Android Developers Guide to Machine Learning 🧠

With MLKit, Tensorflow & Firebase 🔥

Rebecca Franks

@riggaroo

Google Developer Expert

Android @ Over

Pluralsight Author

GDG Johannesburg Organiser

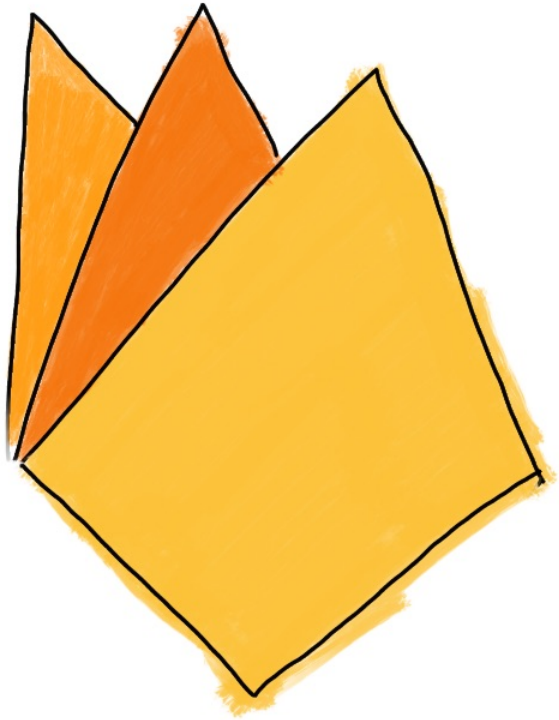
Over

Who considers themselves an expert in Machine Learning?

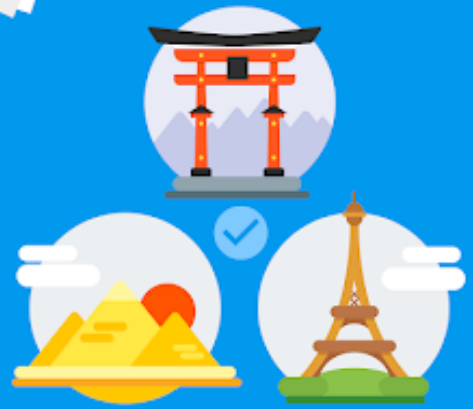
What is Machine Learning?

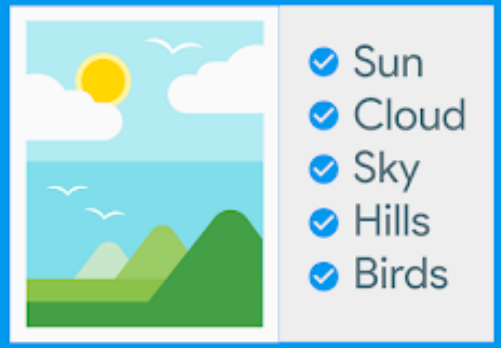
Machine learning is an application of Artificial Intelligence in which we input a lot of data and let the machines **learn** “by themselves”


ML Kit




Main St.
Menu




Sun
Cloud
Sky
Hills
Birds

Google Play
Services

com.google.android.gms.vision.face

Classes

Face	A human face detected in an image or video.
FaceDetector	Detector for finding **Faces** in a supplied image.
FaceDetector.Builder	Builder for creating face detector instances.
Landmark	A point on a detected face, such as an eye, nose, or mouth.
LargestFaceFocusingProcessor	Face processor that focuses on tracking a single "prominent face", in conjunction with the associated **FaceDetector**.
LargestFaceFocusingProcessor.Builder	Builder for creating a LargestFaceFocusingProcessor.

Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 3.0 License, and code samples are licensed under the Apache 2.0 License. For details, see our Site Policies. Java is a registered trademark of Oracle and/or its affiliates.

Last updated August 14, 2017.

com.google.android.gms.vision.face

Classes

Face	A human face det… an image or video.
FaceDetector	Detector f… ing **Faces** in a supplied image.
FaceDetector.Builder	…r creating face detector instances.
Landmark	A point on …ted face, such as an eye, nose, or mouth.
LargestFaceFocusin… …or	Face processor that … on tracking a single "prominent face", in conjunction with the associated **FaceDetect…**
Large… …ocusingProcessor.Builder	Builder for creating a LargestFaceF… …Processor.

Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 3.0 L… …d code samples are licensed under the Apache 2.0 License. For details, see our Site Policies. Java is a registered trademark of Oracle and/or its …

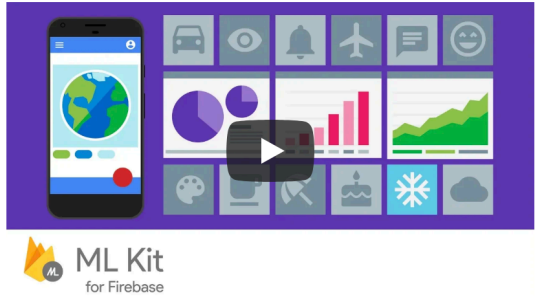
Last updated August 14, 2017.

ML Kit for Firebase

iOS

Use machine learning in your apps to solve real-world problems.

ML Kit is a mobile SDK that brings Google's machine learning expertise to Android and iOS apps in a powerful yet easy-to-use package. Whether you're new or experienced in machine learning, you can implement the functionality you need in just a few lines of code. There's no need to have deep knowledge of neural networks or model optimization to get started. On the other hand, if you are an experienced ML developer, ML Kit provides convenient APIs that help you use your custom TensorFlow Lite models in your mobile apps.

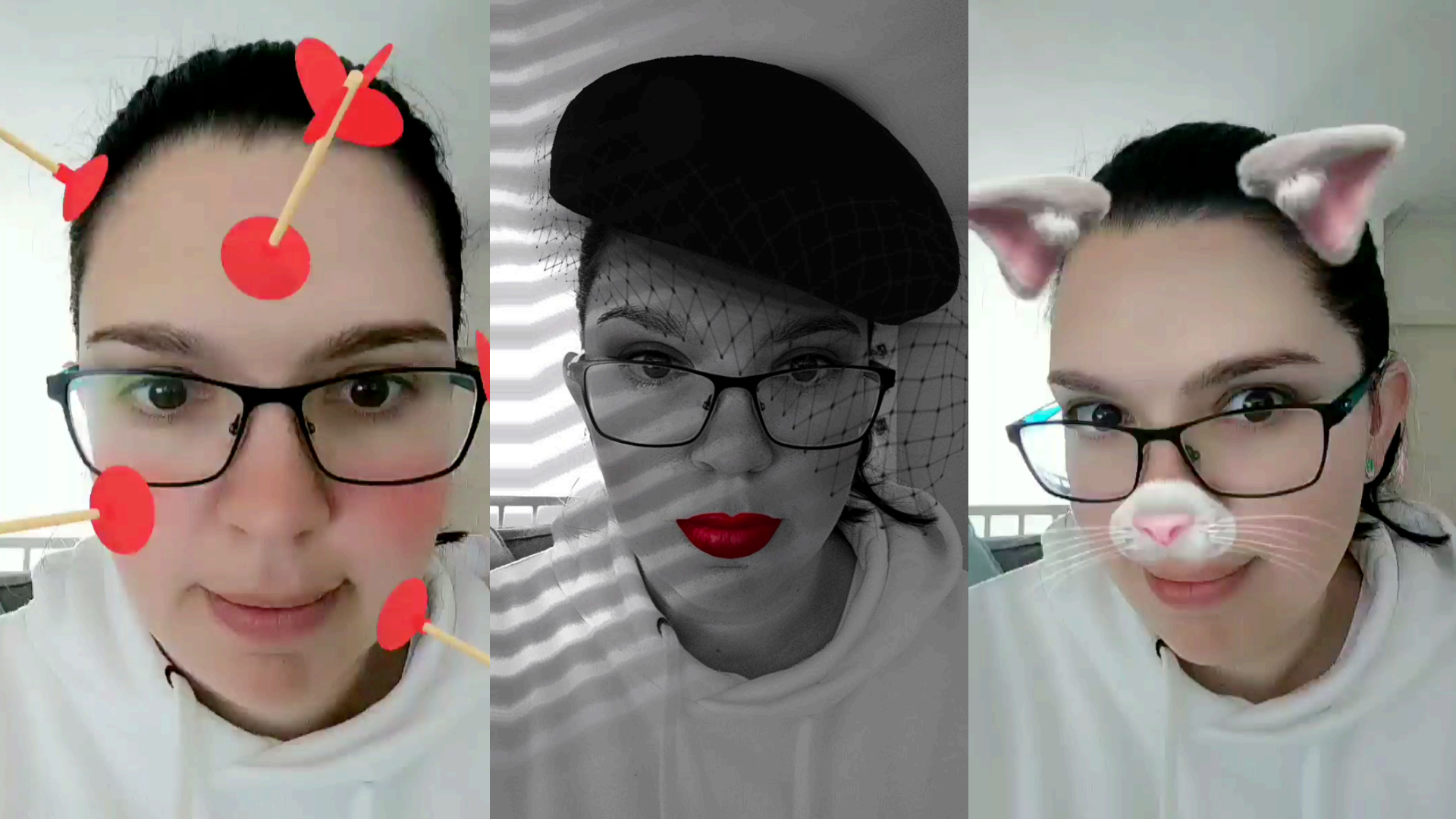


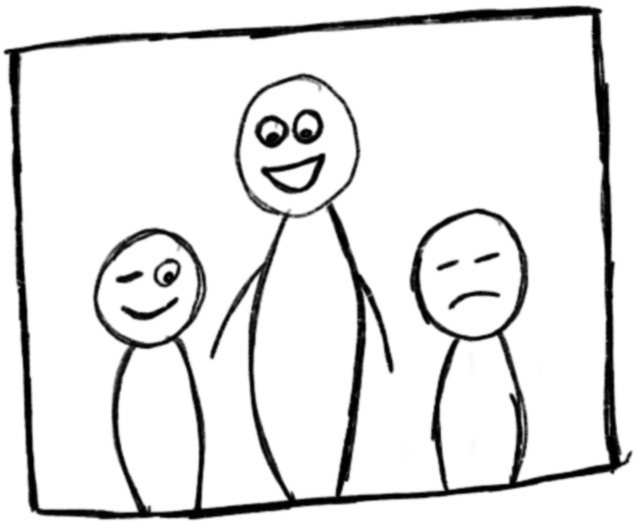
This is a beta release of ML Kit for Firebase. This API might be changed in backward-incompatible ways and is not subject to any SLA or deprecation policy.

Key capabilities

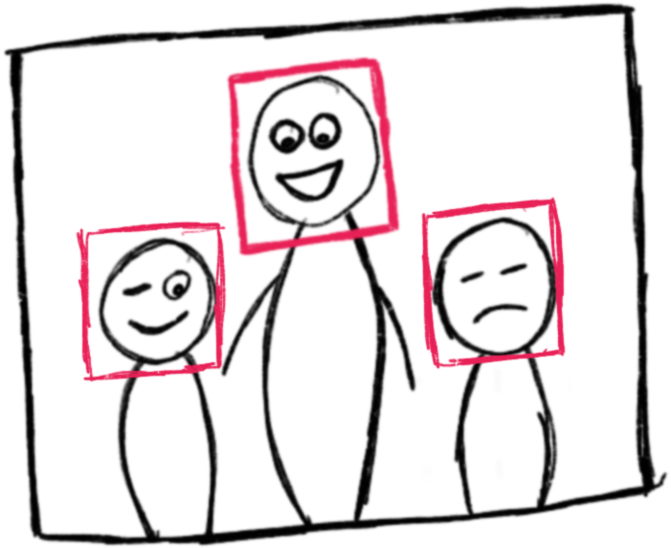
Works offline

Face detection 

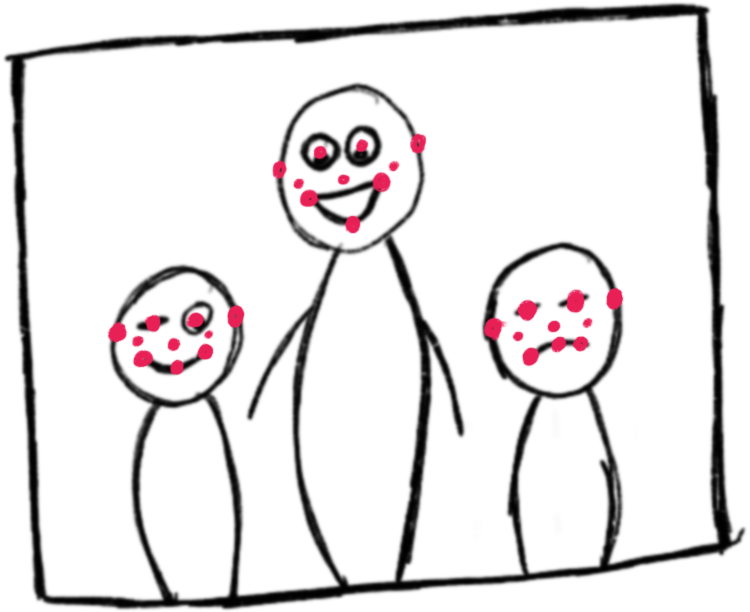




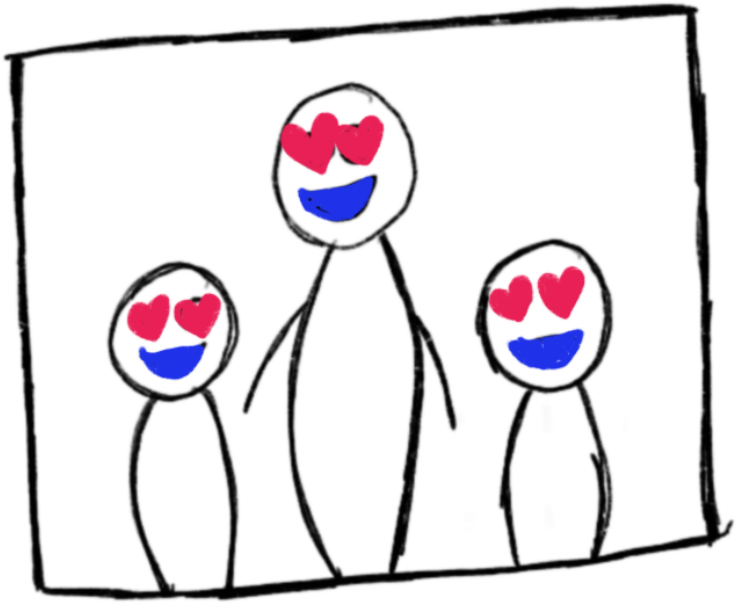
Bounding Boxes



LANDMARKS



PROFIT



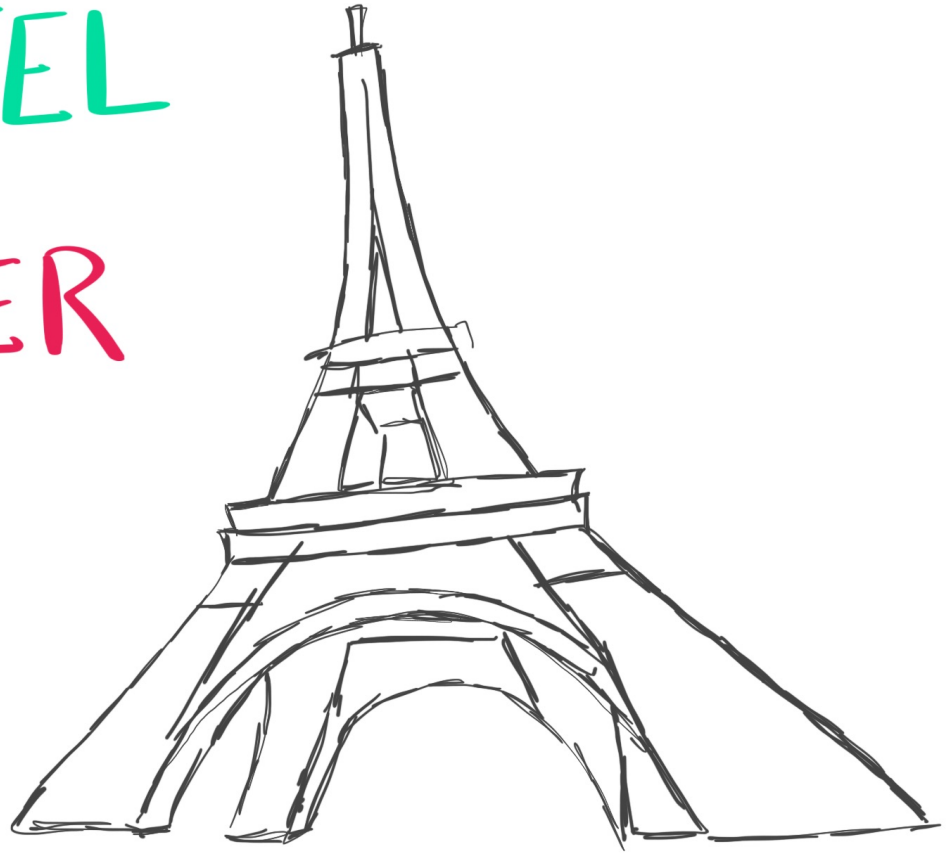
For each face detected:

Face 1 of 3		
Bounding polygon	(884.880004882812, 149.546676635742), (1030.77197265625, 149.546676635742), (1030.77197265625, 329.660278320312), (884.880004882812, 329.660278320312)	
Angles of rotation	Y: -14.054030418395996, Z: -55.007488250732422	
Tracking ID	2	
Facial landmarks	**Left eye**	(945.869323730469, 211.867126464844)
	Right eye	(971.579467773438, 247.257247924805)
	Bottom of mouth	(907.756591796875, 259.714477539062)
	... etc.	
Feature probabilities	**Smiling**	0.88979166746139526
	Left eye open	0.98635888937860727
	Right eye open	0.99258323386311531

Demo

Landmark detection

EIFFEL TOWER

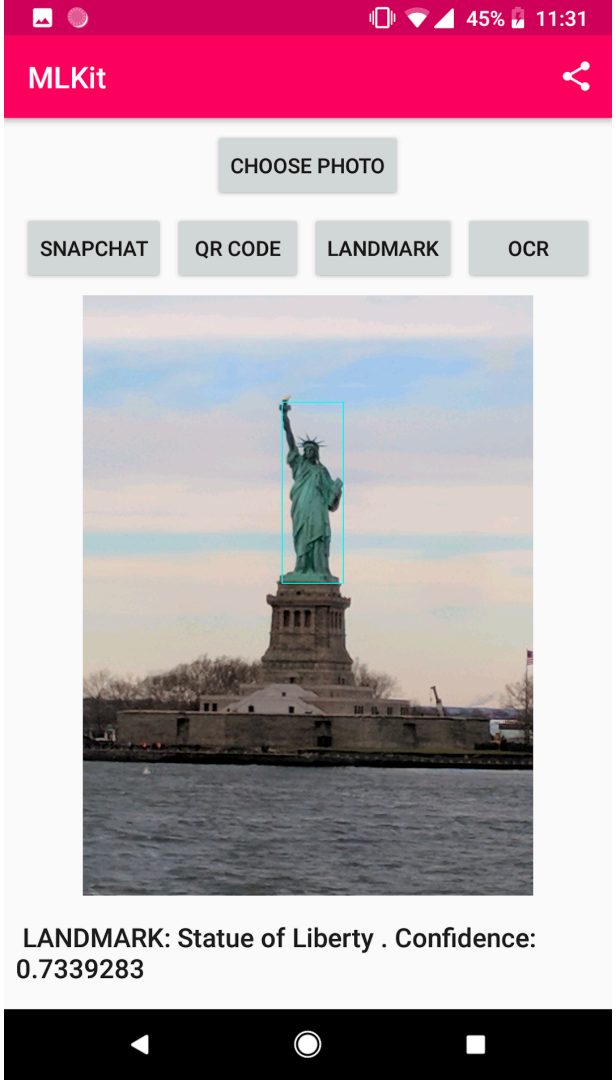


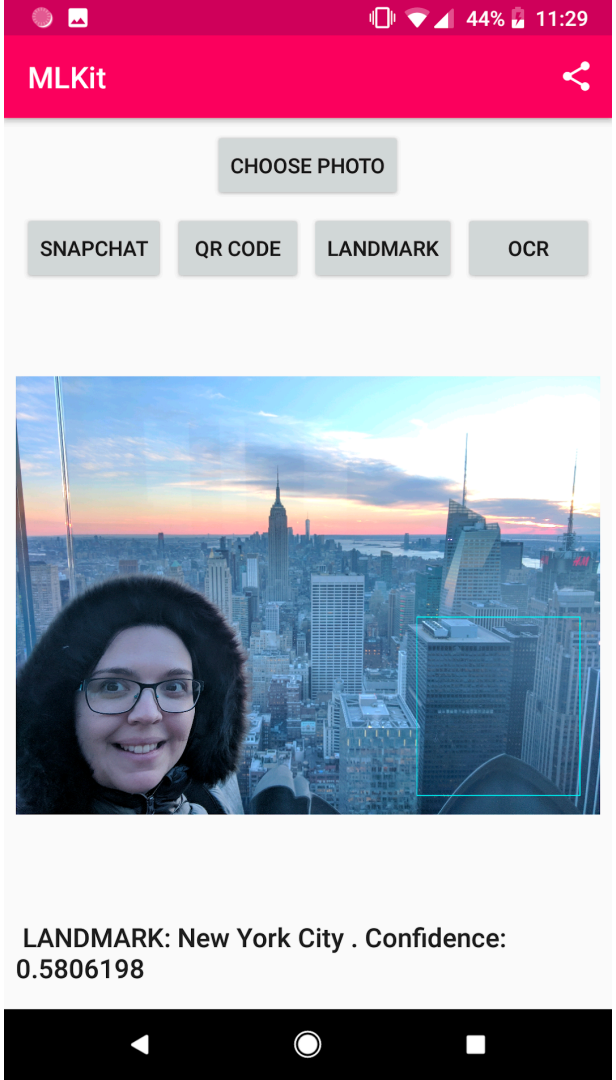
Example Results

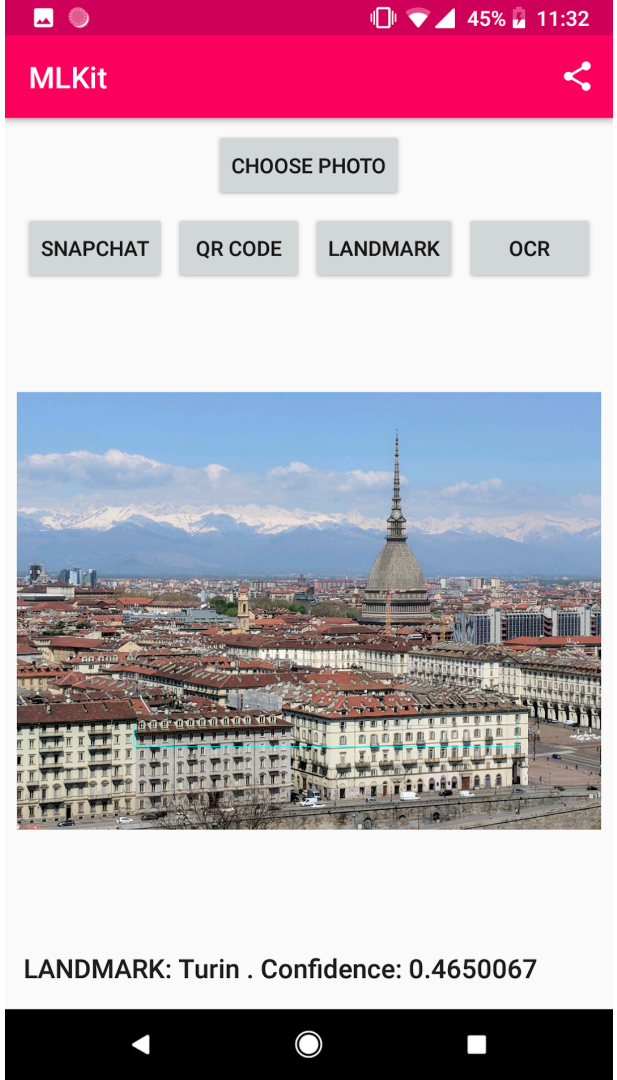
Photo: Arcalino / Wikimedia Commons / CC BY-SA 3.0

Result	
Description	Brugge
Geographic Coordinates	51.207367, 3.226933
Knowledge Graph entity ID	/m/0drjd2
Bounding Polygon	(20, 342), (651, 342), (651, 798), (20, 798)
Confidence Score	0.77150935

Demo


45% 11:31
MLKit
CHOOSE PHOTO
SNAPCHAT
QR CODE
LANDMARK
OCR
LANDMARK: Statue of Liberty . Confidence: 0.7339283


44% 11:29
MLKit
CHOOSE PHOTO
SNAPCHAT
QR CODE
LANDMARK
OCR
LANDMARK: New York City . Confidence: 0.5806198


45% 11:32
MLKit
CHOOSE PHOTO
SNAPCHAT
QR CODE
LANDMARK
OCR
LANDMARK: Turin . Confidence: 0.4650067

Works offline

Image Labelling 

Example results

Photo: Clément Bucco-Lechat / Wikimedia Commons / CC BY-SA 3.0

On-device		Cloud	
Description	Stadium	**Description**	sport venue
Knowledge Graph entity ID	/m/019cfy	**Knowledge Graph entity ID**	/m/0bmgjqz
Confidence	0.9205354	**Confidence**	0.9860726

Works offline

Barcode scanning 📸

Example results

Result	
Corners	(49,125), (172,125), (172,160), (49,160)
Raw value	2404105001722

Result		
Corners	(87,87) (612,87) (612,612) (87,612)	
Raw value	`WIFI:S:SB1Guest;P:12345;T:WEP;;`	
WiFi information	SSID	SB1Guest
	Password	12345
	Type	WEP

```
val options = FirebaseVisionBarcodeDetectorOptions.Builder()
    .setBarcodeFormats(FirebaseVisionBarcode.FORMAT_QR_CODE)
    .build()

val image = FirebaseVisionImage.fromBitmap(bitmap)

val detector = FirebaseVision.getInstance()
                .getVisionBarcodeDetector(options)

detector.detectInImage(image)
    .addOnSuccessListener {
        processedBitmap.postValue(barcodeProcessor.drawBoxes(bitmap, it))
```

```
val options = FirebaseVisionBarcodeDetectorOptions.Builder()
    .setBarcodeFormats(FirebaseVisionBarcode.FORMAT_QR_CODE)
    .build()

val image = FirebaseVisionImage.fromBitmap(bitmap)

val detector = FirebaseVision.getInstance()
                .getVisionBarcodeDetector(options)

detector.detectInImage(image)
    .addOnSuccessListener {
        processedBitmap.postValue(barcodeProcessor.drawBoxes(bitmap, it))
        var result = String()
        it.forEach {
            result += "VALUE TYPE: ${it.valueType} Raw Value: ${it.rawValue}"
            textResult.postValue(result)
```

```
val image = FirebaseVisionImage.fromBitmap(bitmap)

val detector = FirebaseVision.getInstance()
                .getVisionBarcodeDetector(options)

detector.detectInImage(image)
    .addOnSuccessListener {
        processedBitmap.postValue(barcodeProcessor.drawBoxes(bitmap, it))
        var result = String()
        it.forEach {
            result += "VALUE TYPE: ${it.valueType} Raw Value: ${it.rawValue}"
            textResult.postValue(result)
        }
    }.addOnFailureListener{
        textResult.postValue(it.message)
    }
```

82% 16:51
MLKit
CHOOSE PHOTO
SNAPCHAT
QR CODE
LANDMARK
OCR
VALUE TYPE: 8 FORMAT: 256 Raw Value:
https://internationalbarcodes.com/

84% 16:54
MLKit
CHOOSE PHOTO
SNAPCHAT
QR CODE
LANDMARK
OCR
VALUE TYPE: 7 FORMAT: 256 Raw Value: bit.ly/
mlkit-riggaroo

83% 16:52
MLKit
CHOOSE PHOTO
SNAPCHAT
QR CODE
LANDMARK
OCR
www.shutterstock.com · 256149148
VALUE TYPE: 7 FORMAT: 256 Raw Value:
4545454324545864513215643jhhffsdrdyttlopicd
eimmeriououtiuoirmirt,ipep,iep6346hugffy56546
VALUE TYPE: 7 FORMAT: 256 Raw Value:
4545454324545864513215643 6346hugffy56546

Works offline

OCR 🖍️ 🔤


GARY st
Menu

On Device vs Cloud

	On-device	Cloud
Pricing	Free	Free for first 1000 uses of this feature per month: see Pricing
Ideal use cases	Real-time processing—ideal for a camera or video feed	High-accuracy text recognition Document scanning See the Cloud Vision API demo.
Language support	Recognizes Latin characters	Recognizes and identifies a broad range of languages and special characters

```
private fun doOcrDetection(bitmap: Bitmap){
    val detector = FirebaseVision.getInstance()
            .visionTextDetector

    val firebaseImage = FirebaseVisionImage.fromBitmap(bitmap)

    detector.detectInImage(firebaseImage)
        .addOnSuccessListener {
            processedBitmap.postValue(ocrProcessor.drawBoxes(bitmap, it))
            var result = String()
            it.blocks.forEach {
                result += " " + it.text
                textResult.postValue(result)
```

```
private fun doOcrDetection(bitmap: Bitmap){
    val detector = FirebaseVision.getInstance()
            .visionTextDetector

    val firebaseImage = FirebaseVisionImage.fromBitmap(bitmap)

    detector.detectInImage(firebaseImage)
        .addOnSuccessListener {
            processedBitmap.postValue(ocrProcessor.drawBoxes(bitmap, it))
            var result = String()
            it.blocks.forEach {
                result += " " + it.text
                textResult.postValue(result)
           }
        }
        .addOnFailureListener{
            Toast.makeText(/../“Error detecting Text $it”/../)
        }
```

```
private fun doOcrDetection(bitmap: Bitmap){
    val detector = FirebaseVision.getInstance()
            .visionTextDetector

    val firebaseImage = FirebaseVisionImage.fromBitmap(bitmap)

    detector.detectInImage(firebaseImage)
        .addOnSuccessListener {
            processedBitmap.postValue(ocrProcessor.drawBoxes(bitmap, it))
            var result = String()
            it.blocks.forEach {
                result += " " + it.text
                textResult.postValue(result)
           }
        }
        .addOnFailureListener{
            Toast.makeText(/../“Error detecting Text $it”/../)
        }
```

The Sunday Times No. l Bestselling Author The queen of witty, heart-warming.
feel-good love stories Red this could change everyth

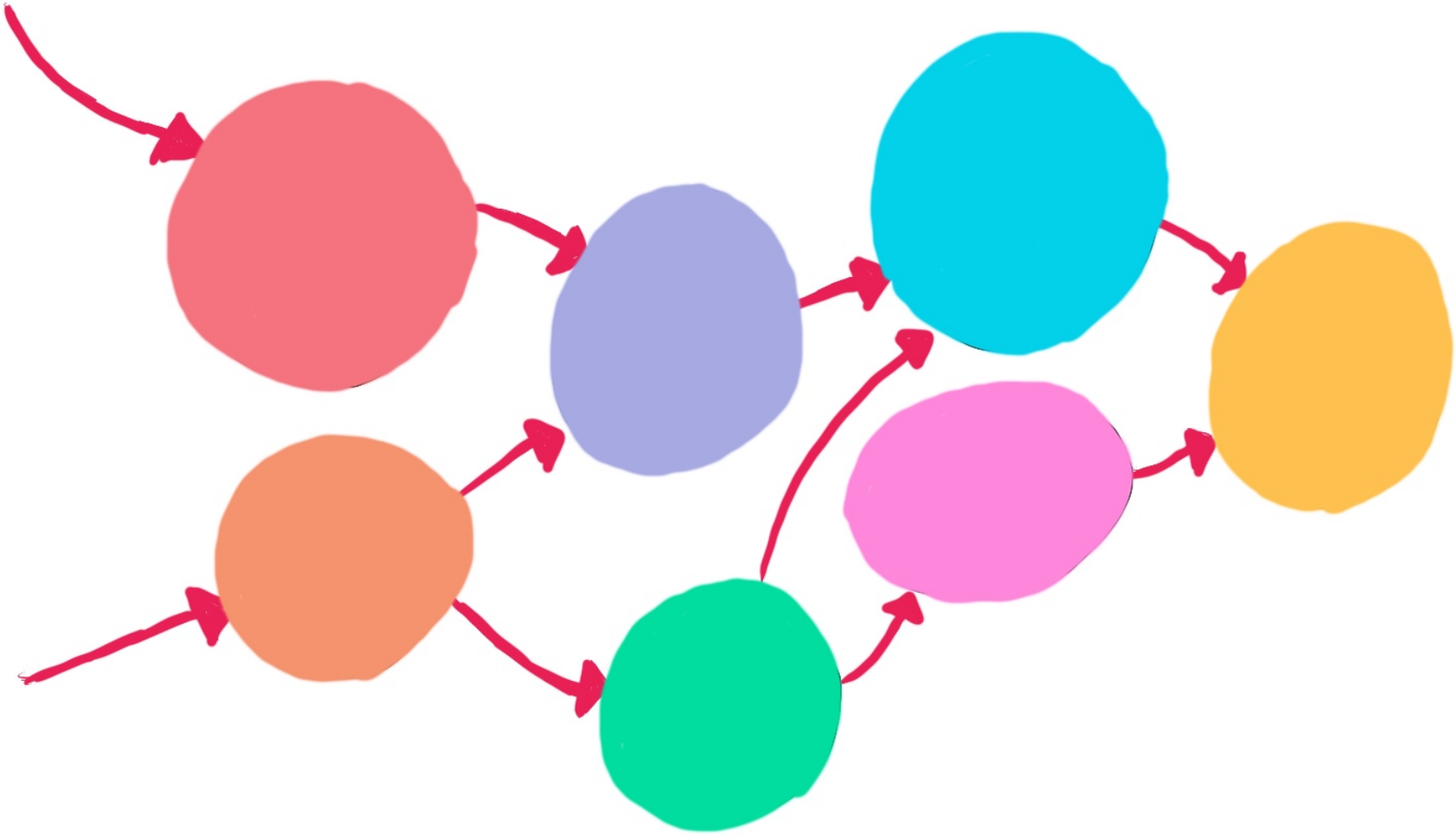
hangry /hapgri/ adj.
bad-tempered or irritable
as a result of hunger.

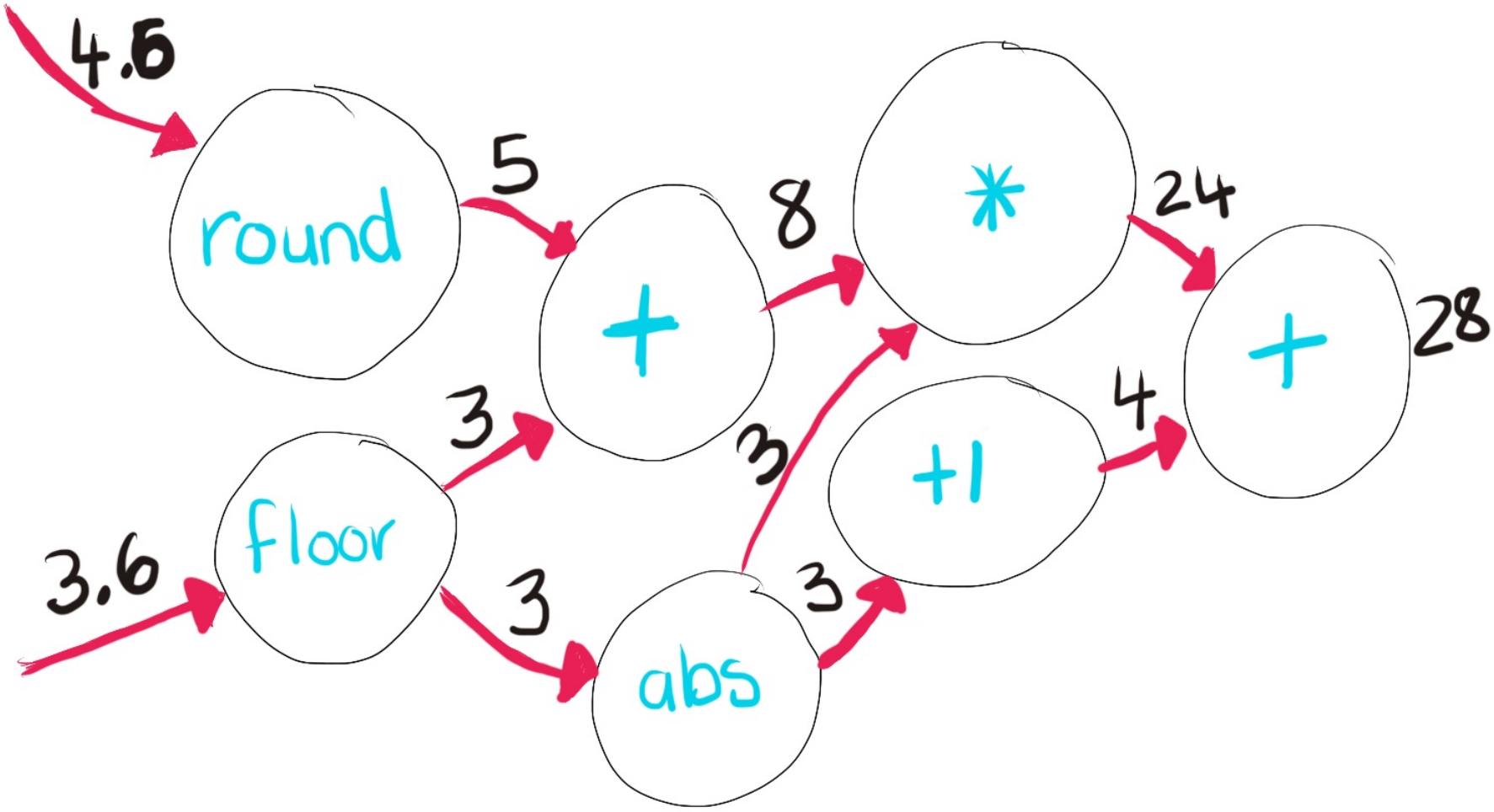
r Develop on Google Pixelbook
We'd like to make it easier for you to develop and test your app by offering you 75% off Google Pixelbook. Redeem your unique code online at: google.com/chromebook/io18 Pick up your discount code
(and socks) before you lepye

Works offline

Custom Tensorflow Models 🐶🌷

TensorFlow

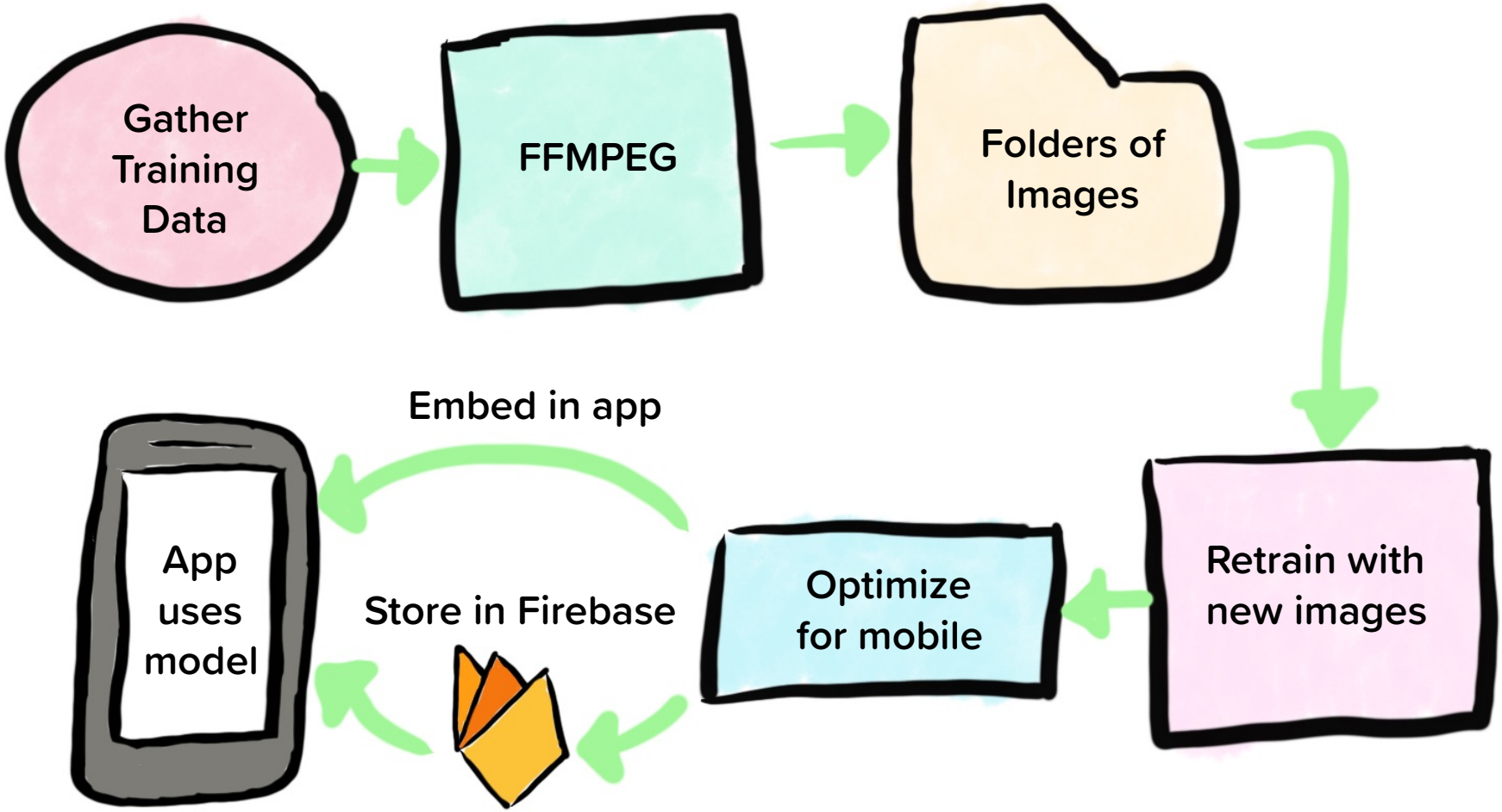



4.6
round
5
+
8
*
24
+
28
3
floor
3.6
3
3
+1
4
3
abs

Retrain existing model 💁‍♀️

`mobilenet_v1`

What if I could tell what kind of chips I was eating? 🤔

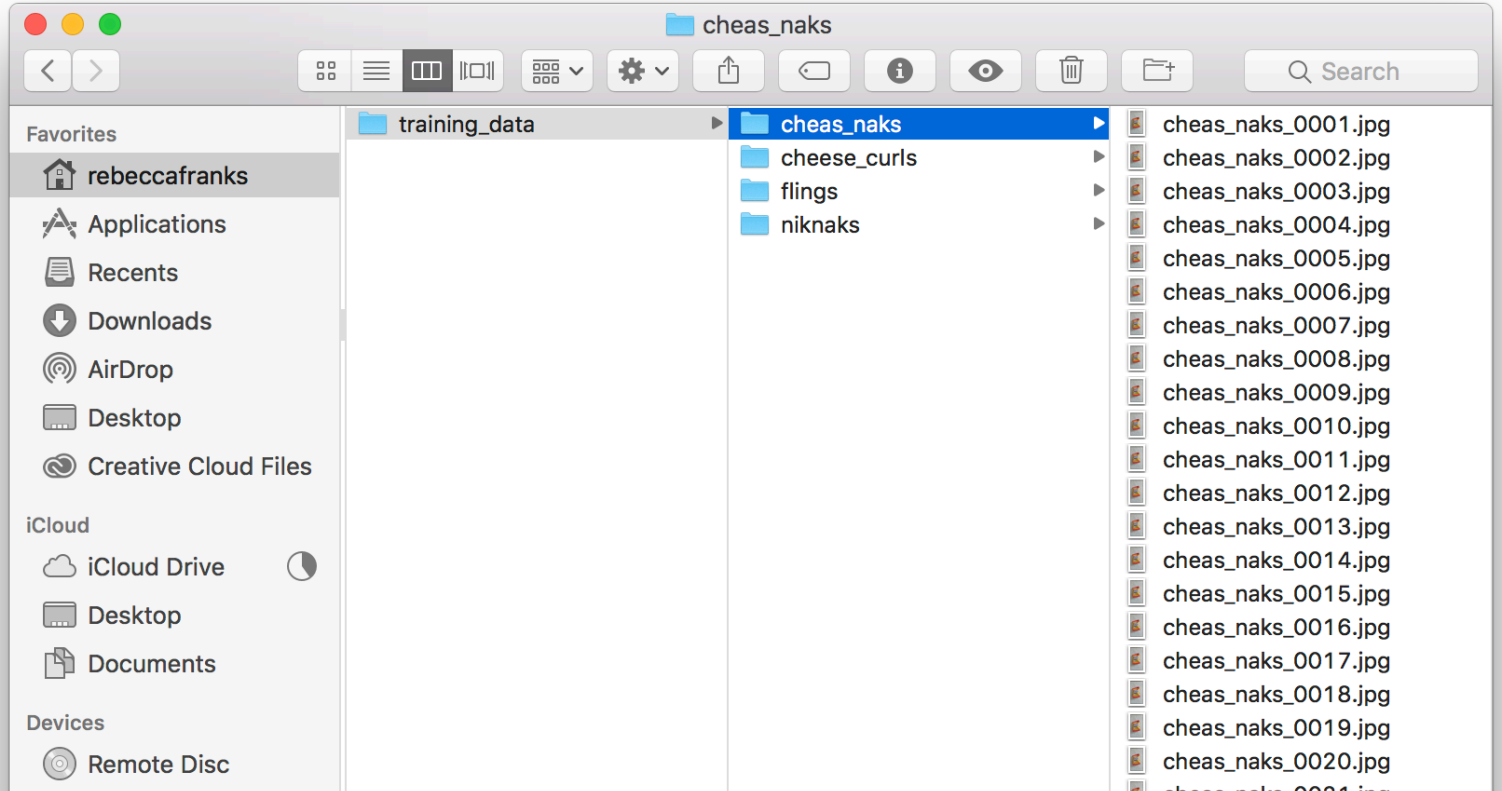

Gather Training Data
FFMPEG
Folders of Images
Retrain with new images
Optimize for mobile
Store in Firebase
Embed in app
App uses model

Gather training data

Export to Images using ffmpeg

```
ffmpeg -i flings.mp4 flings/flings_%04d.jpg
```

Folders of images



Retrain with new images

```
python -m scripts.retrain \
  --bottleneck_dir=tf_files/bottlenecks \
  --how_many_training_steps=500 \
  --model_dir=tf_files/models/ \
  --summaries_dir=tf_files/training_summaries/"${ARCHITECTURE}" \
  --output_graph=tf_files/retrained_graph.pb \
  --output_labels=tf_files/retrained_labels.txt \
  --architecture="${ARCHITECTURE}" \
  --image_dir=training_data/south_african_chips
```

Optimize for mobile

```
bazel-bin/tensorflow/contrib/lite/toco/toco \
  --input_file=AgencyDay/retrained_graph.pb \
  --output_file=AgencyDay/chips_optimized_graph.tflite \
  --input_format=TENSORFLOW_GRAPHDEF \
  --output_format=TFLITE \
  --input_shape=1,${IMAGE_SIZE},${IMAGE_SIZE},3 \
  --input_array=input \
  --output_array=final_result \
  --inference_type=FLOAT \
  --input_data_type=FLOAT
```

Insert into App

https://codelabs.developers.google.com/codelabs/tensorflow-for-poets-2-tflite/
https://github.com/googlecodelabs/tensorflow-for-poets-2

```
/** Classifies images with Tensorflow Lite. */
public class ImageClassifier {

  /** Tag for the {@link Log}. */
  private static final String TAG = "TfLiteCameraDemo";

  /** Name of the model file stored in Assets. */
  private static final String MODEL_PATH = "chips_optimized_graph.tflite";

  /** Name of the label file stored in Assets. */
  private static final String LABEL_PATH = "chips_retrained_labels.txt";
```

Nik Nak 👍🏾
Or
Not 👎🏾

bit.ly/mlkit-riggaroo

What if our model changes? 🤔

Ship an app update 🤷🏻‍♀️

And hope that people download it 🤞🏾

Host on Firebase 🔥

Updates automatically downloaded

```
val cloudSource = FirebaseCloudModelSource.Builder("my_cloud_model")
        .enableModelUpdates(true)
        .setInitialDownloadConditions(conditions)
        .setUpdatesDownloadConditions(conditions)
        .build()


FirebaseModelManager.getInstance()
                .registerCloudModelSource(cloudSource)

……
```

g.co/codelabs/mlkit-android-custom-model

You don’t need to be a ML Expert to take advantage of ML in your apps!

🧠

Thank you!

Resources

- https://codelabs.developers.google.com/codelabs/tensorflow-for-poets
- https://codelabs.developers.google.com/codelabs/tensorflow-for-poets-2-tflite/
- https://codelabs.developers.google.com/codelabs/mlkit-android-custom-model/#0
- https://github.com/riggaroo/android-demo-mlkit