

Digital Talent Scholarship 2022

Device Based model in TF Lite

Lead a sprint through the Machine Learning Track

Agenda

- Training own model
- Running TF model in Android App

Objektif Pembelajaran

- Mengerti cara kerja TensorFlow Lite
- Memahami apa itu model quantization
- Dapat menjelaskan cara kerja GPU delegates
- Mengenal Optimization techniques bagi TensorFlow Lite

Are your students ML-ready?

Recap

PART 1

Machine Learning Models in Mobile and Embedded Systems

Features

- Lightweight
- Low-latency
- Privacy
- Improved Power consumption
- Efficient model format
- Pre-trained models

Components in TensorFlow Lite

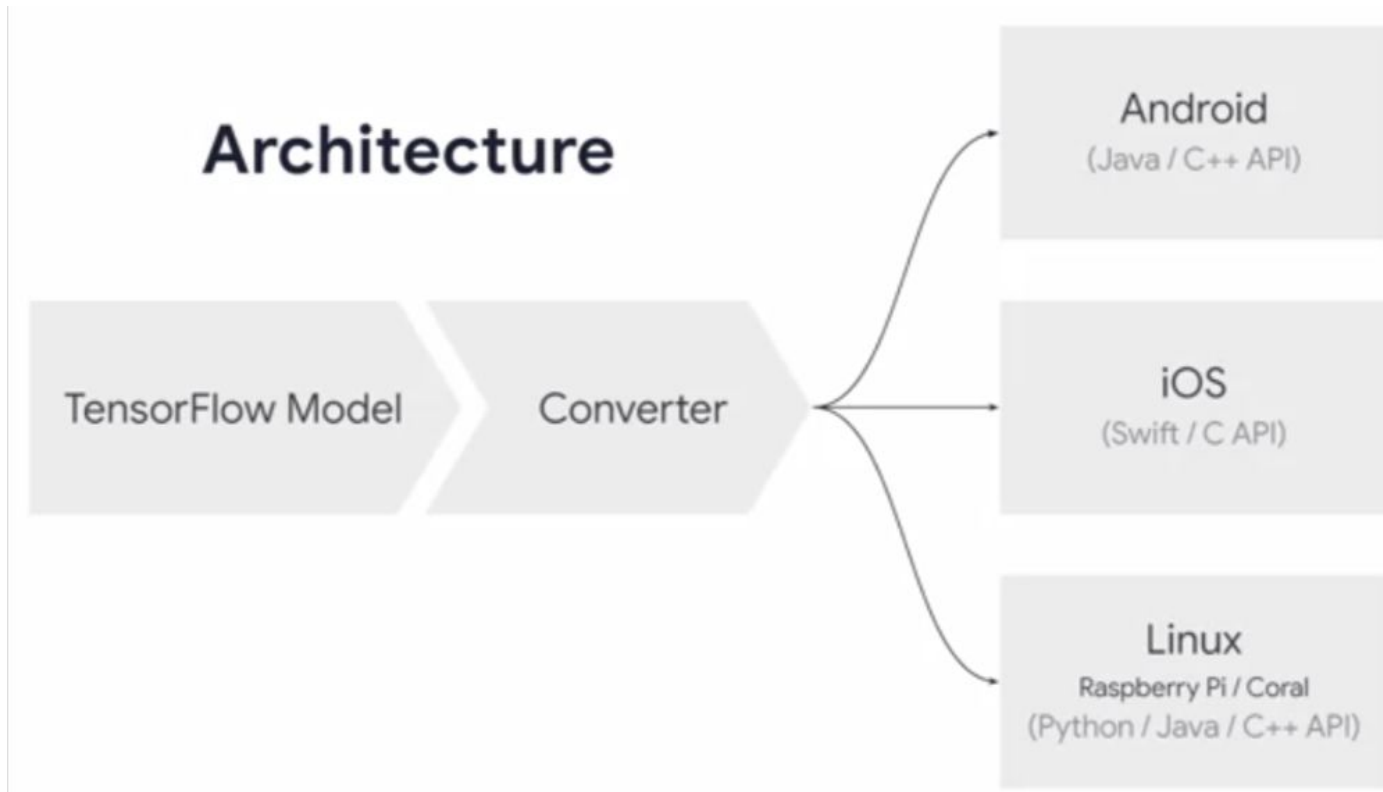
Converter (to TensorFlow Lite format)

- Mengubah Model Tensorflow menjadi bentuk yang lebih efisien dibaca oleh interpreter
- Memperkenalkan optimization untuk meningkatkan kinerja model dan/atau untuk mengurangi size model

Interpreter (Core)

- Support multiple platforms (Android, iOS, embedded Linux, microcontrollers)
- Platform APIs for accelerated inference

Architecture



Performance

Performance

Acceleration	Available
Software	NN API (also a delegate)
	Edge TPU
	GPU
	CPU Optimizations (ARM and x86)

NN API

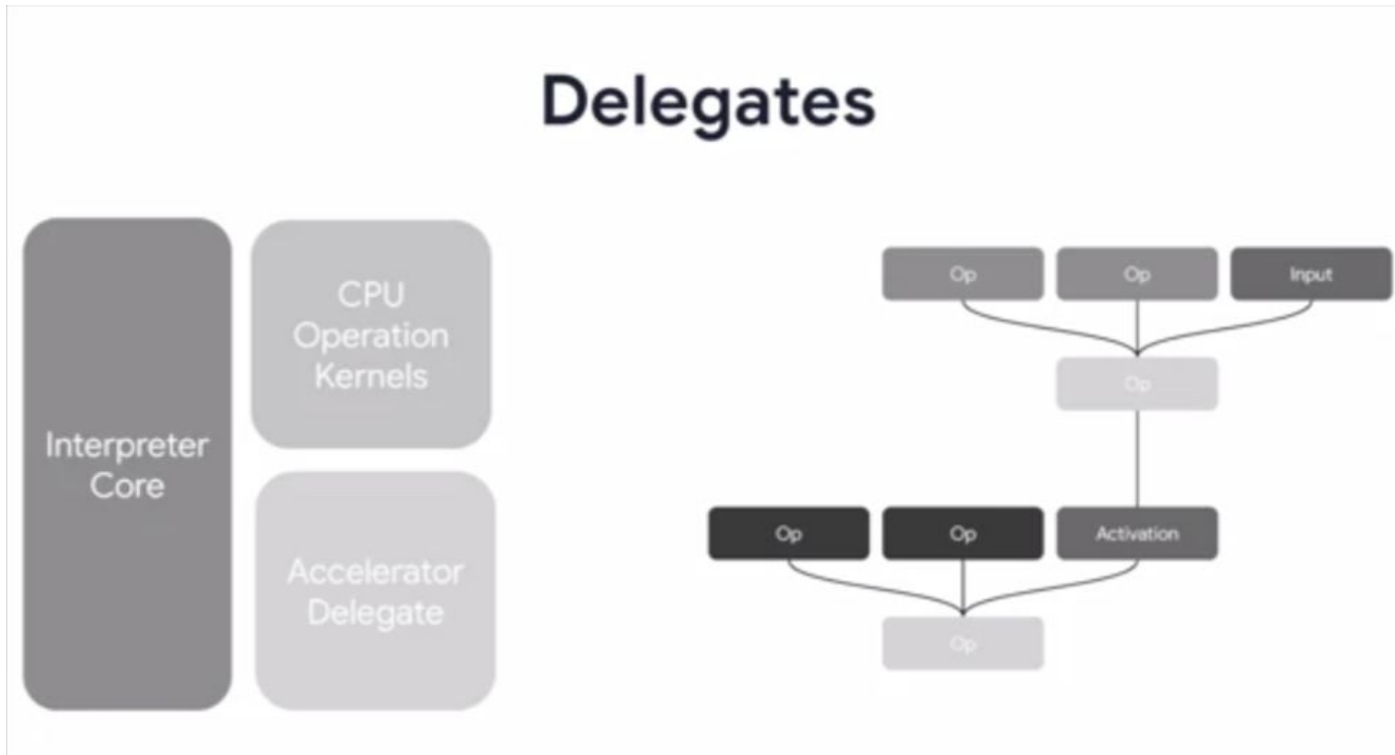
Android Neural Networks API (NNAPI) adalah Android C API yang dirancang untuk menjalankan operasi komputasi intensif bagi machine learning di perangkat Android. NNAPI dirancang untuk menyediakan lapisan dasar fungsionalitas bagi framework machine learning dengan level lebih tinggi, seperti TensorFlow Lite dan Caffe2, yang membuat dan melatih jaringan neural. API ini tersedia di semua perangkat yang menjalankan Android 8.1 (API level 27) atau yang lebih baru.

[TensorFlow Lite, Experimental GPU Delegate - YouTube](#)

Delegasi

Delegasi memungkinkan akselerasi perangkat keras model TensorFlow Lite dengan memanfaatkan akselerator di perangkat seperti GPU dan Digital Signal Processor (DSP).

Delegates



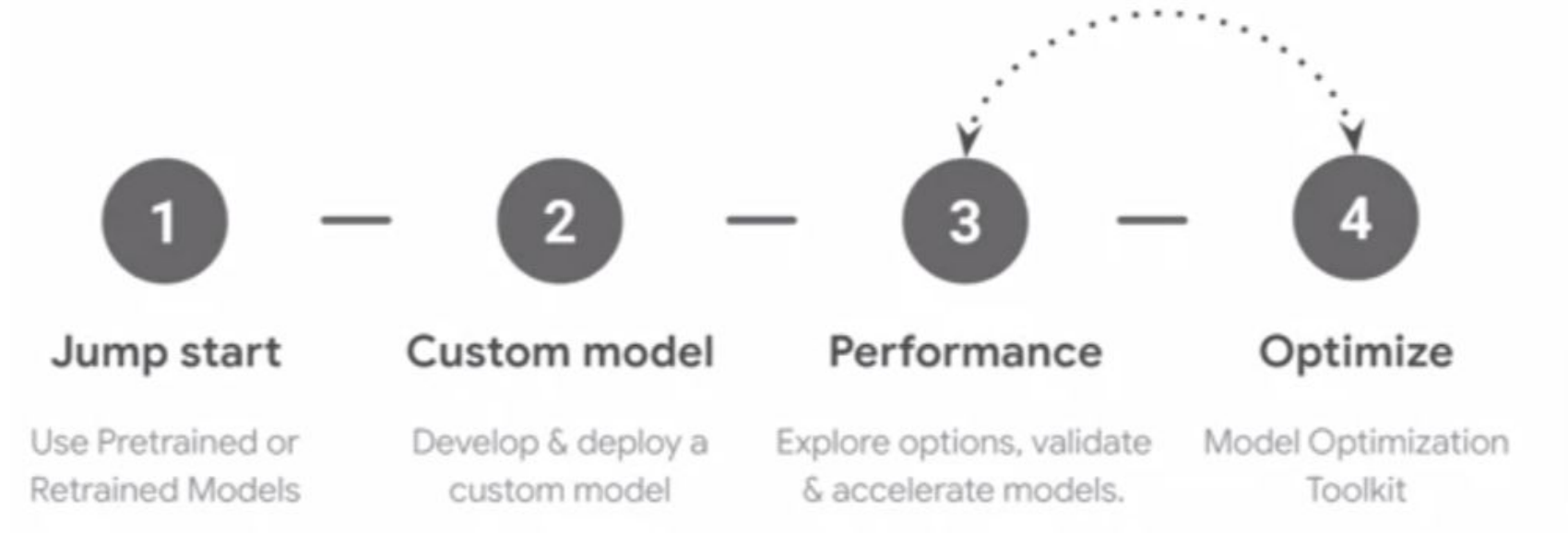
Techniques

- Quantization, Mengurangi presisi weight dan bias
- Weight pruning, Mengurangi jumlah parameter
- Model topology transform, Mengubah bentuk model
 - Tensor Decomposition
 - Distillation

Why Quantize?

- Semua jenis CPU platform disupport
- Mengurangi latency dan inference cost
- Low Memory Footprint
- Allow execution on hardware restricted-to or optimized-for fixed-point operations
- Optimized models for special purpose HW accelerators (TPUs)

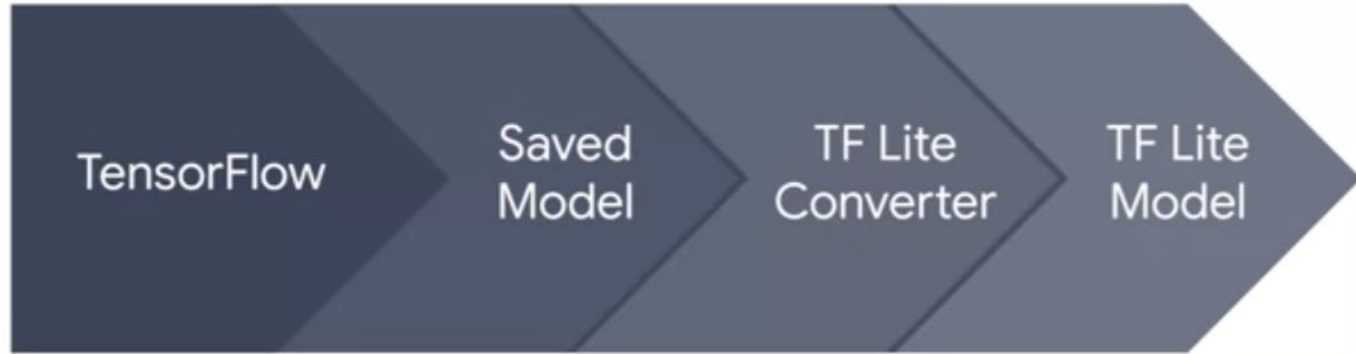
Steps dalam optimizing



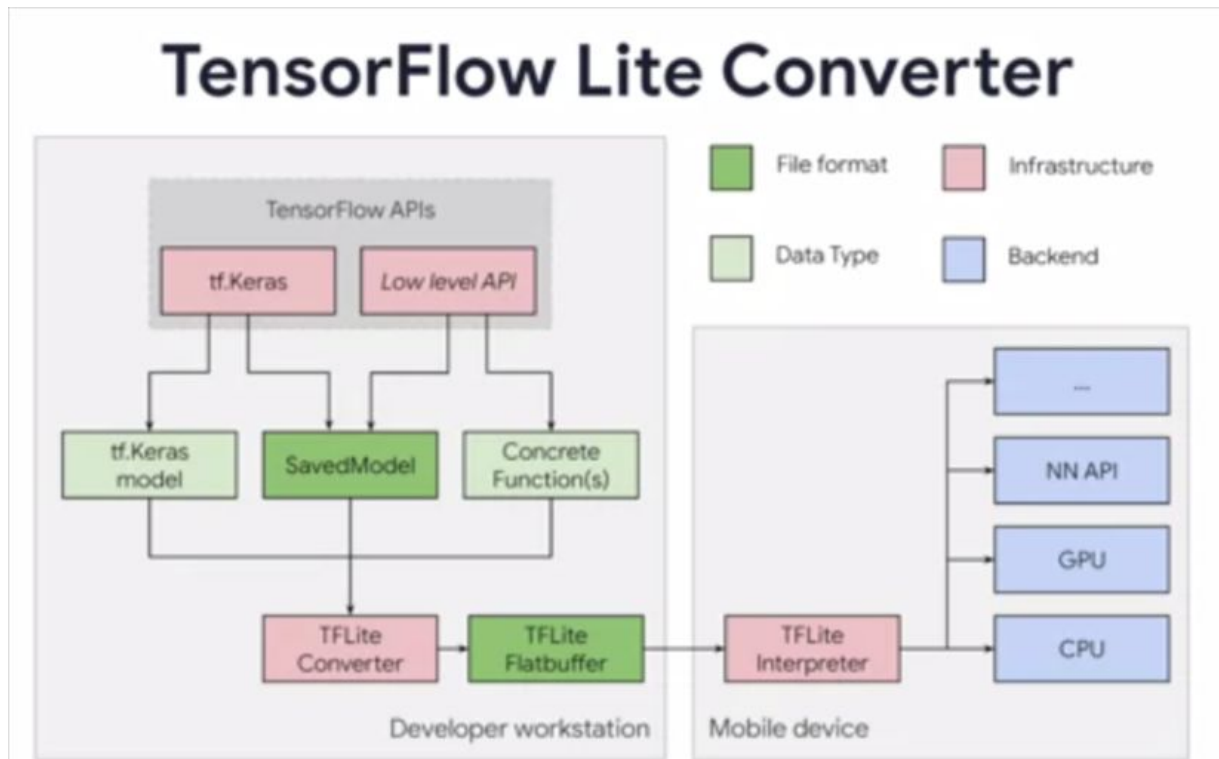
PART 2

Taking a look at the saved model format

Saving, converting, and optimizing a model



Saving, converting, and optimizing a model



Parameters for conversion



SavedModel

- Standar untuk serialize sebuah TensorFlow model
- Akan ada MetaGraph, sebuah data flow graph, untuk menyimpan metadata yang membantu prediksi model.
- Ada snapshot dari trained model (dengan weights dan computation model)
- Tidak memerlukan building-code model required
- Supports model versioning

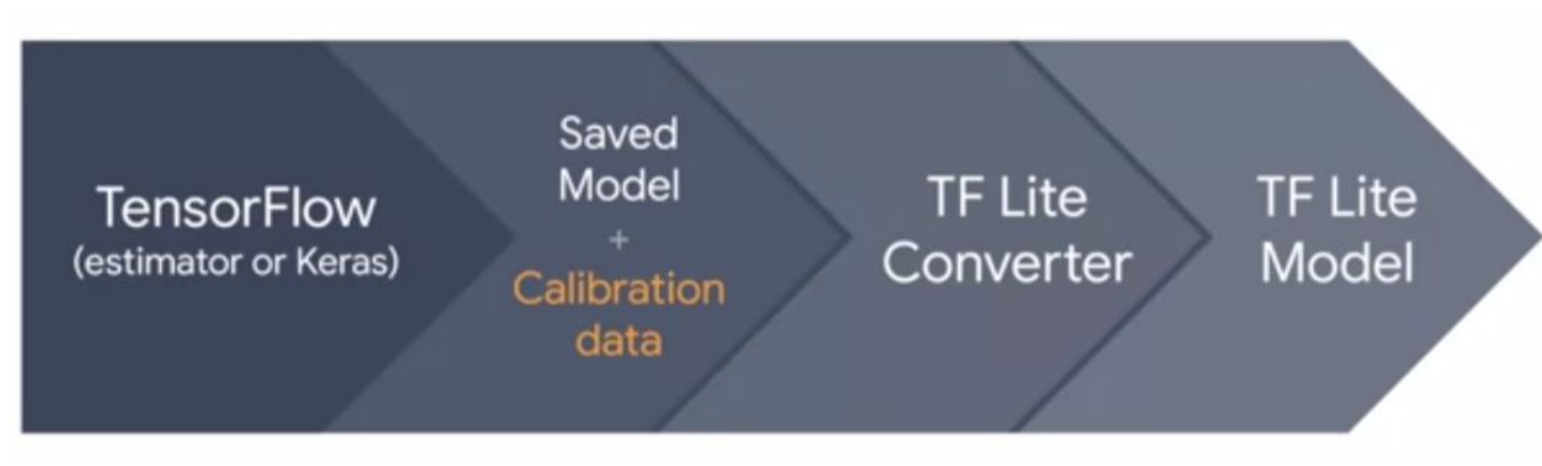
DEMO Example 1

DEMO Example 2

Post-training quantization

- Mengurangi precision dengan 3 kali lipat lower latency
- Sedikit degradation pada akurasi model
- Optimization modes
 - Default (Both size dan Latency)
 - Size
 - Latency
- Efficiently represents an arbitrary magnitude of ranges
- Quantization target specification

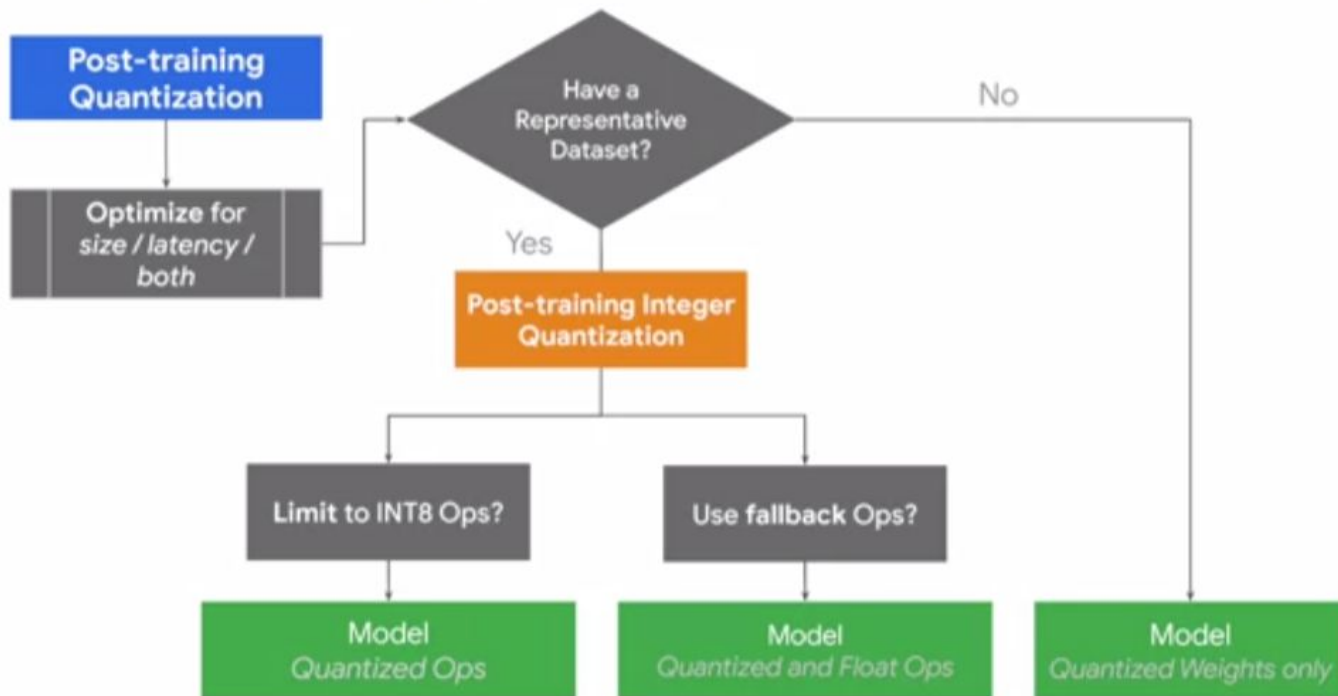
Post-training integer quantization



DEMO Post-training quantization

Paths in Optimization

Optimizing your models in a nutshell



Quantization in deep learning | Deep Learning Tutorial 49 (Tensorflow, Keras & Python) - YouTube

Inside TensorFlow: TF Model Optimization Toolkit (Quantization and Pruning) - YouTube

DEMO TFLite Interpreter in Python

PART 3

First primer on running models on mobile devices

Running Models

Pretrained models

- Image classification
- Object detection
- Smart reply
- Pose estimation
- Segmentation

TensorFlow Hub

- Classification modules
- Feature vector modules
- Embedding modules

Not what you're looking for?

Build a custom model!

Getting a basic model running

Get started

1

Build a model

Create a simple model
for $(y = 2x - 1)$
from simulated data
and train it

2

Export & Convert

Generate the
SavedModel and
convert it to TFLite

3

Verify

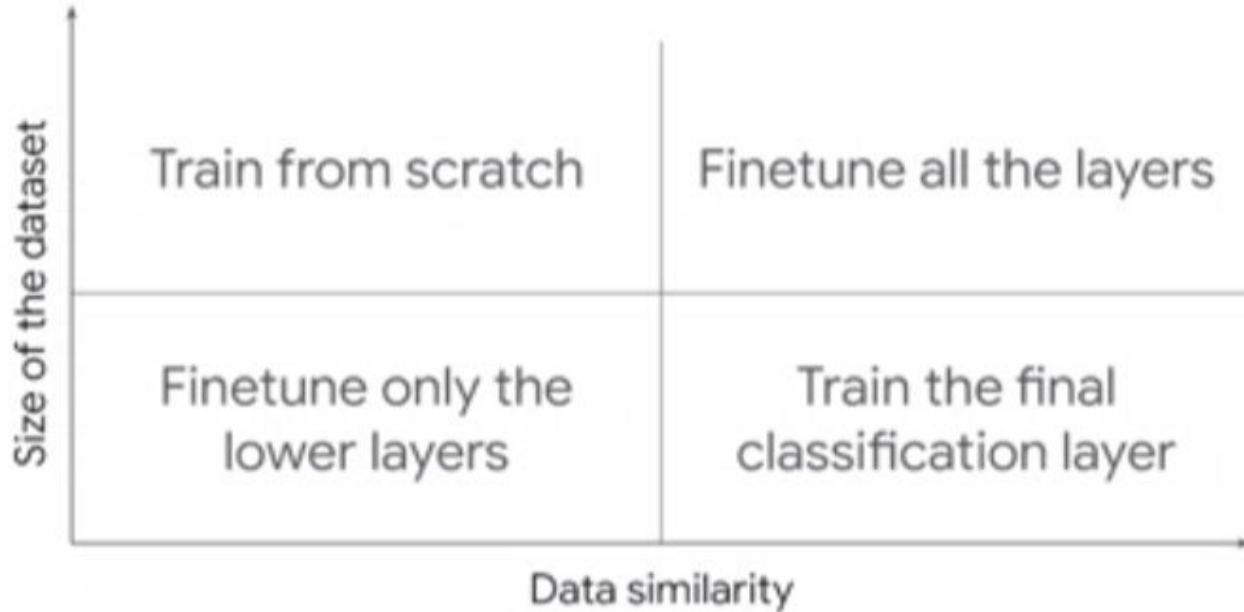
Perform evaluation on
random data to verify the
results

4

Deploy

Deploy the converted
model on a mobile
device (Android/iOS)

Transfer learning



Transfer learning



DEMO Converting a model to TFLite

DEMO Transfer Learning with TFLite



Q & A

Thank You