TensorFlow Lite

The professional course

What is our goal?

To enable our students to **build mobile products**

based on **machine learning at the edge**.

Why?

- High demand for mobile applications.
- Hardware constraints.
- Privacy concernings.
- Highly profitable opportunities.

For who is this course?

- Machine learning engineers.
- Mobile software engineers.
- Enthusiasts of **AI** and **mobile applications** in general.

Who are your instructors?

Michel Meneses
Software Engineer, Machine Learning
M.S. Computer Science, B.S. Computer Engineering (Federal University of Sergipe/Brazil)

Luiz Vitor Reis
Software Engineer, Embedded Systems
B.S. Mechatronics Engineering (University of Brasília/Brazil)

What will you learn?

- How to embed machine learning models on mobile devices (FREE).
- How to evaluate the performance of machine learning models on mobile apps.
- How to optimize machine learning models for mobile devices.
- How to develop an optimized machine learning-based mobile app from scratch.

TensorFlow Lite

Week 1

Agenda

1. Machine learning at the edge
2. Introduction to TF Lite
    a. Goal
    b. Advantages
    c. Architecture
3. Hands-on project
4. Wrap-up

Machine Learning at the Edge

Machine learning at the edge

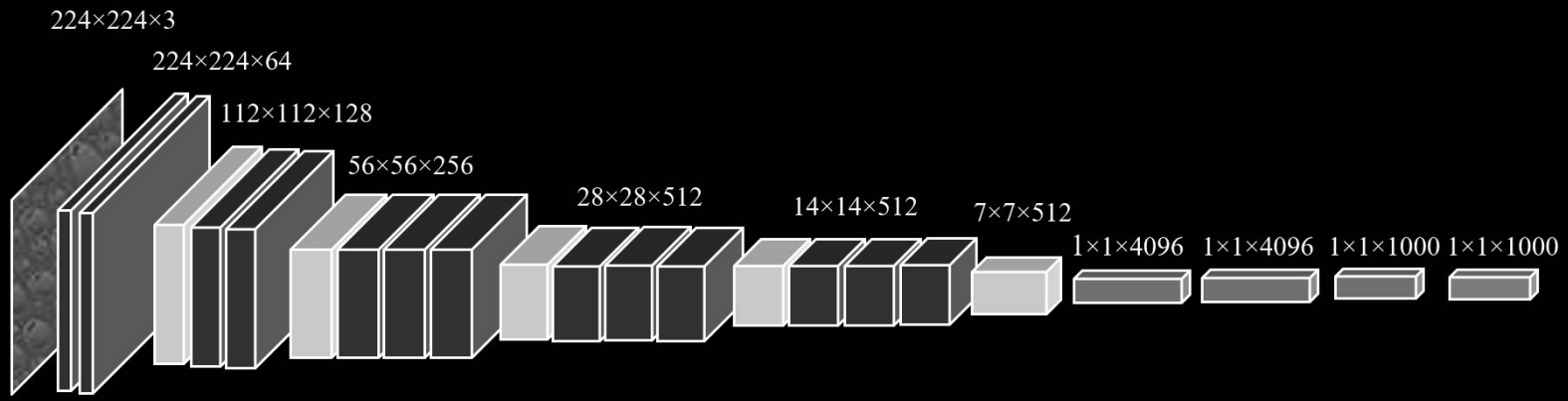
Initially, only simple models were practical (*e.g.*, Viola-Jones face detector).

Machine learning at the edge

Since 2012, deep learning has become the state-of-the-art for many ML problems.

Machine learning at the edge

However, deep learning models are *deep* (*i.e.* large and computationally expensive).

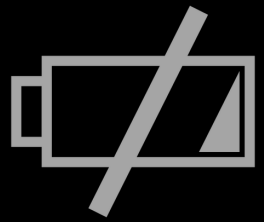


Machine learning at the edge

Designing deep learning mobile applications based on the cloud.

Machine learning at the edge

Issues of cloud-based architectures: internet need, battery draining and time delay.



Machine learning at the edge

Solution: frameworks optimized for **running** deep learning models at the edge.

Introduction to TF Lite

Introduction to TF Lite

TensorFlow Lite was released in 2019 by Google Brain.



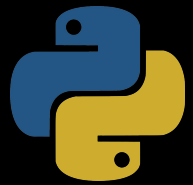

Introduction to TF Lite

Advantages:

- **Latency**: there is no round-trip to a server.
- **Privacy**: no data needs to leave the device.
- **Connectivity**: an Internet connection **is not** required.
- **Power consumption**: network connections are power-hungry.

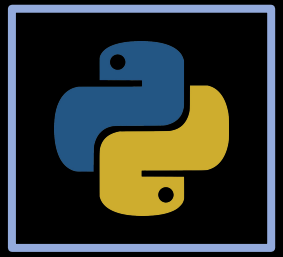
Introduction to TF Lite

TensorFlow Lite is cross-platform.



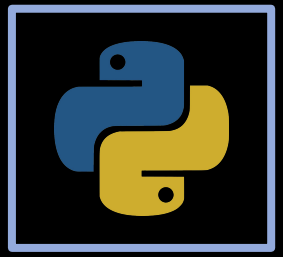
Introduction to TF Lite

TensorFlow Lite is cross-platform.



Introduction to TF Lite

TensorFlow Lite is cross-platform.


Introduction to TF Lite

TF Lite Architecture:

Pick a model

Pick a new model or retrain an existing one.

Convert

Convert a TensorFlow model into a compressed flat buffer with the TensorFlow Lite Converter.

Deploy

Take the compressed .tflite file and load it into a mobile or embedded device.

Optimize

Quantize by converting 32-bit floats to more efficient 8-bit integers or run on GPU.

Introduction to TF Lite

TF Lite Converter (Python):

```
import tensorflow as tf
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
```

Introduction to TF Lite

TF Lite Interpreter (Android/Java):

```
MappedByteBuffer tfLiteModel = FileUtil.loadMappedFile( context: this, MODEL_FILENAME);
this.interpreter = new Interpreter(tfLiteModel, new Interpreter.Options());
interpreter.run(inputImage.getBuffer(), output.getBuffer());
```

Hands-on Project

Hands-on Project

Classifying images of dogs (dataset “Stanford Dogs”)

Hands-on Project

Steps:

1. Build and train a model using TensorFlow
2. Convert it to TF Lite
3. Embed it in Android
4. Run the application

Wrap-Up

Wrap-Up

During this week we have learned:

1. The importance of machine learning at the edge frameworks
2. TensorFlow Lite’s architecture
3. How to embed a deep learning model in a mobile application using TF Lite

Wrap-Up

However, there are several open questions:

1. How can we assess the performance of our embedded model?
2. How can we optimize our embedded model?
3. How can we build a commercial mobile product using TensorFlow Lite?

Wrap-Up

Stay tuned for **Week 2** of *TensorFlow Lite - The professional course*!