[Deep Dish] Food Image Reorganization using CNN

Haichen Liu Abhishek Goswami

BACKGROUND

- When searching for a restaurant on Yelp or TripAdvisor, people may see images of some food that they are interested in.
- However, they do not know what they are or whether they taste as good as they look.
- Hence, Deep Dish is using CNN to recognize different types of food or dishes, so that people can know the name, ingredient, and even the taste.

PROBLEM STATEMENT

- Deep Dish starts from recognizing the names of 15 different types of food coming from America, Italy, Japan, China by the corresponding images.
- In Phase 2, the Deep Dish will use unsupervised learning to recognize food from a Yelp review images with people or other background objects.

DATASETS

- The images are coming from ImageNet and Flickr. We select a set of images of relatively good quality, i.e. with good focus, less background, and being typical.
- Positive examples

This is a clear image for a typical fish & chips dish

- Negative examples
  a. Image is not focus on the tempura dish
  b. Too much background
  c. Not a typical curry lamb

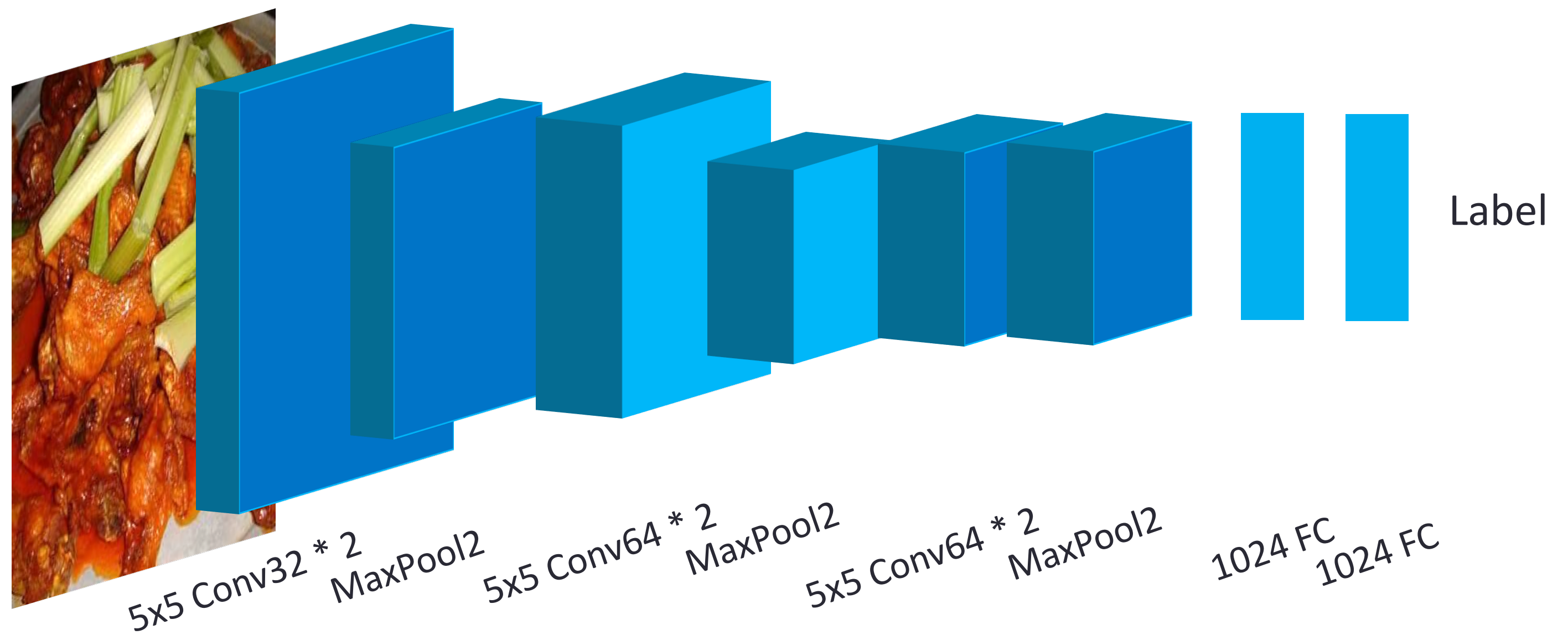
a

b

c

METHODS/ALGORITHMS/MODELS

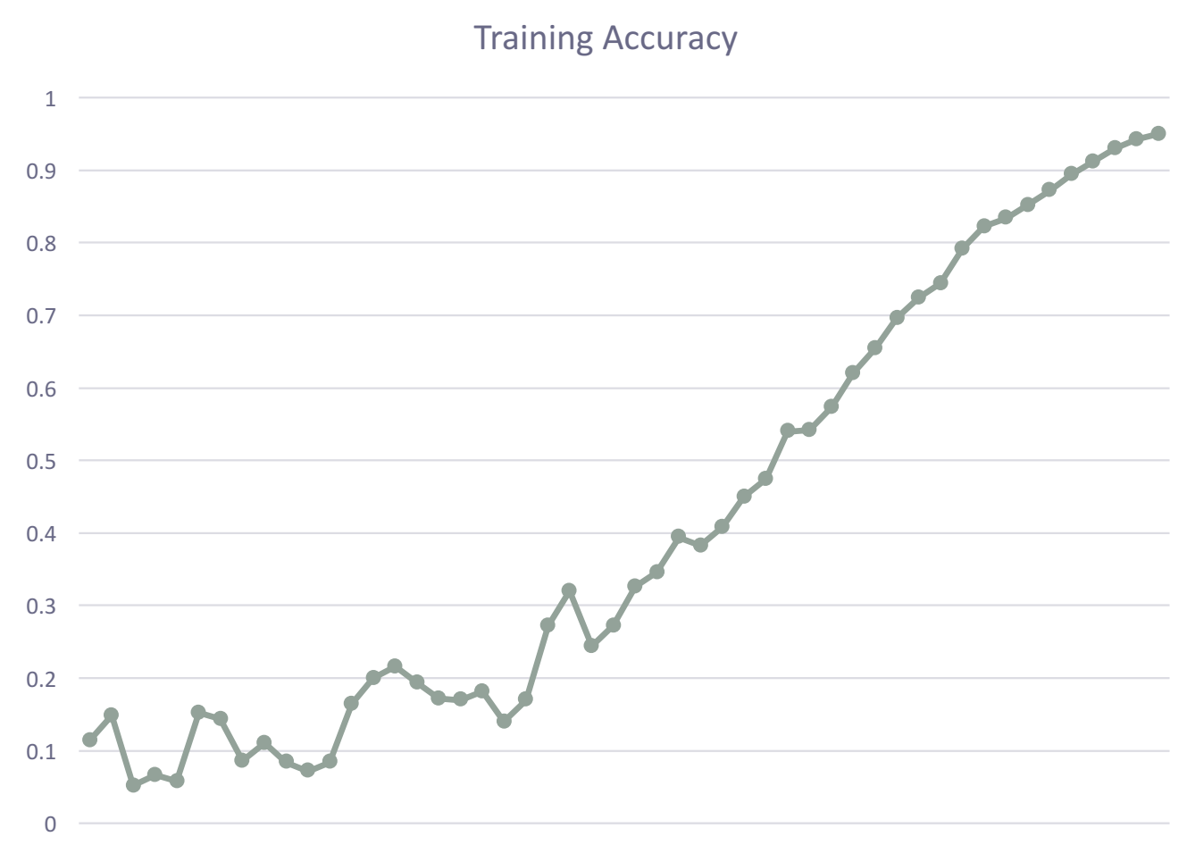
- We tried two models besides the baseline. Both of them are similar to VGG16 but less of numbers of filters in Conv layers, and less of numbers of parameters in dense layers due to the hardware limit of our instance.

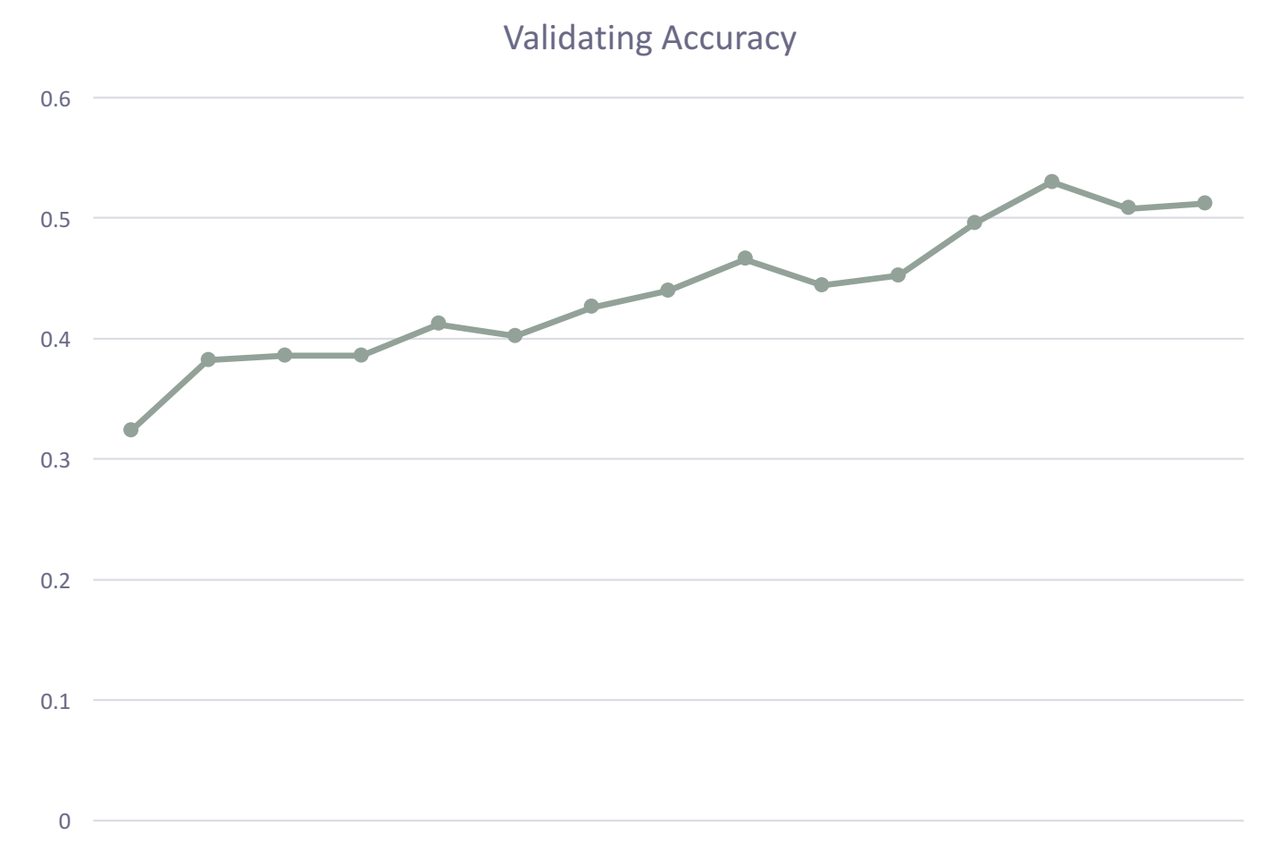


- The size of the 4th and 5th Conv layer of VGG16 should be 128, but due to limit of the instance, we reduce the numbers of filters to 64.
- As noticed, the accuracy of training set achieved over-fit. To reduce the over-fit, the pre-process is added to gray out a subset of the training images.

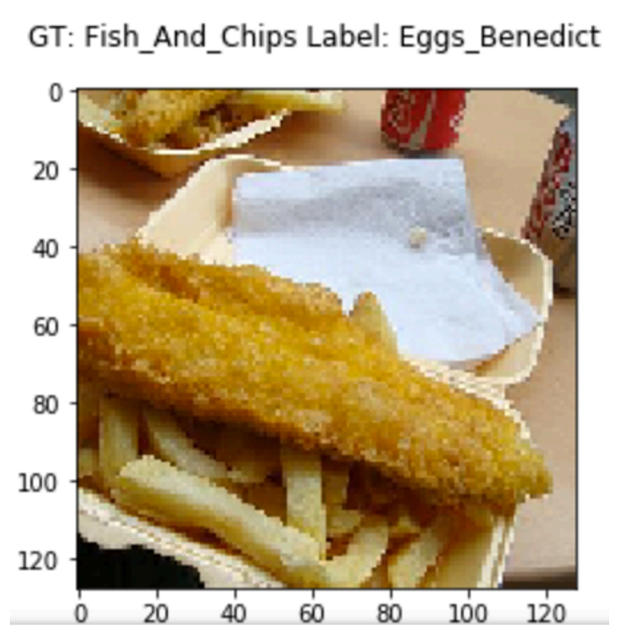
EXPERIMENTAL EVALUATION

- After running 15 epochs, the basic model without any augmentation achieves an accuracy of 51.2% on validating set, while the training sets are over-fitted.
- By printing out the mistakes, we found several error patterns. The model cannot differentiate the food if the color is similar. Here are several examples.
  - Image d has fish and chips, but our model recognizes it as egg Benedict. It makes this mistake because the color of fried fish is similar with the source of the egg Benedict, and that the napkin looks like the egg.
  - Image e is pepperoni pizza, but the model recognizes it as sashimi. The reason could be that sashimi has red and white pieces on it.

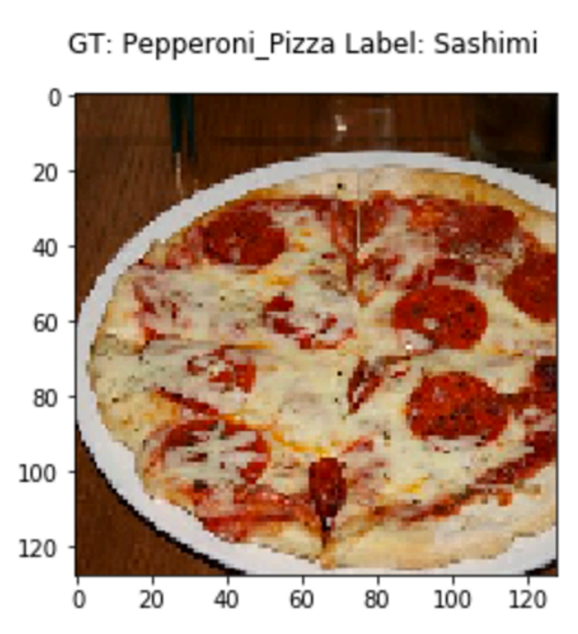




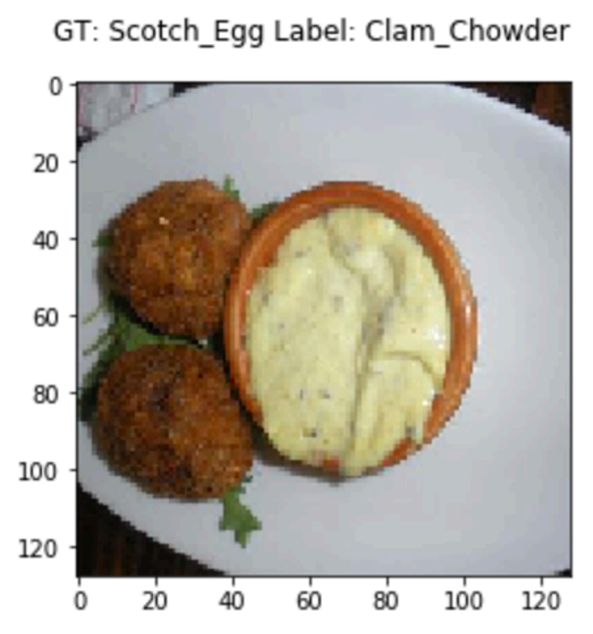
- After this, we try to use data augmentation to reduce the over-fit, and the systematic error. We add a gray-out process for a subset of the images, in order to train the model to recognize the images by using the texture or shape instead of color.
  - after running for 15 epochs, the model achieves an accuracy of 68.2% for validating set.
  - Then by printing out the mistakes, we found some of the mistakes make some sense, while the others not.
  - The first image has Scotch eggs, but the model recognizes it as clam chowder. However, the dip does look like a bowl of chowder.
  - The second image has scotch eggs, but the model recognizes it as cannelloni, which makes none sense



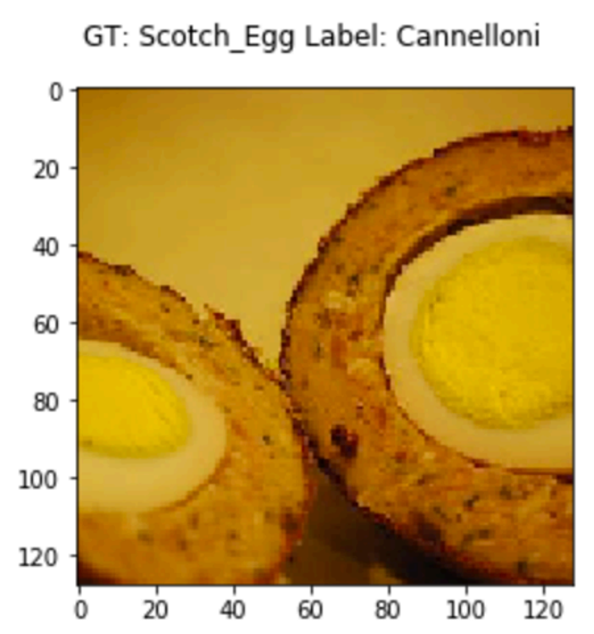
d



e



f



g

CONCLUSIONS & FUTURE WORKS

- The validating accuracy cannot have significant increase after tuning the hyper parameters. The accuracy is floating around 65% for validating set.
- The simplified VGG16 model have the potential to deal with this problem. However, to recognize the food image requires the model to be able to detect the differences of the shape, color and texture of the small pieces of food materials.
- We will try another set of training images of higher quality.
- Next, we will try HOG to highlight the edges of the food, and use it as a feature input to our model.
- Also if we have time, we want to improve our model so that it can deal with images with complex background.