

Übung

VCAT1 – Visual Computing: Maschinelles Lernen

7. Übungsblatt – Regularisierung

1. Aufgabe

- Laden Sie sich das Python-Skript *Bildklassifikation_flowers_simple.py* ' aus dem Ordner 'Supplemental Material' im Moodle-Kurs herunter.
- Öffnen Sie das Skript und versuchen Sie den Code im Detail nachzuvollziehen:
 - Wie viele Layer hat dieses Netzwerk?
 - Welche Aktivierungsfunktion wird verwendet?
 - Wie groß sind die Feature Maps nach jeweils dem 1., 2. und 3. Pooling-Layer?
 - Wie viele Trainings-, Validierung- und Testbilder werden verwendet?
- Führen Sie das Skript aus und speichern Sie die Kurvenverläufe ab.
- Interpretieren Sie die Kurvenverläufe.

2. Aufgabe

- Speichern Sie das Python-Skript *Bildklassifikation_flowers_simple.py* ' als *Bildklassifikation_flowers_advanced.py* ' ab.
- Fügen Sie einen Dropout-Layer im Modell hinzu. Setzen Sie die Dropout-Wahrscheinlichkeit einmal auf 0,2 und einmal auf 0,5 und vergleichen Sie die Ergebnisse der Kurvenverläufe untereinander und mit den Kurvenverläufen aus Aufgabe 1.
- Kommentieren Sie den Dropout-Layer vorübergehend aus und fügen Sie eine Datenerweiterung mit dem folgenden Code durch:

```
# Datenerweiterung (data augmentation)
data_augmentation = keras.Sequential(
    [
        layers.experimental.preprocessing.RandomFlip("horizontal", input_shape=(img_height, img_width, 3)),
        layers.experimental.preprocessing.RandomRotation(0.1),
        layers.experimental.preprocessing.RandomZoom(0.1),
    ]
)
```

Visualisieren Sie die Datenerweiterung wie folgt:

```
# Visualisierung der Datenerweiterung
plt.figure(figsize=(10, 10))
for images, _ in train_ds.take(1):
    for i in range(9):
        augmented_images = data_augmentation(images)
        ax = plt.subplot(3, 3, i + 1)
        plt.imshow(augmented_images[0].numpy().astype("uint8"))
        plt.axis("off")
```

Vergleichen Sie die Ergebnisse der Kurvenverläufe mit den Ergebnissen aus Aufgabe 2b).

- Verwenden Sie nun beide Regularisierungen, d.h. Dropout und Datenerweiterung und vergleichen Sie wiederum die Kurvenverläufe.
- Testen Sie nun Ihr Modell mit neuen Testdaten. Verwenden Sie z.B. hierfür den folgenden Code:

```
# Bildklassifikation
sunflower_url = "https://storage.googleapis.com/download.tensorflow.org/example_images/592px-Red_sunflower.jpg"
sunflower_path = tf.keras.utils.get_file('Red_sunflower', origin=sunflower_url)

img = keras.preprocessing.image.load_img(
    sunflower_path, target_size=(img_height, img_width)
)
img_array = keras.preprocessing.image.img_to_array(img)
img_array = tf.expand_dims(img_array, 0) # Create a batch

predictions = model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```