Selamat! Anda lulus!

Nilai diterima 100% UNTUK LULUS 80% atau lebih tinggi

Ambil kembali tugas di 7h 48m

Pergi ke item berikutnya

Week 4 Quiz

Nilai Kiriman Terbaru 100%

1. Which Devices support TensorFlow Lite for Inference? (Check all that apply) 1 / 1 poin

- [ ] RISC
- [x] Coral

Benar

- [x] Sparkfun Edge

Benar

- [x] Raspberry Pi

Benar

2. With a Raspberry Pi, how can you use TensorFlow? 1 / 1 poin

- (•) Inference and Training
- ( ) It doesn't work on Pi
- ( ) Inference Only
- ( ) Training Only

Benar

3. If you only want to do inference on a Pi, what's the best way? 1 / 1 poin

- ( ) Install the full TensorFlow with Pip install
- ( ) Compile all of TensorFlow from Source and run it
- ( ) Do nothing, the Pi base image has TensorFlow in it
- (•) Install the standalone interpreter using pip

Benar

4. When using ImageNet on a Raspberry Pi for Image Classification, how many classes are supported? 1 / 1 poin

- (•) 1000
- ( ) 800
- ( ) 500
- ( ) 100

Benar

5. How do you initialize the standalone interpreter in Python? 1 / 1 poin

- tf.lite.load(saved_model)
- (●) tf.lite.Interpreter(directory_of_lite_Model)
- tf.lite.load(lite_model)
- tf.lite.Interpreter(directory_of_saved_model)

Benar

6. How do you get the input tensors for a model with the standalone interpreter? **1 / 1 poin**

- (●) Call get_input_details() after calling allocate_tensors() on the interpreter
- Call get_input_details() after initializing the interpreter
- Call get_input_tensors() after initializing the interpreter
- Call get_input_tensors() after calling allocate_tensors() on the interpreter

Benar

7. How do you perform inference using the interpreter? **1 / 1 poin**

- Call invoke(), and pass it both the input and output tensors
- Call invoke(), and pass it the input tensor
- Just call invoke(), TensorFlow can do the rest
- (●) Set the Input tensor with the set_tensor command and then call invoke()

Benar

8. How do you read the results of inference using the interpreter? **1 / 1 poin**

- Call invoke(), pass it the input and output tensors, and then read the output tensor
- Call invoke(), and the the output will be rendered automatically
- (●) Call invoke(), and then call get_tensor() on the interpreter to read the output
- Call invoke(), pass it the input tensor, read the results

Benar