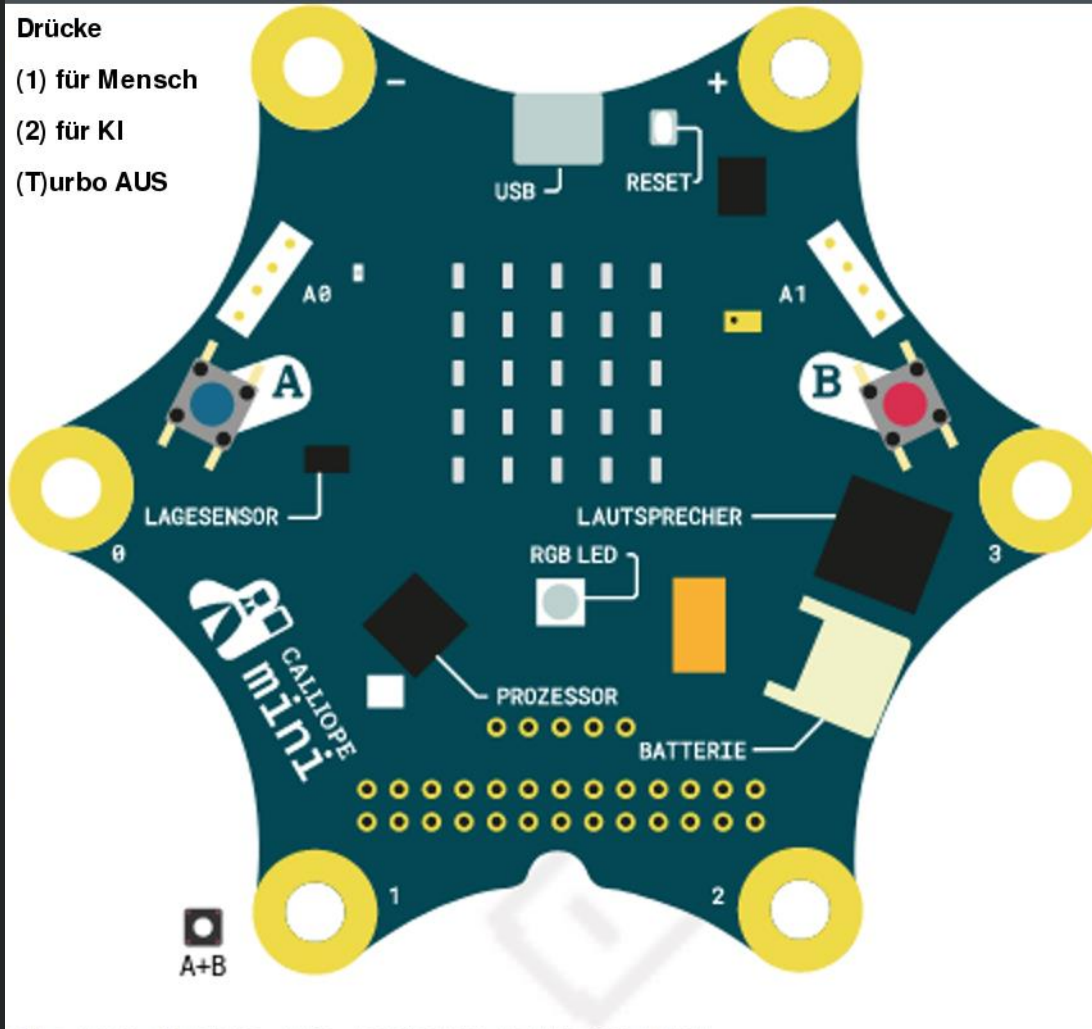


Drücke

(1) für Mensch

(2) für KI

(T)urbo AUS

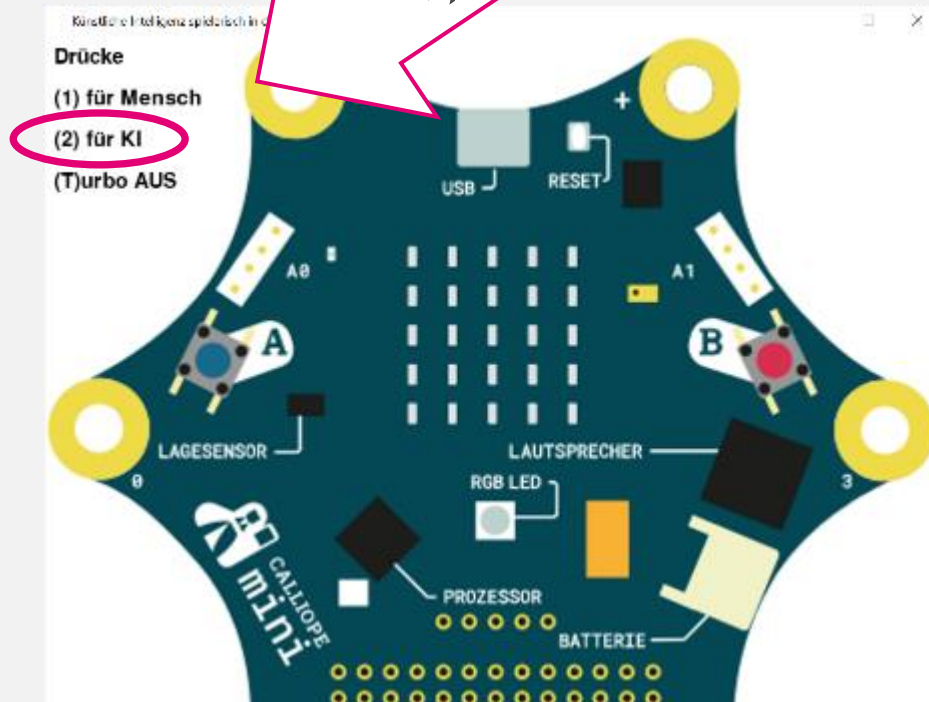


Von **ki-trainieren-sklearn.py**
angelerntes neuronales Netzwerk

```
(ki-calliope-rennspiel-basis) goethe@ubuntu:~/github/telekom/ki-in-schulen/Calliope-Rennspiel/Python$ python ki-rennspiel.py sklearn modelle/sklearn-py-modell-20210316141204.pkcls
pygame 1.9.4
Hello from the pygame community. https://www.pygame.org/contribute.html
Orange nicht verfügbar.
SciKit-Learn Backend, lade ML-Modell modelle/sklearn-py-modell-20210316141204.pkcls
█
```

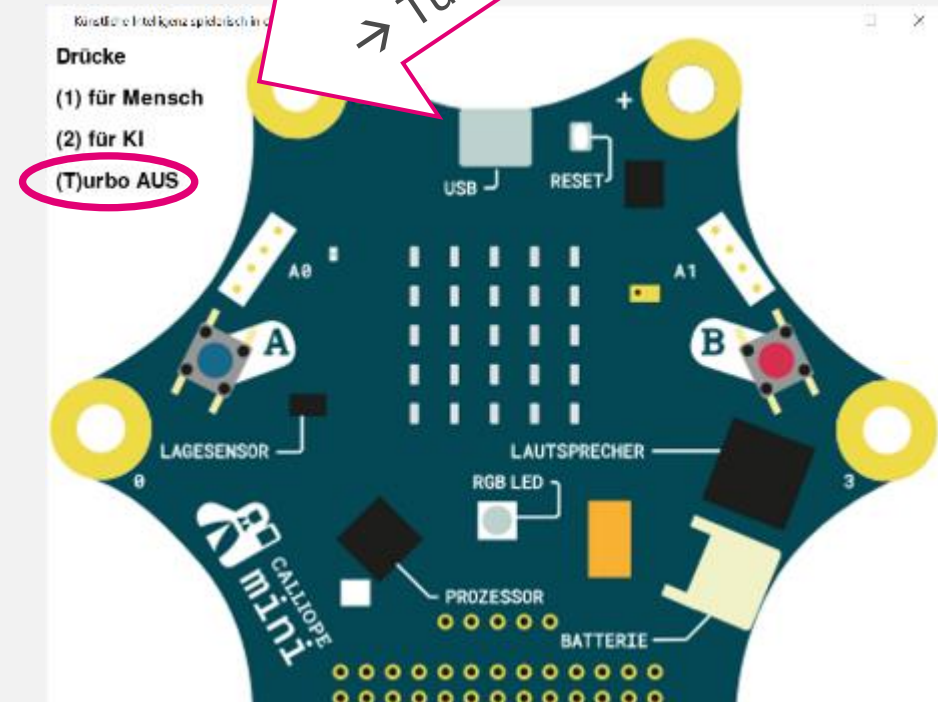
„Beim Fahren
Zuschauen“

Zahl 2 Drücken
→ KI steuert



„IQ-Test“

Taste T Drücken
→ Turbo einschalten



Beispiel-IQ: 10 Minuten Trainingsdaten



Beispiel-IQ: 60 Minuten Trainingsdaten

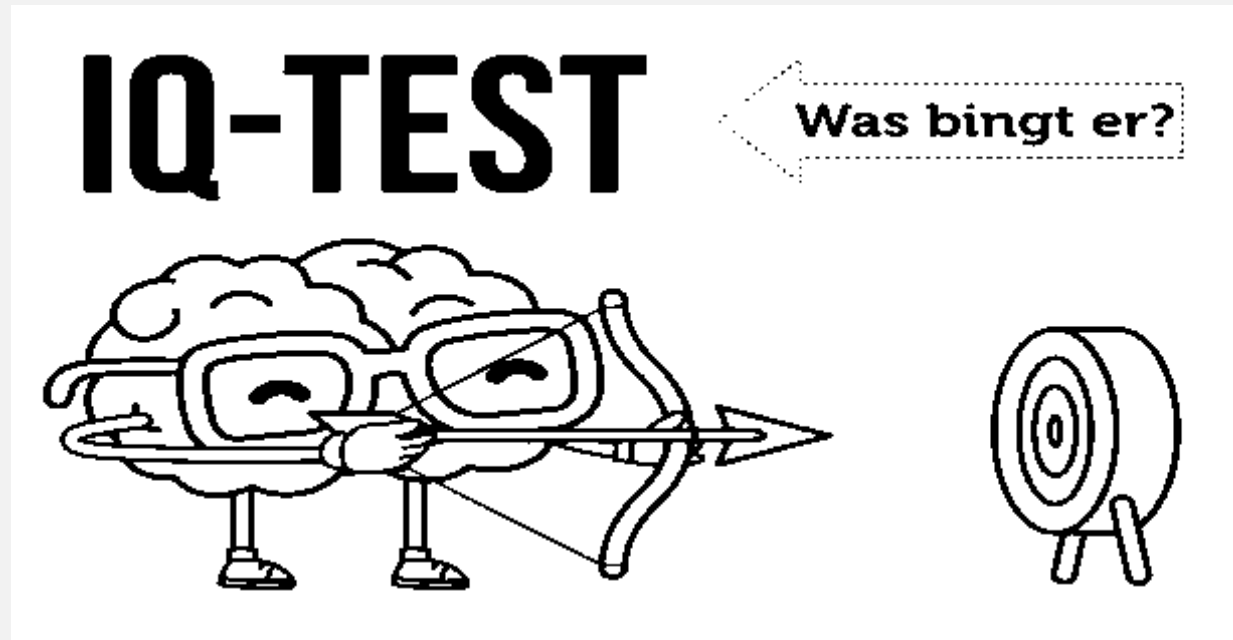


Beispiel-IQ: 120 Minuten Trainingsdaten



3 „Hirntransplantation“ ;-) auf Calliope

künstlich intelligenter Calliope übernimmt die Steuerung und muss „IQ-Test“ bestehen.



KI -Steuerungsphase

iq-test-erstellen.py

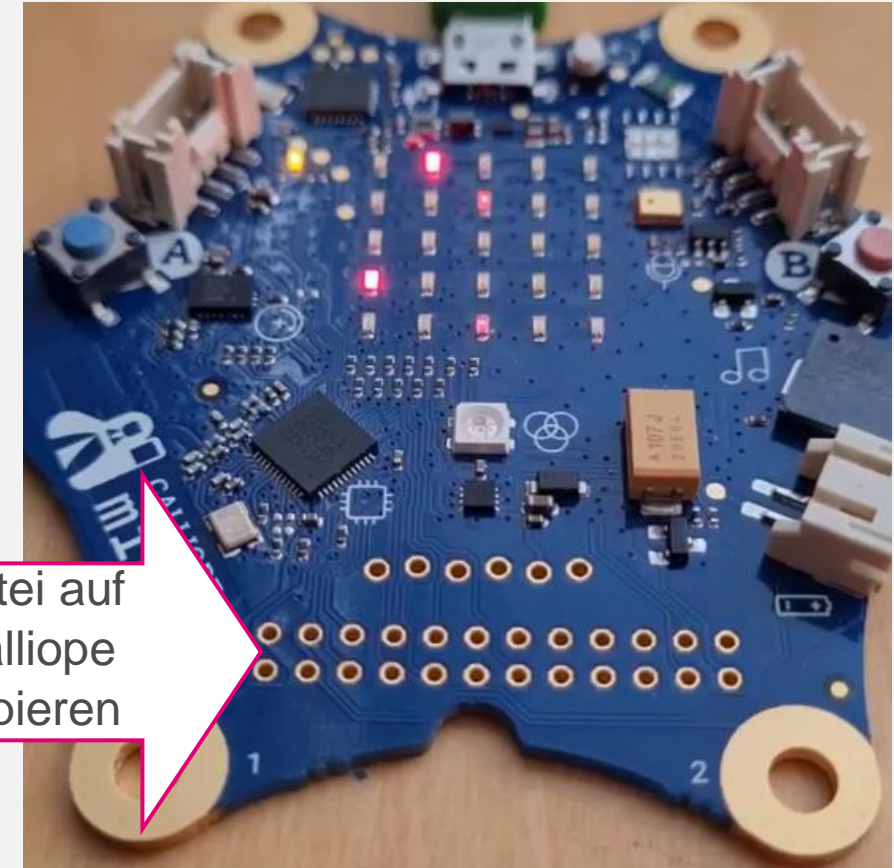
```
(ki-calliope-rennspiel-basis) goethe@ubuntu:~/github/telekom/ki-in-schulen/Calliope-Rennspiel/Python/iq-test-calliope$  
python iq-test-erstellen.py -r ../modelle/sklearn-py-modell-20210302220807.json
```

Von **ki-trainieren-sklearn.py**
angelerntes neuronales Netzwerk

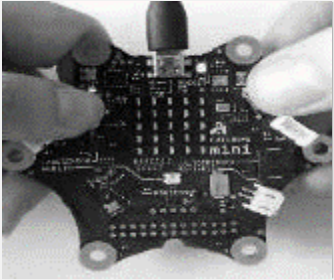
Skript **iq-test-erstellen.py**
erzeugt Datei **iq-test-calliope.hex**

```
ls -al  
total 1100  
drwxrwxr-x 2 goethe goethe 4096 Mar 31 16:27  
drwxrwxr-x 8 goethe goethe 4096 Mar 3 18:01  
-rw-rw-r-- 1 goethe goethe 340 Mar 3 18:00 DME.md  
-rw-rw-r-- 1 goethe goethe 550084 Mar 16 22:45 base.hex  
-rw-rw-r-- 1 goethe goethe 550084 Mar 30 15:33 iq-test-calliope.hex  
-rw-rw-r-- 1 goethe goethe 4840 Mar 16 22:45 iq-test-erstellen.py
```

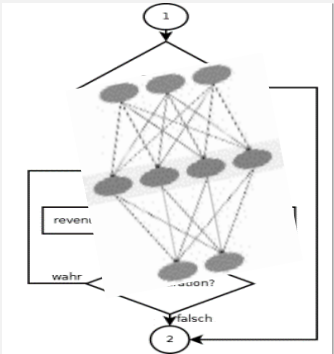
Datei auf
Calliope
kopieren



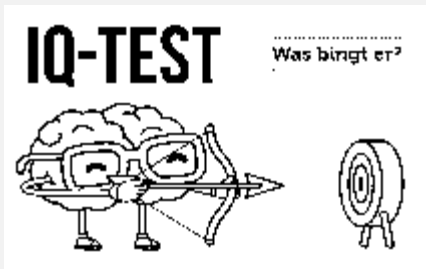
Warum heißt es nun „maschinelles Lernen“?



1. Wir haben viele Trainings-**Daten** durch Spielen erzeugt.



2. Wir haben nun den **Calliope angelernt** und getestet.



3. Wir haben den Calliope alleine spielen lassen und **wissen nun, wie gut er ist.**