Produce Pal

Developer

Kathryn LaVigna

katielavigna@gmail.com

Application Summary

Have you ever been to a farmer's market or specialty grocery store and seen an item and thought, "what kind of strange looking fruit/vegetable is that?" Gone are the days of having to search Google Images for *spikey broccoli* (that's romanesco) or *giant green grapefruit* (some might call this a pummelo) to figure out what the real name for that item is. With Produce Pal, simply use your iOS device's camera to snap a photo of your item, or upload one from your camera roll, and within seconds the app will give you the two most likely classifications of that fruit or vegetable.

Technologies Used

- Google Colab – Hosted Jupyter Notebook with access to free GPU and TPU. Used to create train/test datasets and for training of model.
- Keras – Deep learning API with a Python interface. Acts as an interface for the machine learning platform Tensorflow. Used transfer learning with the pre-trained MobileNet for training of Produce Pal model.
- Xcode – Apple's integrated development environment used to develop software for iOS devices. Used to build the Produce Pal application software.
- coremltools – A Python package that allows easy conversion of third-party models to the Core ML format for integration of machine learning models into iOS apps. Used coremltools to convert Keras model to Core ML format.

Technical Details

Produce Pal is an edge-based computer vision application which aims to classify different types of produce. The classifier works best on fresh, whole fruits and vegetables.

Produce Pal's underlying model was created utilizing transfer learning with MobileNet as the base model. MobileNets are a class of efficient models designed with a streamlined architecture for building lightweight deep neural networks. This model has been trained on more than one million images from the ImageNet database, and is able to classify images into 1000 different categories, including types of fruits and vegetables. We use transfer learning in order to utilize the rich feature representations of the MobileNet model which have already been developed over training on millions of other images. This reduces the computing power and size of dataset needed to train a custom produce classifier.

The dataset behind Produce Pal contains approximately 100 images from 45 classes of fruits and vegetables, with a focus on inclusion of obscure produce that the average person is less likely to be familiar with (e.g. pummelo, Romanesco, donut peach, rutabaga, etc.). Training a CNN from scratch on this size dataset is unlikely to be successful, which was the motivation behind transfer learning. The use of pre-trained weights in the MobileNet model allows us to create a custom model for this specific problem, while drastically reducing the time and complexity of the model training process. After training over 25 epochs, the model reached 71% testing accuracy with 1.10 loss.

Produce Pal was developed using Xcode to allow the classifier to run locally on an iOS device. The classifier was built as a Keras model but needs to be converted to CoreML. CoreML is a machine learning framework created by Apple for the integration of machine learning models into iOS applications. This was done using Apple's coremltools python package.

GitHub Repository

https://github.com/katielavigna/produce_pal