# Machine Learning Engineer Nanodegree

## Capstone Proposal

Gautham Venkatesha Reddy
January 24th, 2018

## Domain Background

‘Yelp’ has large troves of data gathered and contributed by many Yelp reviewers. This data must be leveraged to provide the company with competitive advantage in the ever competitive environment in the online domain. Yelp reviewers have written a total of 142 million reviews **[1]** at the end of 2017 and thus, it is humanly impossible to sift through such large collections of images and texts and manually classify the data. Thus, machine learning models can be employed to give exceptional performance. Traditional programming is limited and inflexible to address the company’s current problems.

Correctly classifying images of food will not only help ‘Yelp’ become a dominant player it’s field, it will also indirectly help small businesses improve their quality of service and help prioritize and display relevant images of their signature dishes on their Yelp business page. This will attract more customers and help generate monetary value for these businesses.

Since these datasets have only recently incorporated images in their ‘Yelp dataset challenge’, no prior work has been done where they incorporate CNNs to classify the type of food in their images. Thus no benchmark and base has been set for this particular project. However, there have been attempts to classify food on a different dataset and one of the user has used an Inception Architecture and has achieved an accuracy percentage in the high 80s**[2]**.

## Problem Statement

For machine learning Capstone project, I will attempt to classify if a given image is that of a ‘food’, ‘drink', ‘establishment/environment (indoor or outdoor)’ or ‘menu’. If the image is that of a food, then I will proceed to classify the type of food in the image, such as salads, deserts (which will include cakes and ice creams), burgers, pizzas, fries, steaks, noodles (includes pad thai, lo mein, chowmien), rice and alcoholic drinks (such as beer, wine, rum, vodka etc).

**Note:** The actual labels of the type of food may vary based on new findings in the dataset. However, the overall problem statement will be the same.

## Datasets and Inputs [3]

The dataset is available on the company’s ‘Challenge’ page (https://www.yelp.com/dataset/challenge) as part of their ‘Yelp dataset Challenge’ competition. The first dataset contains six files (business.json, checkin.json, photos.json, review.json, tip.json, user.json) in JSON format (SQL is also available, I will be working with JSON format) and the second dataset contains images. I will mostly be working with ‘photos.json’ file which contains relevant information about image dataset (such as labels).

**Example of ‘photos.json’ file:**

```
{"photo_id":"VZXDC7VBdIXXjE3omVqeMg","business_id":"JzB7NITHQ7gVHGVZ1ntgIQ","
caption":"Black Angus Steak Sandwich... Huge!!!","label":"food"}
{"photo_id":"c6Em6dDZ4aVKDI8Lc2BQog","business_id":"JzB7NITHQ7gVHGVZ1ntgIQ","
caption":"","label":"outside"}
{“photo_id":"VAoFn_z9QF0qVmT5vTdwWA","business_id":"JzB7NITHQ7gVHGVZ1ntgIQ","
caption":"","label":"food"}
```

The “photo_id” tag is the filename of the photo in the image dataset and “label” tag contains only two labels, ‘food’ and ‘outside’, which I will use to train if the image of the photo is either a food or an environment/place. If it is food, I will proceed to classify the type of food stated in the problem statement. To get labels on the type of food present, I will use the “caption” tag and if a particular food, for example ‘steak’ appears in the caption string, then the steak label is ‘1’. In the above example, in the first row, since both ‘Steak’ and ‘Sandwich’ are present, I will randomly choose one of them if there is such an overlap.

**Data specifications and preprocessing ideas:**

- The images in the dataset do not have a constant dimension, thus I will proceed to reshape all images into a standard dimension.
- This dataset contains nearly 200,000 images with labels such as ‘food’, ‘drink’, ‘menu’, ‘inside’ and ‘outside’, and the frequency of occurrence of these labels are vastly different. Thus I will use an F1 metric as my evaluation metric. Since, Keras does not possess this metric, I will create a custom F1 metric.
- While splitting the data, I need to maintain the same percentage of samples, thus I intend to use stratified K-Fold cross validation split to split by data into training, testing and CV datasets.

**Note:** Please check the references section at the end, for the link to download the dataset.

## Solution Statement

For image classification, I intend to use a self built convolutional neural network architecture as well as transfer learning architectures such as VGG16, VGG19, ResNet and Inception architectures . Due to the variety of the images, CNN would give superior performance. Usage of Data Augmentation will certainly boost the accuracy. All these tasks have a well defined loss function and I intend to use ‘categorical cross entropy’ loss function and F1 as my performance metric.

## Benchmark model

For the CNN network, since this dataset is new and nobody has attempted to set any benchmarks yet. Thus, I hope to achieve results better than random guessing. For the first part of classification, I hope to achieve an accuracy better than 50%, hopefully in the high 80’s. And for the second part of food classification, I hope accomplish the same (With reasonably good accuracy). Training a vanilla CNN with with hidden units (such as 3-4 layers) should provide reasonably good accuracy but the performance can be further increased by creating more complex models. Thus, I hope to find superior performance using transfer learning models such as VGGs, ResNet and Inception architectures. Since these layers are extremely deep, it can capture many different features and thus make great predictions. A benchmark accuracy set in one of the projects stated above has achieved an accuracy greater than 85%, I hope to reach this target or maybe higher on the Yelp dataset.

## Evaluation Metric

For CNN classification, I will use categorical cross entropy loss function as my cost function with custom created F1 metric in Keras as my evaluation metric,

$Cost = -\sum_{j} t_j log(y_j)$, where $t_j$ is the target labels and $y_j$ is predicted label for all ‘j’,

$Precision = TP/(TP + FP), Recall = TP/(TP + FN)$

$F1 = 2 * Precision * Recall/(Precision + Recall)$

## Project Design

The details of the work flow for the project are listed below.

**Load the dataset:** After importing all the required libraries, I will proceed to load and read all the required datasets into the program. Since, the labels and data are well collected in JSON format, it is very easy to extract all the required information and store them in a python list or dictionary.

**Steps:**

- **Divide the dataset:** I will first proceed to divide the dataset into training, CV and testing data. The ‘X’ label will contain the path names of the images and this path names will be loaded during training.
- **Resize images:** The images will be resized to a particular shape to pass as correct number of inputs.
- **Create CNN architecture:** I will proceed to first build my own CNN architecture in a sequential manner with 5-6 hidden layers with ‘relu’ activation and last layer will be a fully connected layer with ‘sigmoid’ activation. Dropouts will be added to prevent overfitting. Max pooling layer will also be added to capture the most import feature in the given window as well as for dimensionality reduction. I will then proceed to use transfer learning architectures such as VGG, ResNet and Inception models to achieve better performance. The best architecture will be selected based on the training time required, accuracy and response time for each detection. All these parameters will be evaluated and the reason for their choice will be provided by me based on the fastness and accuracy of the application.

## References

1. About Yelp, https://www.yelp.com/about
2. https://github.com/stratospark/food-101-keras#Loading-and-Preprocessing-Dataset
3. Yelp Dataset information and download link: https://www.yelp.com/dataset/challenge
4. Other previous work on Yelp dataset: https://www.yelp.com/dataset/challenge/winners