Cereal Classifier: TensorFlow Android App

Ekaba Bisong[a]

Carleton University, School of Computer Science
Presented to: Machine Learning & Artificial Intelligence Ottawa, Sep. 25 2017

[a]M.Sc. Candidate
https://ekababisong.org/

Presentation Outline

Background

Background

- Training an image recognition model from scratch is non trivial.
- The major constraint is the processing and computing cost.
- Leveraging pre-trained models build on the shoulders on engineering man-hours.
- Run a pre-trained model on a pre-made Android app.
- Limit engineering overhead to solve peculiar recognition use cases.

The Complexity of Image Recognition

- The time cost of training a recognition model from scratch.
- Deep and wide Convolutional Neural Networks are expensive.
- Architectural rationale for conv-nets design might be daunting for non-academic enthusiast.
- Well labeled datasets are challenging and costly to put together.

Leveraging Pre-trained Models

- Transfer learning: building on acquired knowledge to generalize to new examples.
- Cuts down development time, and opens the door for various products to incorporate object recognition technology
- It is common practice to share weights of pre-trained models
- See Caffe library - Model Zoo

Retraining the Model

Pre-trained Image Models

- **InceptionV3:** is a pre-trained image classifier to classify 1000 image classes from the ImageNet Large Vision Recognition dataset in 2012.
- **MobileNet:** a mobile-optimized pre-trained image model
- InceptionV3 is optimized for accuracy whereas MobileNet for efficiency on low-memory machines

TensorFlow

- Machine Learning open-source library optimized for Deep learning tasks
- Computations are represented as stateful graphs
- Tensorflow Lite now available for Android development as of May 2017.

The Retrain Script

- Packaged as part of TensorFlow open-source repository
- See - Transfer learning with InceptionV3 or MobileNet
- Broad application range (details in the paper).

Bottlenecks

- The last or final layer before the output layer.
- Retraining begins at the bottleneck graph, with all the other layers retaining their pre-trained state.
- Final layer called bottleneck, because learned representation is *fully connected* at this layer.

Optimizing for Mobile

Pruning Unsupported Operations

- Mobile platforms are not built for computational graph processing

- TensorFlow Lite or TensorFlow Mobile only supports selected operations relevant for running trained models

Optimize Trained Graphs

- TensorFlow script called optimize_for_inference removes graphs with unsupported operations.
- "Removes parts of a graph that are only needed for training."
- See script at TensorFlow GitHub repo

Compress the Model

- Useful to reduce download size of app
- Use gzip utility to compress model graph
- We can do better by first quantizing network weights!

Quantize the Model

- To quantize simply means to reduce the number of floating points of a networks weight hence making them smaller, which is ideal for mobile apps.

- TensorFlow comes pre-packaged with a script called quantize_graph to carry out this action on a trained model.

- Re-compress the model with gzip.

Cereal Classifier App

About the App

- The application is trained to classify the following cereals: apple cinnamon cheerios, berry kix, chocolate cheerios, cocoa puffs, honeycomb, kelloggs all bran, kelloggs corn pops, kelloggs frosted flakes, kelloggs rice krispies, multi grain cheerios, and sugar crisp.

- Click here to view on Google App Store

Data Collection Process

- Took a 1 minute recording of cereals packets.
- Broke into frames with `ffmpeg` utility.
- Placed into folder with label names for retraining with `InceptionV3`

Stub TensorFlow Android

- Download stub TensorFlow Android application at:
  TensorFlow Android Camera Demo

Infusing your Trained Model

- In the `assets` folder of the downloaded stub Android app directory, place your trained weights and text file containing labels there.

Cereal Classifier App Directory
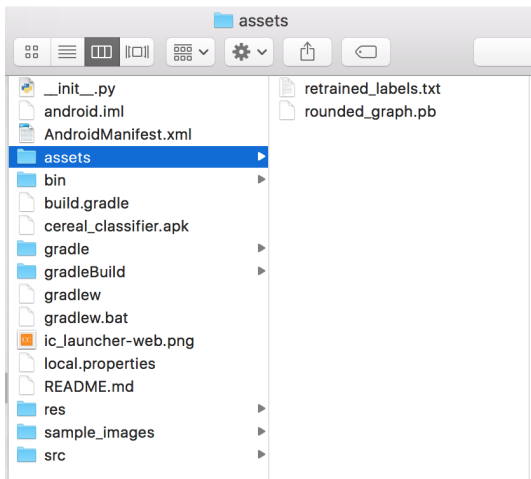
Figure 1: Cereal Classifier App Directory

Future Work

Expanding the classified products and adding speech recognition (using Speech API) to call out items during grocery shopping with the aim of easing shopping or enabling shopping for the visually impaired. That's the concept of the final product.
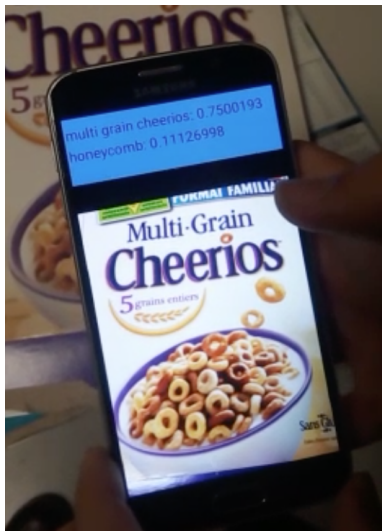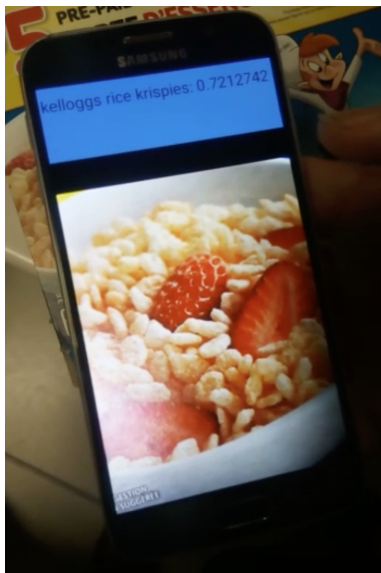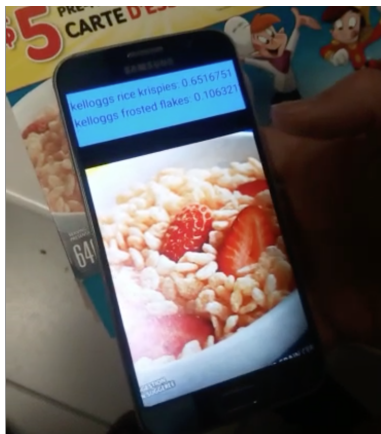
Demo Images

Figure 2: Demo Image A

Figure 3: Demo Image B

Figure 4: Demo Image C

Figure 5: Demo Image D

Resources

- TensorFlow for Poets: Retraining an Image Model.
- TensorFlow for Poets 2: Running TensorFlow on Mobile.
- InceptionV3 Paper: Going Deeper with Convolutions.