Introduction

Food image classification has many uses in everyday tasks. Users may want to find a picture of a specific dish online to display in social media, or upload a picture to a social network and have it suggest the name of the food based on the image. Classifying images of food also has use in meal tracker applications by improving the speed at which the user can log their meals.

In this project, we explore the complexities associated with training a neural network to perform food classification. Our final model uses an InceptionV3 network with weights pre-trained from ImageNet, which we incrementally refine on our dataset (Figure 1).

Dataset

- The initial dataset is Food-101, which contains 101 classes with 1000 images per class.
- We added an additional 4000 images from our custom web-crawler script.
- We pre-processed each image before training to be a standard 224x224 size.
- We split the dataset using a 90/5/5 training/dev/test split, resulting in 9,980/505/505 images respectively.
- We used on-the-fly data augmentation techniques to make the most of our small dataset (Figure 2).

Outcomes

- SGD had better performance than Adam/RMSProp optimizers (Figure 3), and momentum had a noticeable impact on validation accuracy (Figure 4).
- We could only achieve around 66% accuracy on our test set.
- Visualizing the output filters of the last convolutional layer does not show any food-specific features. It is likely that the network has not learned enough features to make accurate predictions (Figure 5).
- In error analysis (Figures 6, 7, 8), we found the top 10 most accurately predicted classes tend to have simple, consistent shapes and well-defined edges, while the bottom 10 have traits of variety in the presentation.
- Future work would include expanding the dataset, committing to longer training times, trying different optimizers and deeper CNN architectures.

Methods

Figure 1: InceptionV3 architecture used for our final model. 314 layers deep and 22 million trainable parameters. Customized to have 150x150x3 input features, 101 output classes.

Figure 2: On-the-fly data augmentation samples. We used this technique to try to reduce our variance gap.

Analysis

- Optimizer: SGD, LR=1.0, Momentum=0.9, L2 = 0.0001, Val Acc = 0.7367
- Adam, LR=1.0, Momentum=0.8, L2 = 0.0001, Val Acc = 0.7367
- RMSProp, LR=1.0, Momentum=0.9, L2 = 0.0001, Val Acc = 0.7367
- We can see that SGD had a larger impact than Adam or RMSProp optimizers.

Figure 3: Optimizer search. SGD had larger impact than Adam or RMSProp optimizers.

Figure 4: Effect of momentum on the validation accuracy.

Figure 5: Visualization of the last layer of filters in the CNN. It is hard to discern meaningful food image features being learned by the network.

Figure 6: Accuracy of top and bottom 10 performing classes.

Figure 7: Top 10 performing classes from left to right: edamame, miso soup, pho, lobster bisque, crème brûlée, spaghetti carbonara, seaweed salad, oysters, hot and sour soup, dumplings. Note the simple, consistent shapes.

Figure 8: Bottom 10 performing classes from left to right: pork chop, ceviche, steak, omelette, foie gras, bread pudding, apple pie, tuna tartare, huevos rancheros, chocolate mouse. These classes have a wide range of different presentations.