

KI Aufgaben in Python

Torben Friedrich Görner

January 2023

1 Einleitung

Heute wird es spannend aber auch etwas knifflig. Keine Sorge, du musst nicht jede Zeile im Code zu 100% verstehen. Wir nutzen eine große Bibliothek (**Tensorflow**) für neuronale Netze und Trainingsalgorithmen. Du kannst unser vorgehen mit dem kaufen einer fertigen TK-Pizza vergleichen. Wir haben uns ein fast fertiges Produkt geholt und müssen es jetzt nur noch für unsere Zwecke anpassen. Wir nehmen also die Pizza aus der Verpackung heizen den Ofen vor und belegen vielleicht noch ein paar Zutaten dazu.

2 Aufgabe zur Mustererkennung

Gegeben sei eine Zahlenreihe $X = 2, 4, 6$ und eine Zahlenreihe $Y = 4, 8, 12$. Wenn ich dir nun die Zahl 10 als weiters x gebe, welchen y Wert erwartest du dann ? ... Vermutlich die 20, da du erkannt hast, das 4 das doppelte von 2 ist, 8 das doppelte von 4, ... und entsprechend 20 das doppelte von 10. Kann eine KI auch solche vorhersagen treffen ?

```
import tensorflow as tf
from tensorflow import keras

model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1])])

model.compile(optimizer='sgd', loss='mean_squared_error')

xs=[2, 4, 6] # Zahlenreihe 1 (x für die wir jeweils ein y zuordnen)
ys=[4, 8, 12] # Zahlenreihe 2 (y das jeweils einem x zugeordnet ist)

model.fit(xs, ys, epochs=500)

print(model.predict([10]))
```

2.1 Teste den gegebenen Code

Tensorflow ist die fertig Pizza, die wir nun zubelegen und aufbacken wollen. Keras ist eine Open Source Deep-Learning-Bibliothek die wir ebenfalls nutzen wollen. Zunächst definieren wir unser neuronales Netz. Es bekommt einen Eingabewert (Die Zahl x für die wir das y vorhersagen wollen). Dann wird das neuronale Netz compiliert. Wir definieren einen *Optimizer* (sgd = stochastische Gradientenabstieg. Ein iteratives Verfahren zur Optimierung einer Zielfunktion mit geeigneten Glattheitseigenschaften (Keine Sorge, dass musst du nicht im Detail verstehen. Stell dir vor, dass es definiert wie trainiert werden soll.)) und einen *loss* (der Fehler zum messen wie 'schlecht' unser neuronales Netz ist). Nun definieren wir im Code die beiden Zahlenreihen zum trainieren. Dann wird das Training gestartet, die Epochen geben dabei die Iterationen des Trainings an. Wie im Studio wenn du Hanteln drückst (10 Wiederholungen, 20 oder in unserem Fall 1000). Dann können wir die erste Vorhersage starten. Wir testen für den Wert 10.

2.2 Verbesserungen

Unser Modell ist nicht perfekt. Es hat durch Zufall verbindungen zwischen den Zahlen gesucht und der loss (unser Fehler ist nicht 0). Wie können wir unser neuronales Netz verbessern ?

2.3 Neues Muster

Denke dir ein eigenes Muster aus und dazu Trainingsdaten für X und Y . Trainiere und Teste dein neuronales Netz.