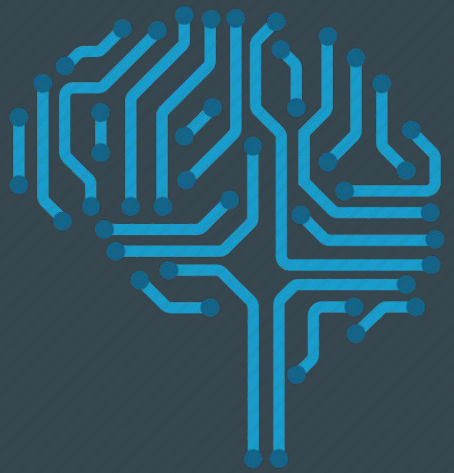
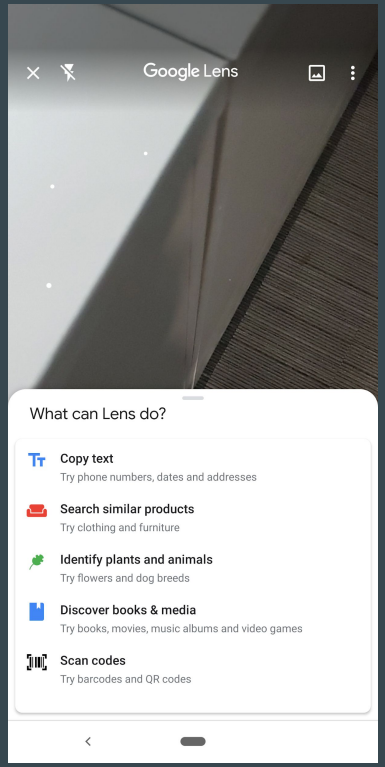
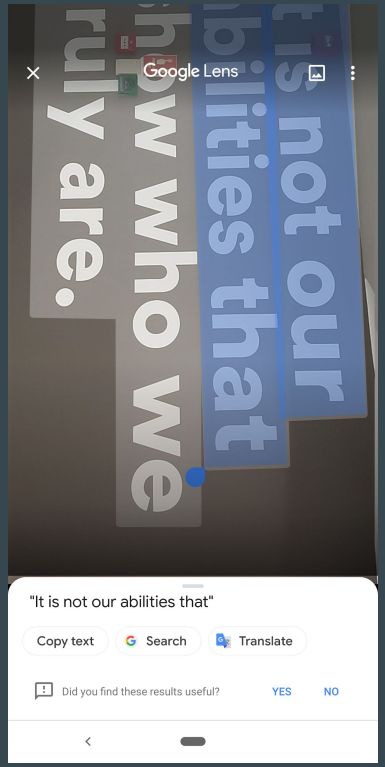
On-Device Machine Learning

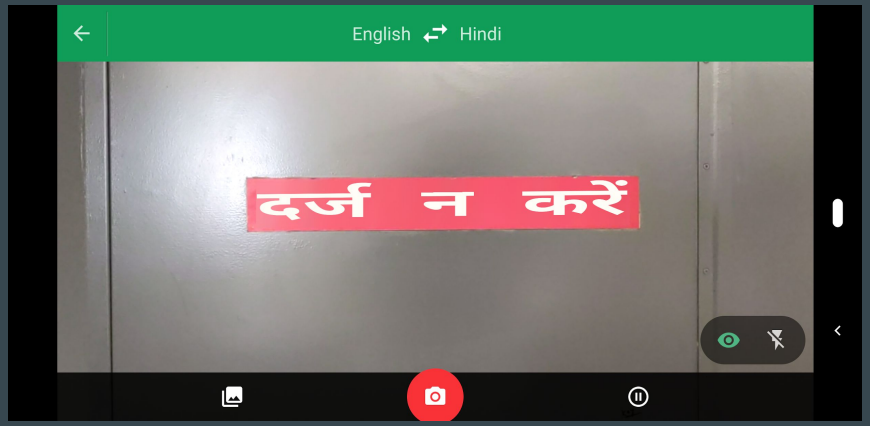
...



What is machine learning







ML Applications

Image labeling

Identify objects, locations, activities, animal species, products, and more

Text recognition (OCR)

Recognize and extract text from images

Face detection

Detect faces and facial landmarks

Barcode scanning

Scan and process barcodes

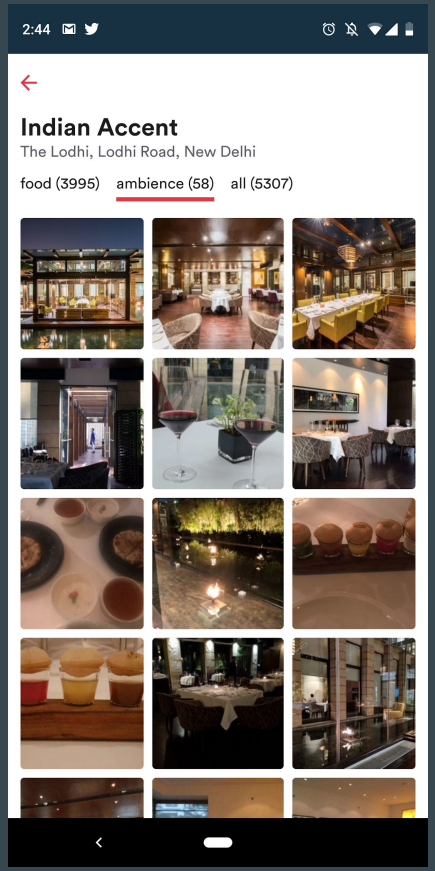
Landmark detection

Identify popular landmarks in an image

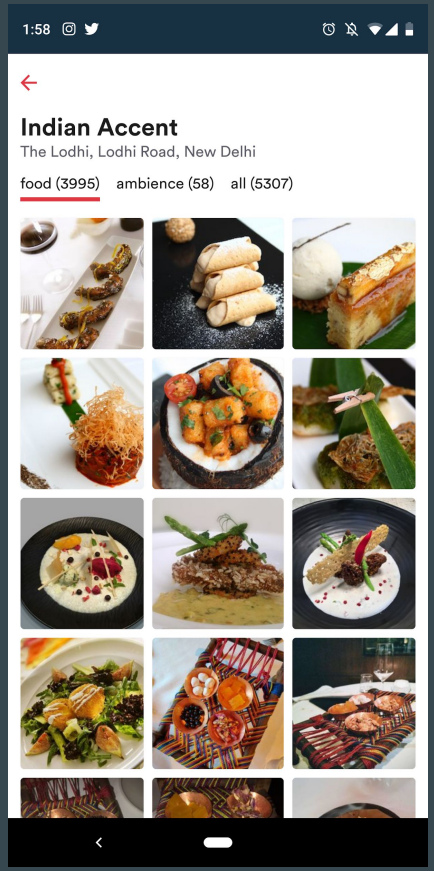
Smart reply (coming soon)

Provide suggested text snippet that fits context

ML Kit



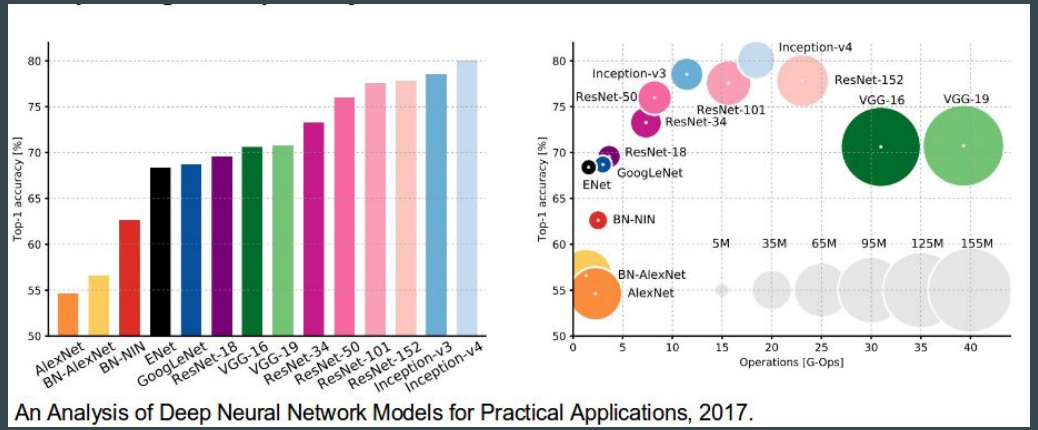
zomato



User Generated Content

Connecting the dots

- Image Classification
- ImageNet
- Large Scale Visual Recognition Challenge (ILSVRC)
- MobileNet



An Analysis of Deep Neural Network Models for Practical Applications, 2017.

MobileNet

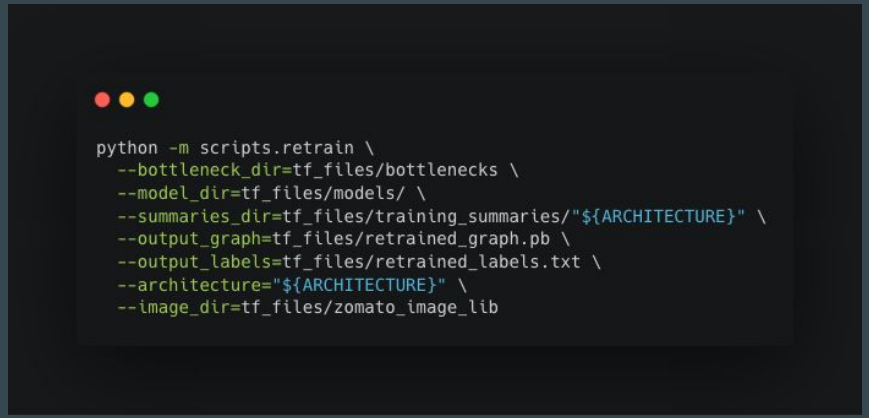
- Runs quickly
- High accuracy
- Reasonable performance in resource-constrained environment
- small, low-latency, low-power model

Table 8. MobileNet Comparison to Popular Models

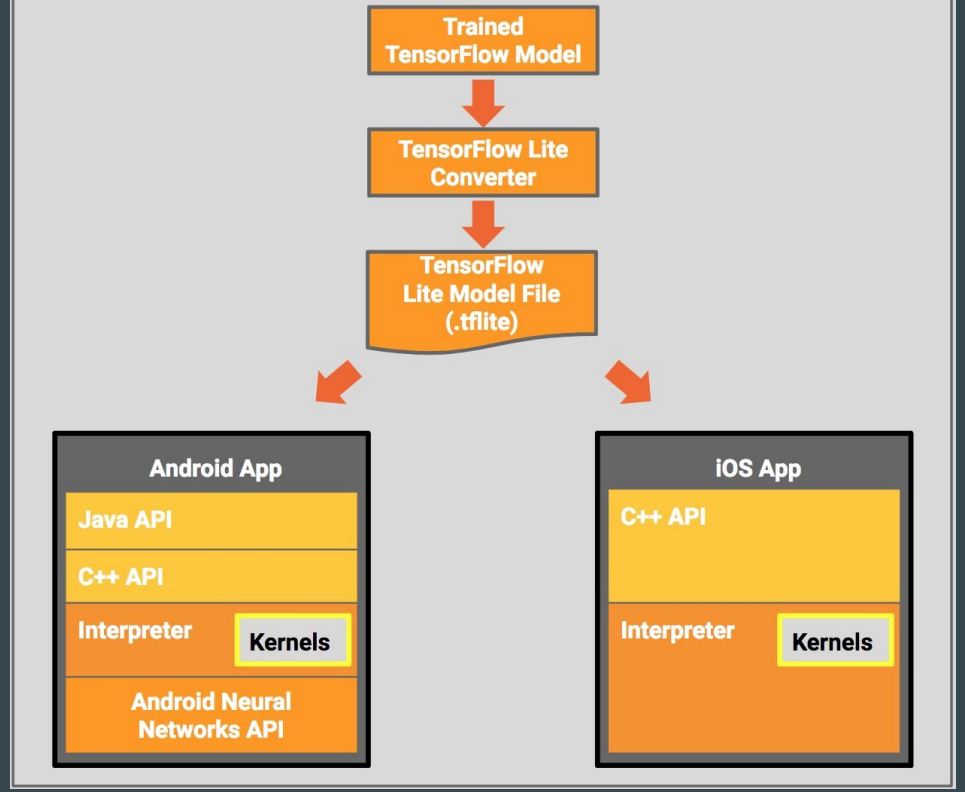
Model	ImageNet Accuracy	Million Mult-Adds	Million Parameters
1.0 MobileNet-224	70.6%	569	4.2
GoogleNet	69.8%	1550	6.8
VGG 16	71.5%	15300	138

Re-Training the model

- Training models from scratch requires a lot of labeled training data and a lot of computing power
- Transfer learning - Shortcut
- Re-use the feature extraction capabilities
- Train just the last classification layer
- Surprisingly effective - works with moderate amounts of training data (thousands, not millions of labeled images)



Architecture



Converting to TFLite Model

```
IMAGE_SIZE=224
tflite_convert \
  --graph_def_file=tf_files/retrained_graph.pb \
  --output_file=tf_files/optimized_graph.lite \
  --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
```

Android Interpreter

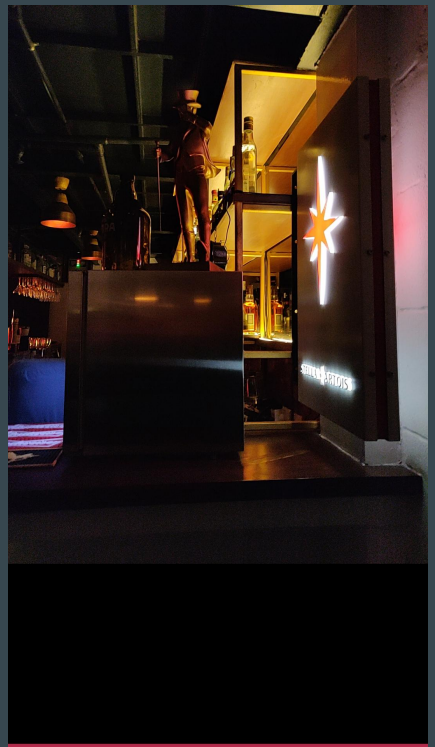
```
ImageClassifier(Activity activity) throws IOException {
    //This line instantiates a TFLite interpreter. The interpreter is basically the orchestra
    conductor.
    //It takes in MappedByteBuffer containing the model.
    tflite = new Interpreter(loadModelFile(activity));

    // Create input buffer.
    imgData =
        ByteBuffer.allocateDirect(
            4 * DIM_BATCH_SIZE * DIM_IMG_SIZE_X * DIM_IMG_SIZE_Y * DIM_PIXEL_SIZE);
    imgData.order(ByteOrder.nativeOrder());

    // Create label list and output buffer.
    labelList = loadLabelList(activity);
    labelProbArray = new float[1][labelList.size()];
    Log.d(TAG, "Created a Tensorflow Lite Image Classifier.");
}
```

Android Interpreter

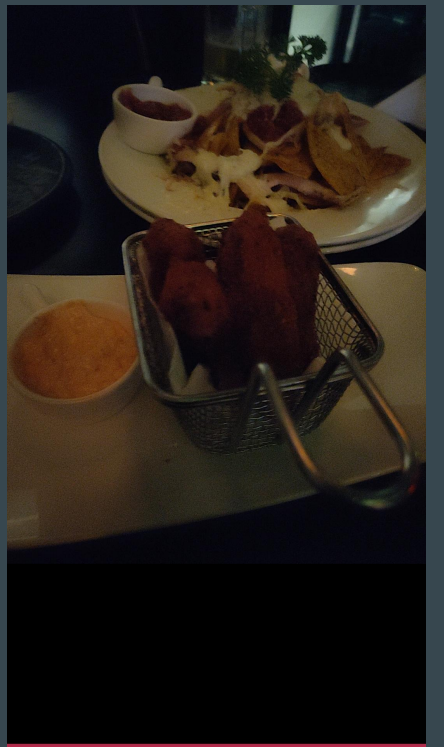
```
String classifyFrame(Bitmap bitmap) {
 //Convert and copies the input Bitmap to the imgData ByteBuffer
 convertBitmapToByteBuffer(bitmap);
 //The interpreter sets the values in the output array to the probability calculated for each class.
 tflite.run(imgData, labelProbArray);
 //Profit?
 String textToShow = printTopKLabels();
}
```



AMBIANCE

22ms
ambiance: 1.00
food: 0.00
human: 0.00
menu: 0.00

Using this demo app with our custom model we can perform real time image processing (20-50 ms)



FOOD

23ms
food: 1.00
menu: 0.00
ambiance: 0.00
human: 0.00

Taking it for a spin

Custom Model Demo

On to the other half of our problems..

JobScheduler

```
private void scheduleCameraJob(Boolean Immediate) {
       final Uri MEDIA_URI = Uri.parse("content://" + MediaStore.AUTHORITY + "/");

       JobInfo.Builder builder = new JobInfo.Builder(11,
               new ComponentName(this, PhotosContentJob.class.getName()));
       // Look for specific changes to images in the provider.
       builder.addTriggerContentUri(new JobInfo.TriggerContentUri(
               MediaStore.Images.Media.EXTERNAL_CONTENT_URI,
               JobInfo.TriggerContentUri.FLAG_NOTIFY_FOR_DESCENDANTS));
       // Also look for general reports of changes in the overall provider.
       builder.addTriggerContentUri(new JobInfo.TriggerContentUri(MEDIA_URI, 0));

       if (Immediate) {
           // Get all media changes within a tenth of a second.
           builder.setTriggerContentUpdateDelay(1);
           builder.setTriggerContentMaxDelay(100);
       } else {
           builder.setTriggerContentUpdateDelay(1);
           builder.setTriggerContentMaxDelay(100);
       }

       JobInfo myCameraJob = builder.build();
       JobScheduler jobScheduler = (JobScheduler) getSystemService(Context.JOB_SCHEDULER_SERVICE);
       int result = jobScheduler.schedule(myCameraJob);
       if (result == JobScheduler.RESULT_SUCCESS) {
           Log.e("JobScheduler", " JobScheduler OK");
       } else {
           Log.e("JobScheduler", " JobScheduler fails");
       }
   }
```

JobScheduler

```java
@Override
public boolean onStartJob(JobParameters params) {

    // Check if triggered due to content change
    // Iterate through them and collect IDs (Specific files)
    // If we found some ids that changed, we want to determine what they are.
    // Query with content provider to ask about all of them.
    // Finally check if these are from DCIM - collect all loop through them.

    s1 = mClassifier.classifyFrame(Bitmap.createScaledBitmap(myBitmap,224,224,false));
    String[] split = s1.split("\n");

    if (split[1].contains("food"))
        renderNotification(s, "food");

    if (split[1].contains("ambiance"))
        renderNotification(s, "restaurant");

    if (split[1].contains("menu"))
        renderNotification(s, "menu");
}
```

Demo

Thank You

Google Codelabs

- https://codelabs.developers.google.com/codelabs/tensorflow-for-poets-2-tflite/
- https://codelabs.developers.google.com/codelabs/tensorflow-for-poets/index.html

ML Literature

- http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture9.pdf
- https://arxiv.org/pdf/1704.04861.pdf
- https://www.cs.unc.edu/~wliu/papers/GoogLeNet.pdf
- https://keras.rstudio.com/reference/application_mobilenet.html

Job Scheduler

- https://developer.android.com/reference/android/app/job/JobScheduler

 : @SajalGupta94