Classifying food images to generate recipes and calorie estimation

Overview

Today, people on Instagram and other forms of social media post pictures of amazing food, which make your mouths water for them. These foods are from different categories and contain different amounts of calories. One thing we all crave is to be able to have these different delicacies for ourselves. But while doing so, we are thinking about whether these foods fit our dietary constraints or meet our calorie consumption limits. We also worry about the fact that we cannot don't know how to make these delicacies.

By providing me with the food image, I plan on giving the food classification, and calorie estimate, and generating a recipe for you. With this information, you will be able to make an informed decision on making the item or not. You will also be able to easily and efficiently replicate the image from Instagram and appeal to the masses!

This can help solve many different problem statements. For example, if one was traveling abroad, they could use this to determine if the item they were having was vegetarian or not. It could also help them catalog the item to make once they are back home. Another example would be if someone was in a restaurant and wanted to know the recipe on how to make a certain dish, they could use this to save the recipe for later. Finally, they could use the application to determine if they were within their calorie consumption limit if they were dieting. The applications of this project can be vast and will provide insightful information, allowing the user to make an informed and thorough decision.

This project is focused on solving three major problem statements- Food classification from the image, recipe generation, and calorie count estimation. I will complete this by using computer vision to model and answer the questions that I have. I will be utilizing many different sources to help improve my model and increase the performance of my model.

Literature Survey

The following papers will help me solve my problem statement, using food images to determine the food and the ingredients that make up the food.

- Recipe Generation:
    - This paper talks about the usage of food images to give the recipe required to make the item for yourself.
    - https://paperswithcode.com/paper/inverse-cooking-recipe-generation-from-food
- Food Classification:
    - This paper talks about the usage of food images to determine what the food item is.
    - https://paperswithcode.com/paper/foodx-251-a-dataset-for-fine-grained-food
    - Another paper:
    - https://github.com/thatbrguy/Multilabel-Classification
    - https://nanonets.com/blog/multi-label-classification-using-deep-learning/
- Calorie Estimation:
    - Using food images, this paper talks about how to calculate the number of calories in the item presented.
    - https://paperswithcode.com/paper/deep-learning-based-food-calorie-estimation
    - Training code in Python: https://github.com/meghanamreddy/Calorie-estimation-from-food-images-OpenCV
- Deep Dish Learning to Classify Food Dishes:
    - The following article uses deep learning to classify the dish properly from across the world
    - http://cs231n.stanford.edu/reports/2017/pdfs/6.pdf
- How to easily classify Food using deep learning and TensorFlow
    - In this paper, they use TensorFlow and deep learning techniques to be able to classify the food
    - https://nanonets.com/blog/multi-label-classification-using-deep-learning/
- Feature extraction from a trained model:
    - I used this paper to determine how I could use a pre-trained model to my advantage, given the large data size.
    - https://appliedmachinelearning.blog/2019/07/29/transfer-learning-using-feature-extraction-from-trained-models-food-images-classification/
- iFood Challenge Winner code:
    - https://www.kaggle.com/c/ifood-2019-fgvc6/discussion/94425
    - https://github.com/clovaai/assembled-cnn

Dataset

I will utilize the datasets from each of the papers as well as external datasets to improve the accuracy of models

- https://www.lftechnology.com/blog/ai/image-calorie-estimation-deep-learning/
- https://www.ivl.disco.unimib.it/activities/food-recognition/
- https://github.com/Yiming-Miao/Calorie-Predictor
- https://data.world/adamhelsinger/food-nutrition-information
- https://github.com/karansikka1/iFood_2019
- http://im2recipe.csail.mit.edu/dataset/download

These datasets contained information regarding the different images that were used for the food classification. They also contain text information regarding the recipes, calorie counts, and other written information about the images and food.

Downloads

README
Layers (381 MiB)
Ingredient detections (102 MiB)
Recipe1M images
- training (94 GiB)
- validation (21 GiB)
- test (20 GiB)

Recipe1M+ images
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f (approx. 210 Gib each tar file)

Layer2+
- layer2+.json (2.5 GiB)

Model training files
- data.h5.gz (89 GiB, recipe1M)
- vocab.bin.gz (33 MiB)
- classes1M.pkl (26 MiB)
- train.tar (24 GiB, recipe1M)
- val.tar (5 GiB, recipe1M)
- test.tar (5 GiB, recipe1M)
- encs_train_1024.t7 (29 GiB)
- encs_val_1024.t7 (6 GiB)
- encs_test_1024.t7 (6 GiB)
- remove1M.txt (133 KiB)

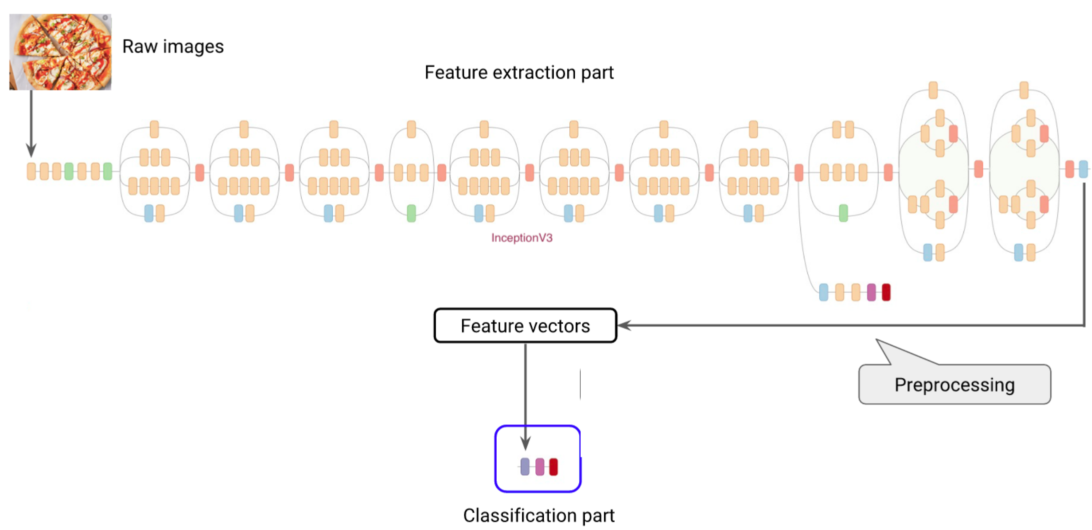
Recipes with nutritional information
- recipes_with_nutritional_info.json (213 MiB)

I had a large dataset that consisted of over 100 GB, this included recipes, calorie estimations, and food images that I could use to train my model.

Algorithm

The fundamental algorithms are classification, object detection, and segmentation. The classification algorithm will help classify and organize the acquired images. Object detection will contribute to

The food classification problem is a prediction resolved with a multi-layer resnet-101 network with Adam optimizer. Recipe generation problem is resolved by three major sub-networks as Food understanding, multi-label classification, and conditional text generation(NLP). Extract the image representation with resnet-50 encoder and obtain the ingredients. My recipe generation system takes a food image as an input and outputs a sequence of cooking instructions, which are generated by means of an instruction decoder that takes as input two embeddings. The first one represents visual features extracted from an image, while the second one encodes the ingredients extracted from the image. The cooking instruction transformer nlayer has concatenated attention, independent attention, and sequential attention. The instruction decoder is composed of two attention layers and one linear layer which is followed by softmax nonlinearity. The multi-label classification model is trained in a mini-batch size of 300, and a learning rate of 0.0001. The instruction generation model is trained with a batch size of 256, Adam optimizer, dropout ratio of 0.3, and 400 epochs. The calorie count estimation problem is resolved with RNN. There are five major steps in the execution - Image Acquisition, object detection, image segmentation, volume estimation, and calorie estimation. Image segmentation is resolved with the Grabcut algorithm. In order to estimate the calories, we need to resolve the scaling factor first. From the reference of one yuan coin paper, the volume is computed. A calorie is estimated from the food mass and density value.



Architecture for InceptionV3 Transfer learning

Food Classification

In the food classification problem, I had to classify 251 fine-grained food categories.

```
0 macaron
1 beignet
2 cruller
3 cockle_food
4 samosa
5 tiramisu
6 tostada
7 moussaka
8 dumpling
9 sashimi
10 knish
11 croquette
12 couscous
13 porridge
14 stuffed_cabbage
15 seaweed_salad
16 chow_mein
17 rigatoni
18 beef_tartare
19 cannoli
20 foie_gras
21 cupcake
22 osso_buco
23 pad_thai
24 poutine
25 ramen
26 pulled_pork_sandwich
27 bibimbap
28 chicken_kiev
29 apple_pie
30 risotto
```

fig: Sample food category list

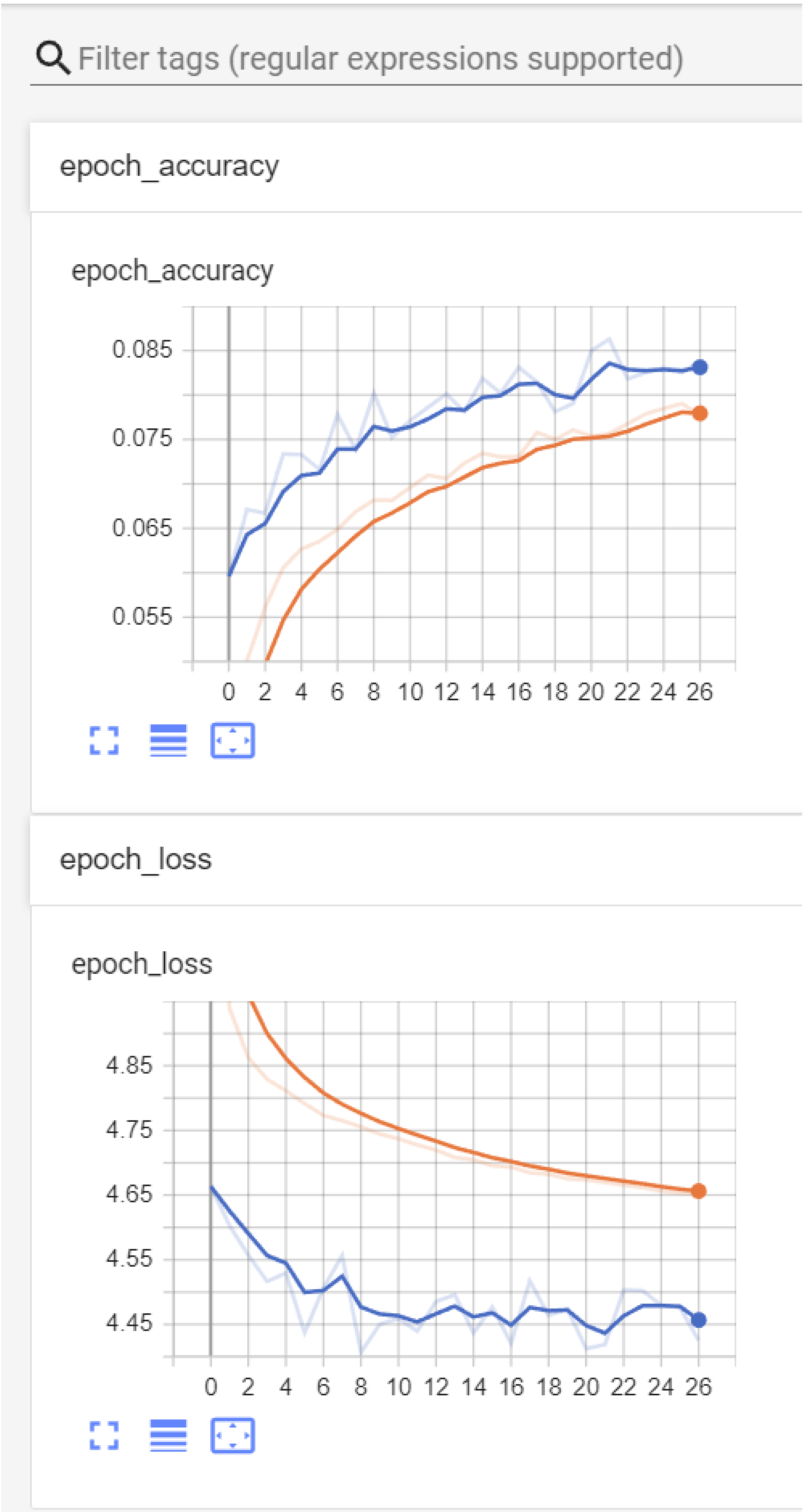
Above we can see an image of some of the classifications that were made. To help me with the classification I used various different approaches to best solve my problem.

First I began with transfer learning involving the VGG16 model trained with ImageNet weights and changing the output layer with the custom CNN architecture with adam optimizer. The model was performing very poorly even after 24 epochs, this resulted in an accuracy of 8%.

After that, I realized that the SGD optimizer is better with the VGG16 model, but this also failed as the results were just as similar as that with the Adam optimizer.

Then I tried to train all the layers of VGG and added some additional layers on top, this finally helped me achieve somewhat better results of 25% after 60 epochs.

In the end, I ended up using the inceptionv3 pre-trained model, and this resulted in an accuracy of 92%. This had a penalty of 0.05 in the FC layer, SGD optimizer with a learning rate of 0.0001 and 0.9 momentum.

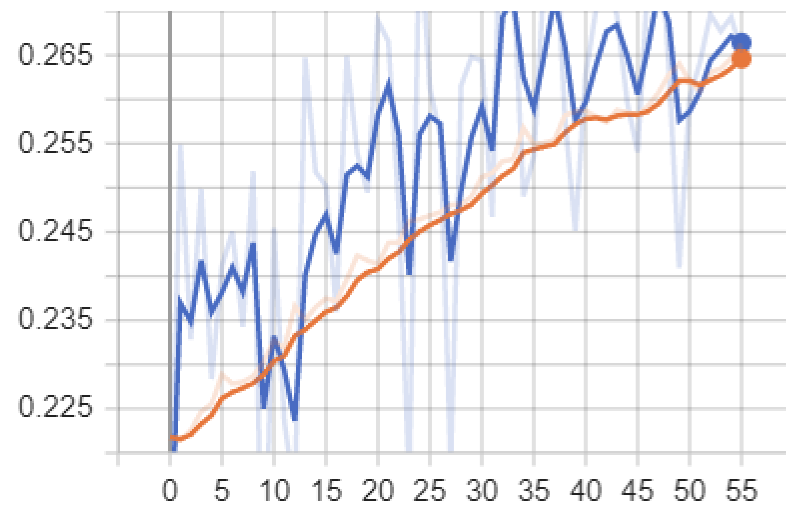


Filter tags (regular expressions supported)
epoch_accuracy
epoch_accuracy
0.085
0.075
0.065
0.055
0 2 4 6 8 10 12 14 16 18 20 22 24 26
epoch_loss
epoch_loss
4.85
4.75
4.65
4.55
4.45
0 2 4 6 8 10 12 14 16 18 20 22 24 26

Filter tags (regular expressions supported)

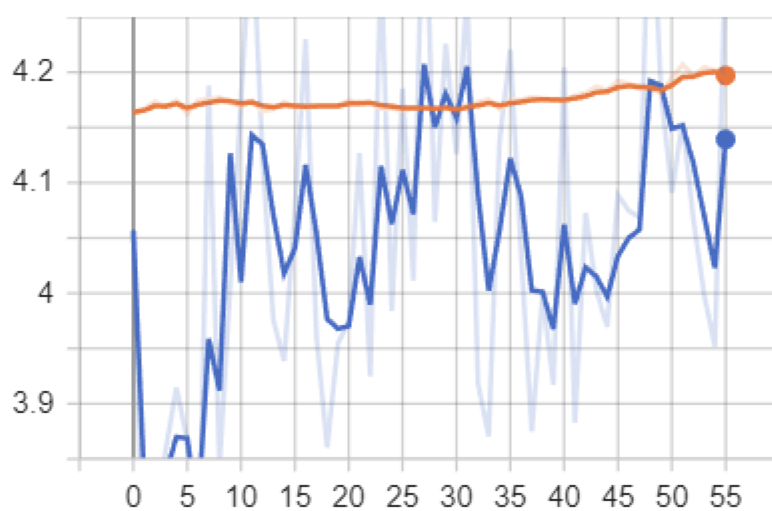
epoch_accuracy

epoch_accuracy



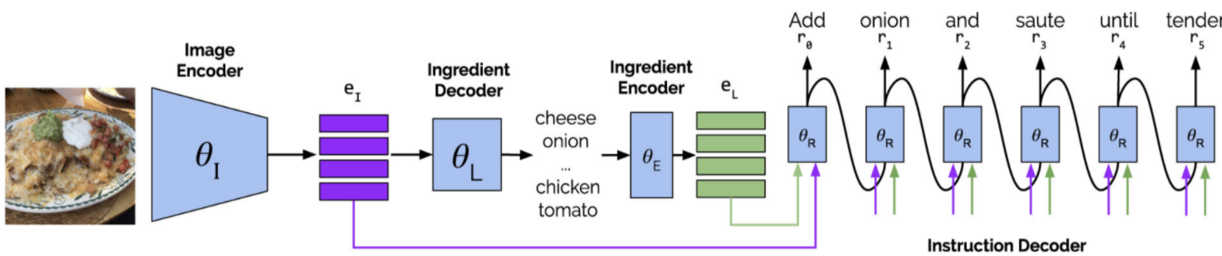
epoch_loss

epoch_loss



Recipe Generation

For the recipe generation model, was also a bit complex, requiring three different levels of training to achieve an acceptable solution. For this problem, I would have to start by understanding the food from the image and then extracting the ingredients. I would then have to classify the food title and finally use natural language processing (NLP) to prepare a recipe to provide to the user.



Architecture for Food recipe generation

I would be using Facebook's inverse cooking project and dataset to help me with solving my problem statement. In this, they used multiple res-net encoders to identify the features and obtain the ingredients. From the ingredients, they attempted to do a multi-class classification with the recipes. After getting the recipes, they are using NLP to get a proper recipe that the user can follow to make the dish.

The dataset contained over 100 GB of data. To process the data in a more efficient manner, I used paper space to rent out a much more powerful machine.

Screenshots below from Paperspace machine

Stage 1

paperspace@psnqdn0lf: ~/SeeFood-Project/ReceipeGen/inversecooking/src

```
  File "/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/python3.7/site-packages/nltk/data.py", line 752, in load
    opened_resource = _open(resource_url)
  File "/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/python3.7/site-packages/nltk/data.py", line 877, in _open
    return find(path_, path + [""]).open()
  File "/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/python3.7/site-packages/nltk/data.py", line 585, in find
    raise LookupError(resource_not_found)
LookupError:
**********************************************************************
  Resource punkt not found.
  Please use the NLTK Downloader to obtain the resource:

  >>> import nltk
  >>> nltk.download('punkt')

  For more information see: https://www.nltk.org/data.html

  Attempted to load tokenizers/punkt/PY3/english.pickle

  Searched in:
    - '/home/paperspace/nltk_data'
    - '/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/nltk_data'
    - '/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/share/nltk_data'
    - '/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/nltk_data'
    - '/usr/share/nltk_data'
    - '/usr/local/share/nltk_data'
    - '/usr/lib/nltk_data'
    - '/usr/local/lib/nltk_data'
    - ''
**********************************************************************

(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ vi build_vocab.py
(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ (venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ vi build_voca
b.py
(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ python build_vocab.py       (venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inver
(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ python build_vocab.py --recipe1m_path /home/paperspace/SeeFood-Project/ReceipeGen/mydataset
[nltk_data] Downloading package punkt to /home/paperspace/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
Loading data...
Loaded data.
Found 1029720 recipes in the dataset.
68589it [01:19, 869.79it/s]
```

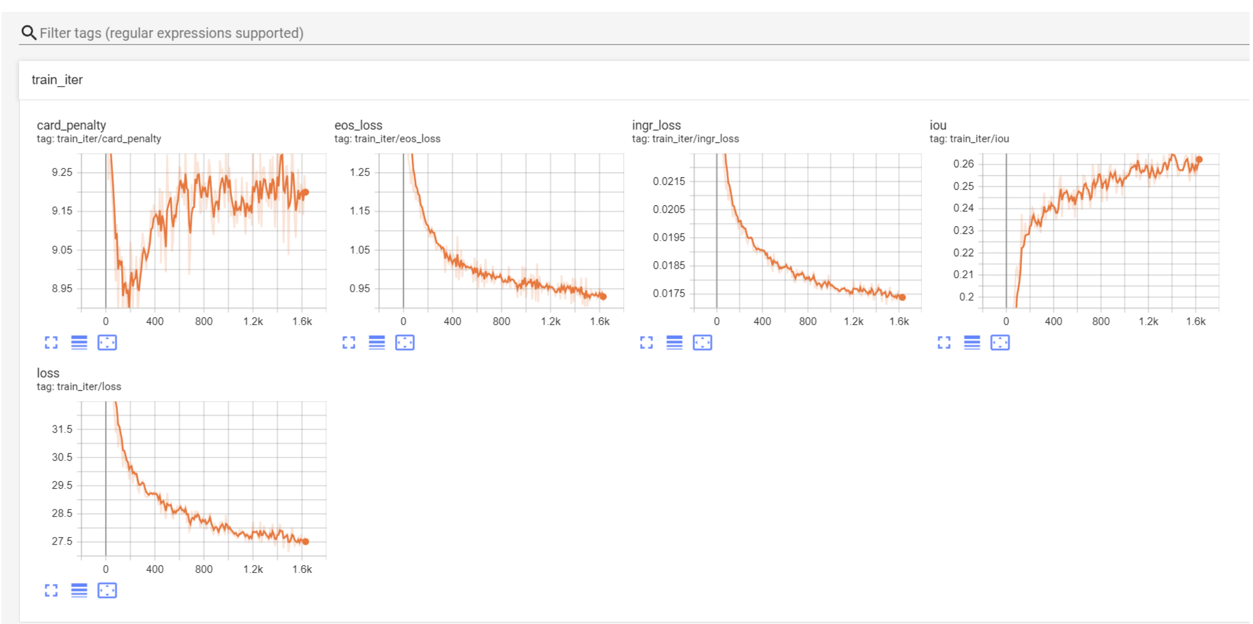
Stage 2

paperspace@psnqdn0lf: ~/SeeFood-Project/ReceipeGen/inversecooking/src

```
  File "/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/python3.7/site-packages/nltk/data.py", line 752, in load
    opened_resource = _open(resource_url)
  File "/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/python3.7/site-packages/nltk/data.py", line 877, in _open
    return find(path_, path + [""]).open()
  File "/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/python3.7/site-packages/nltk/data.py", line 585, in find
    raise LookupError(resource_not_found)
LookupError:
**********************************************************************
  Resource punkt not found.
  Please use the NLTK Downloader to obtain the resource:

  >>> import nltk
  >>> nltk.download('punkt')

  For more information see: https://www.nltk.org/data.html

  Attempted to load tokenizers/punkt/PY3/english.pickle

  Searched in:
    - '/home/paperspace/nltk_data'
    - '/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/nltk_data'
    - '/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/share/nltk_data'
    - '/home/paperspace/SeeFood-Project/ReceipeGen/inversecooking/venv-receipe/lib/nltk_data'
    - '/usr/share/nltk_data'
    - '/usr/local/share/nltk_data'
    - '/usr/lib/nltk_data'
    - '/usr/local/lib/nltk_data'
    - ''
**********************************************************************

(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ vi build_vocab.py
(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ (venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ vi build_voca
b.py
(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ python build_vocab.py       (venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inver
(venv-receipe) (base) paperspace@psnqdn0lf:~/SeeFood-Project/ReceipeGen/inversecooking/src$ python build_vocab.py --recipe1m_path /home/paperspace/SeeFood-Project/ReceipeGen/mydataset
[nltk_data] Downloading package punkt to /home/paperspace/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
Loading data...
Loaded data.
Found 1029720 recipes in the dataset.
236937it [04:33, 922.74it/s]
```

Stage 3

Tensorboard



The following chart shows the performance of the recipe generation. Each epoch took approximately 16 minutes and a total of 400 epochs were evaluated.

Stages	IoU	F Score
Ingredient Prediction	32.11	48.61
Recipe Generation	32.52	49.08

Calorie Estimation

For the calorie estimation I decided to use the results from the recipe generation. I started by looking at the data provided in my dataset. I would use the information about each recipe to generate the number of calories.

I was provided with the fat/saturation, protein, sodium, and sugar amounts for every 100 grams serving size. Using this I calculated the amount of energy for the serving size

Type of energy	Energy consumed for each gram	Calorie consumed for each gram
Fat/Saturation	37 Kilojoules	8.88 Calories
Protein	17 Kilojoules	4.08 Calories
Sodium	8 Kilojoules	1.92 Calories
Sugar	17 Kilojoules	4.08 Calories

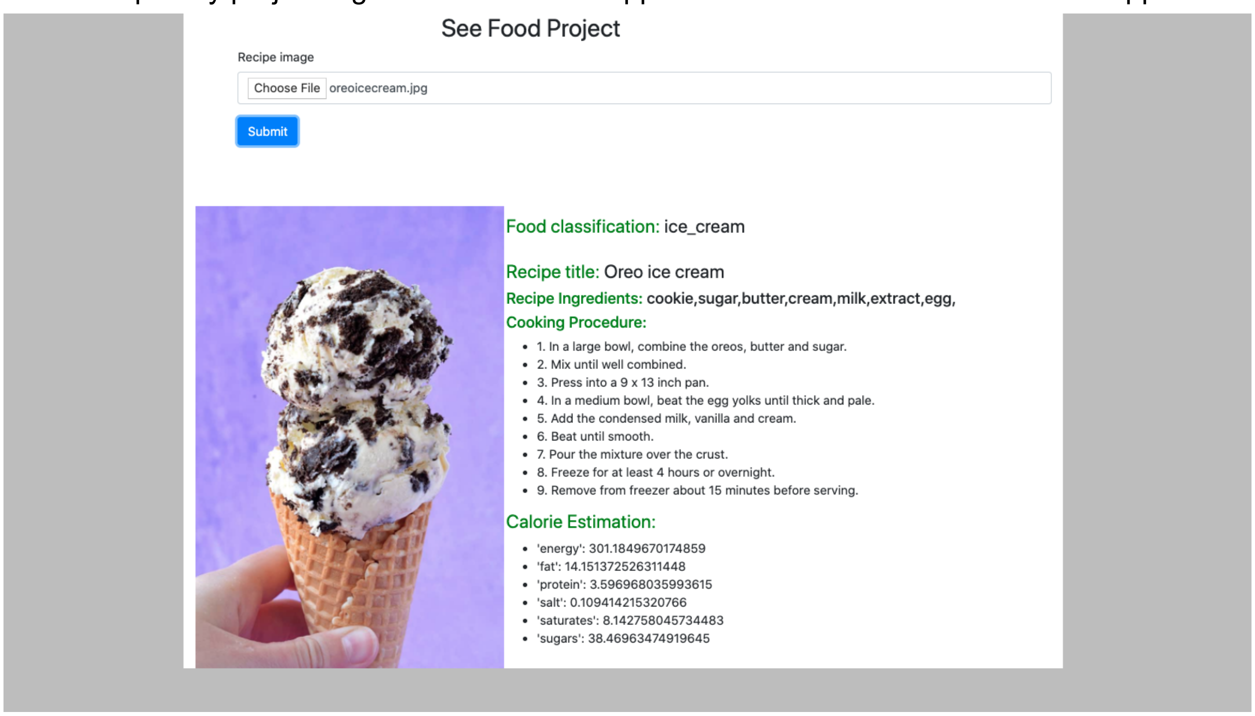
My final equation to calculate the number of calories would be 0.24((fat * 37) + (protein*17) + (sodium *8) + (sugar * 17)) = total calories. Since the number of calories for each kilojoule is approximately 0.24.

Below is a screenshot of the JSON file from which I collected nutritional information


```
[
    {
        "fsa_lights_per100g":{
            "fat":"green",
            "salt":"green",
            "saturates":"green",
            "sugars":"orange"
        },
        "id":"000095fc1d",
        "ingredients":[
            {
                "text":"yogurt, greek, plain, nonfat"
            },
            {
                "text":"strawberries, raw"
            },
            {
                "text":"cereals ready-to-eat, granola, homemade"
            }
        ],
        "instructions":[
            {
                "text":"Layer all ingredients in a serving dish."
            }
        ],
        "nutr_per_ingredient":[
            {
                "fat":0.8845044000000001,
                "nrg":133.80964,
                "pro":23.110512399999998,
                "sat":0.26535132,
                "sod":81.64656,
                "sug":7.348190400000001
            },
            {
                "fat":0.46,
                "nrg":49.0,
                "pro":1.02,
                "sat":0.023,
                "sod":2.0,
                "sug":7.43
            },
            {
                "fat":7.415,
                "nrg":149.25,
                "pro":4.17,
                "sat":1.207,
                "sod":8.0,
                "sug":6.04
            }
        ],
        "nutr_values_per100g":{
```


Web-App

To put my project together I built a web app. Below is a screenshot of the web app



Here I loaded the image oreoicecream.jpg. The image was then classified as ice cream, given the title Oreo Ice Cream, and listed out the ingredients used to create it (cookie, sugar, butter, cream, milk, extract, egg). Then I was given the instructions to make the item myself, and finally, I was given the nutritional information about the item.

Summary

With the growth of social media, I am able to see a more in-depth picture of different travel spots across the globe. A lot of these pictures are spread from Instagram, and other image-sharing sites. Many of these images are of the different delicacies in their respective part of the globe. The more I look at these images, the more I want to go be able to touch and eat these delicacies. My goal for this project was for you to be able to feed the image into my platform, and I would provide you with classification information, recipe information, and the calorie estimation based on the image.

With many other use cases for my model, I know that this is something that people will continue to use for many years to come. This is something that I can say will provide the greatest amount of flexibility to people when they travel. Also to get recipes from when they are going to a restaurant, and gather information regarding the cooking process, how many calories are in a food item, and how to properly make something they really like.

Each model will be validated for cost function, root mean error, mean estimation error F1 score, and IoU score. Each problem statement has a different architecture and involves NLP techniques for result generation.