Yelp Restaurant Photo Classification

Rajarshi Roy
Stanford University
rroy@stanford.edu

Abstract

The Yelp Restaurant Photo Classification challenge is a Kaggle challenge that focuses on the problem predicting user labels of restaurants based on user review photographs. This project approached the problem with the CaffeNet convolutional neural network architecture with transfer learning from a trained AlexNet and a custom ensemble approach to achieve a score of 0.7797, which is close to the highest achieved score of 0.8278.

1. Introduction

The goal of the Yelp restaurant photo classification challenge [1] is to build a model that automatically tags restaurants with multiple labels using a dataset of user-submitted photos. Currently, Yelp users manually select restaurant labels when they submit a review. Selecting the labels is optional, leaving some restaurants un- or only partially-categorized. These labels are:

0: good_for_lunch
1: good_for_dinner
2: takes_reservations
3: outdoor_seating
4: restaurant_is_expensive
5: has_alcohol
6: has_table_service
7: ambience_is_classy
8: good_for_kids

The training dataset provides the labels for 2000 restaurants, 234840 photos and a mapping between the photos to the restaurant. The test dataset consists of 237152 photos and a mapping to 10000 restaurants. The trained model must tag each of these 10000 restaurants with one or more of the 9 labels that apply.

This is an interesting problem since the labeling must be predicted on restaurants but the number of photos for each restaurant varies. So the input size is for the model is not constant. Furthermore, popular image classification challenges like ImageNet are single-output, multi-class problems where an image has to be tagged with one label, which can fall in multiple categories. This problem in this challenge is a multi-output, binary classification problem where the restaurants have to be tagged with 9 labels, where each label can fall into 2 categories: applies or does_not_apply.

The input to the trained model in this project will be the set of user-review photos for a restaurant in jpeg format. The number of photos and size of photos are variable. The output from the model will be a list of predicted labels for the restaurant. For example, the output of the model may be (1,2,3,8) which predicts that labels 1,2,3 and 8 apply to the restaurant while labels 0,4,5,6 and 7 do not apply to the restaurant.

The first stage of the model explored in this project will predict label scores: [applies, does_not_apply] for each of the photos for a restaurant. The photo label predictor is convolutional neural network with a fixed input size of one image. After the label scores for all photos of a restaurant are predicted, the model will combine the scores to predict the overall label scores for restaurant. Some options of this combination logic are explored in this project. Finally, the restaurant is tagged with labels that have higher “applies” score than “does_not_apply” score.

The constraints of this project were time and computation. The time frame to execute the bulk of this project was three weeks. The execution involved setting up frameworks, preparing data, training models and predicting outputs. The computation resource used for this project was a personal desktop equipped with a NVIDIA GTX970 graphics card (Maxwell GM204 architecture and 4GB video memory).

2. Related work

The core problem of this challenge is essentially an image classification problem photos of restaurants should indicate their labels. An often-cited benchmark of image classification technique performance is the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) [2]. Currently, the organizers provide a labeled dataset with 1000 categories for image classification. The training set contains over 1.2 million images of the average image resolution of 256×256 pixels. There is also a dataset of 150,000 labeled images for validation and test.

The introduction convolutional neural networks to tackle the ImageNet challenge in 2012 with the AlexNet architecture [3] greatly improved accuracy over existing methods and achieving a top-5 error of 15.3%. This work indicates that convolutional neural networks are very well suited to the image classification problem and influenced the decision in this project to use convolutional neural networks as the photo label classifier. The AlexNet architecture also demonstrates the usefulness of two techniques: dropout [3] and model ensembles.

Further advances in the ImageNet challenge was made in the following years with the ZFNet[4] architecture that achieved top-5 error rates of 14.8% in 2013, the VGGNet[5] architecture with top-5 error of 7.3% in 2014, the GoogLeNet[6] architecture with top-5 error of 6.7% in 2014 and the ResNet[7] architecture with top-5 error of 3.6% in 2015. These architectures indicate a trend of better performances with deeper layers in VGGNet, more complex convolution ensemble layers in GoogLeNet and gradient bypasses with residual layers in ResNet.

Due to the time constraints of this project however, the idea of transfer learning from a relatively simpler architecture like AlexNet was an attractive strategy. The concept of transfer learning was explored in a study that used the overfeat network that was trained for the ImageNet challenge in 2013 to other similar image classification tasks and obtained state-of-the-art level accuracy [8]. Another study concluded that transfer learning from part of a pre-trained network and even fine-tuning the weights could attain very good performance [9]. There are two scenarios that can be explored with transfer learning. Firstly, layers other than the end layers of a pre-trained network can be used with a different end layer that is custom to the new problem. The learning rate of the pre-trained layers can be set to 0. As a result the pre-trained portion will be used as a fixed feature extractor while the custom final layers can learn to predict scores for the new problem from the fixed features. Secondly, the pre-trained layers could have a low learning rate set. This way, the pre-trained portion gets initial pre-learned weights but fine-tunes to new features with training.

The CaffeNet [10] architecture is an open-source transfer-learning example that transfer-learns from the AlexNet model trained for the ImageNet challenge to predict scores for the FlickrStyle dataset. The ImageNet dataset has 1000 output classes whereas the FlickrStyle dataset has 20 output classes. Thus, all the last 1000 output fully connected layer of the AlexNet architecture is replaced with a 20 output fully connected layer. This layer is initialized with Gaussian weight initialization and all other layers are initialized with AlexNet weights for ImageNet.

3. Methods

Convolutional neural networks architectures for image classification have a fixed input size (typically one image). For this challenge we have to classify restaurant labels with a variable number of images. One possible approach to this problem would be to use recurrent neural networks [11], which operate on a sequence of inputs with variable lengths. However, recurrent neural networks are suited to data, which inherently have some sequential correlation such as frames of a video of words in a sentence. In this case, we are presented with a shuffled set of user-review images for a restaurant that have no sequential information.

Another possibility is to take create a network which has n input images where n is the smallest number of photos associated with a restaurant in the test and training datasets. In this case n is 8. Then restaurants with a larger set of photos will have to have their images subsampled many times. The downside to this approach is that restaurants with a large set of photos (26) may not fit the model well and also training and testing for these restaurants will take a longer time.

The approach taken in this project does not attempt to directly build a restaurant label classifier. Rather the approach is to make a photo label classifier that will predict label scores for photos. Then these scores can be combined in various ways to predict labels for a restaurant associated with those photos.

3.1. Data Processing

The first phase in a training photo label classifier is to have a training data set with photos and associated training labels. The training data provided by the challenge is a map of photos to restaurants and labels for each restaurant. The approach taken was to tag all photos of a restaurant with the labels of the restaurant. This creates a dataset with photos directly tagged with labels. The advantage of this approach is that well proven convolutional neural networks for image classification can now be used on this dataset. There are certain disadvantages with this approach. Firstly the information of how certain photos are grouped together for the same restaurant is lost. That information may have been valuable for prediction. Secondly not all photos of a restaurant may valuable for all labels. As a result, accuracy for just photo label classification may not be as high as restaurant label classification.

The second phase of data processing is to prepare the data for the network architecture. The architecture used for training is the CaffeNet architecture [10] that was mentioned briefly and will be discussed in further detail in the following section. This architecture is transfer learned from the AlexNet architecture for the ImageNet challenge. The AlexNet data augmentation approaches were to randomly crop 227x227 pixel square patches from the 256x256 pixel input image, randomly mirror images and subtract RGB channel averages. The input images provided for this challenge varies in aspect ratio and are sized to make the longest side 517 pixels. The first process was to resize the images with fixed aspect ratio to make the shorter side 256 pixels wide. Then the same data augmentations of 227x227 crops, random mirroring and

RGB channel average subtraction are applied using Caffe's built-in data layer options (training) and PyCaffe's predict function options (testing). Note that the channel averages used were not from the training data but rather the AlexNet channel averages. Also, a large number for training images were square due to Instagram's popularity so resizing those images to shorter side 256 pixels made them 256x256 pixel images like ImageNet images. The similarity in image inputs between ImageNet and the Yelp dataset is conducive to transfer learning.

Finally, for validation and ensemble purposes, the dataset was divided into 4 parts. Then all four combinations of 3/4th training+1/4th validation datasets were created and processed with labels to lmdb files using the convert_imageset script provided in the Caffe toolset.

3.2. Photo Label Classifier

The photo label classifier should be a model that outputs 9 labels (good_for_lunch, good_for_dinner, takes_reservations, outdoor_seating, restaurant_is_expensive, has_alcohol, has_table_service, ambience_is_classy, good_for_kids) x 2 classes (applies, does_not_apply) =18 scores given an input image.

Due to the proven capabilities of the CaffeNet model for the FlickrStyle dataset, a modified version of it was used for this project. The CaffeNet has the same layers as AlexNet except for the fc8 layer since the CaffeNet is designed to output for 20 classes for FlickrStyle as opposed to 1000 classes for ImageNet. Similarly, the output layer of the CaffeNet was modified for this project to suit the multi-output binary classification nature of this problem.

One possibility for this problem would be to use a single net for all 9 labels. Then the fc8 layer would output scores for all 9 labels and a multi-output loss function such as in this study [12] can be minimized used to train the network.

The approach taken in this project was to have 9 separate binary output nets and a 2-class loss function for each net to minimize for training. This disadvantage of this approach is that 9 separate models have to be trained and 9 separate models have to be used for prediction. With the 4-fold ensemble scheme that was tried in this project, 36 models had to be trained. The advantage of this approach is that the Caffe provided loss functions could be used and thus save debug effort if the multi-output loss function is implemented wrongly.

Another modification to the CaffeNet model was to replace the ImageData input layer of CaffeNet with Data input layer with lmdb backend to speed up training.

3.3. Restaurant Label Classifier

After the photo label scores are calculated, the restaurant labels have to be predicted from the photo label scores. One possible option is to average the "applies" score and "does_not_apply" score for each label independently across all the photos for a restaurant. Then the higher score will determine if the label applies to the restaurant or not. Results from this approach and some experimental improvements will be discussed in greater detail in the results and discussion section.

4. Discussion

To measure the accuracy of the photo label classifier, four-fold cross validation was performed. Each fold consisted of 176130 training images and 58710 validation images. A separate network was trained for each fold and label combination. The accuracy metric for each label was the fraction of images that were correctly tagged with the label. Table 1 summarizes the final validation accuracy of each net.

Label:	Fold1	Fold2	Fold3	Fold4	Average
good_for_lunch	0.797	0.808	0.796	0.799	0.800
good_for_dinner	0.746	0.737	0.745	0.751	0.745
takes_reservations	0.777	0.775	0.782	0.769	0.776
outdoor_seating	0.569	0.522	0.563	0.554	0.552
is_expensive	0.737	0.745	0.743	0.735	0.740
has_alcohol	0.803	0.795	0.791	0.795	0.796
has_table_service	0.826	0.832	0.824	0.825	0.827
ambience_is_classy	0.721	0.738	0.727	0.724	0.728
good_for_kids	0.748	0.733	0.753	0.742	0.744

Table 1: Photo Label Classifier Accuracy

The highest validation accuracy of 82.7% was achieved for the has_table_service label. The lowest validation accuracy of 55.2% was attained for the outdoor_seating label. A random sampling of 10 images from the training dataset in Figure 1 shows that most photos in the dataset are photos of restaurant interiors. Thus, it is understandable if there is low correlation between the photos and the outdoor_seating label.

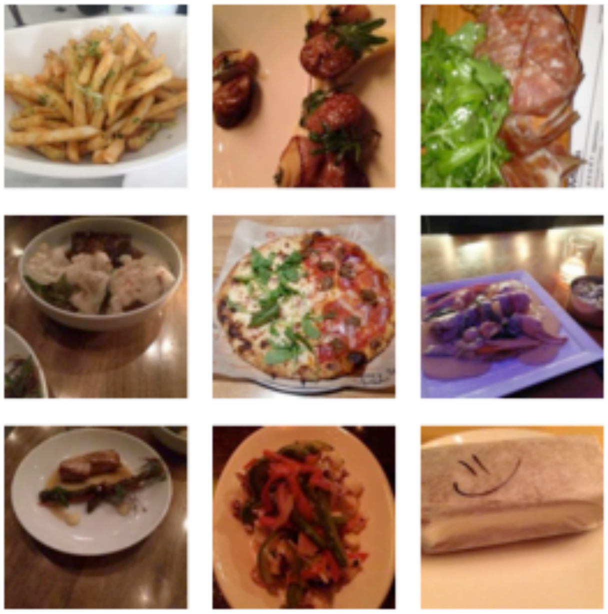

Figure 1: Random image samples of training dataset.

Another way to analyze if the photo label classifiers are trained correctly is to show the highest scoring images of each label.

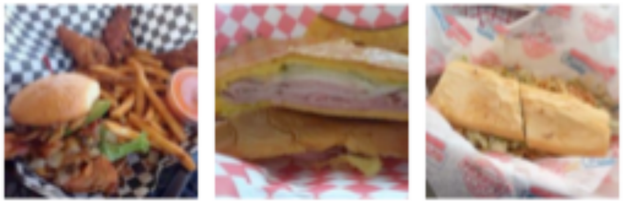

Figure 2: Top-3 good_for_lunch photos

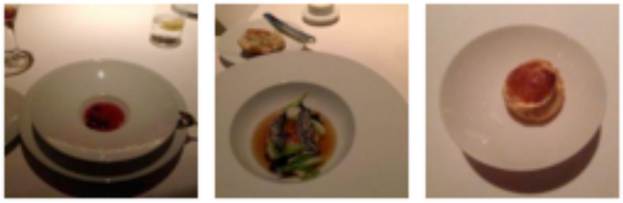

Figure 3: Top-3 good_for_dinner photos

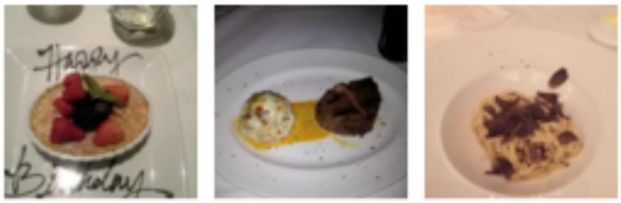

Figure 4: Top-3 takes_reservations photos

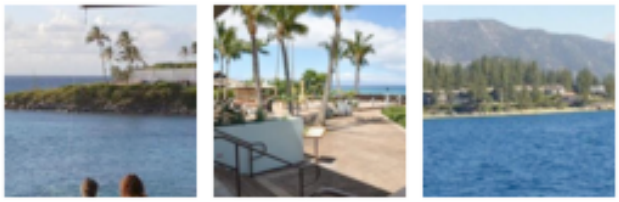

Figure 5: Top-3 outdoor_seating photos

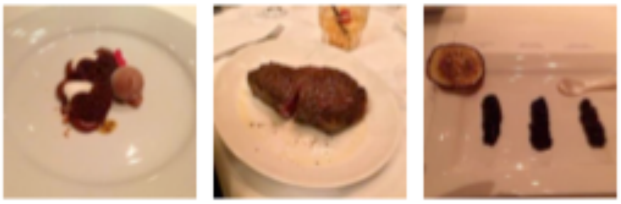

Figure 6: Top-3 restaurant_is_expensive photos

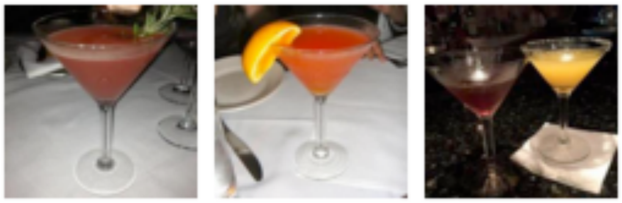

Figure 7: Top-3 has_alcohol photos

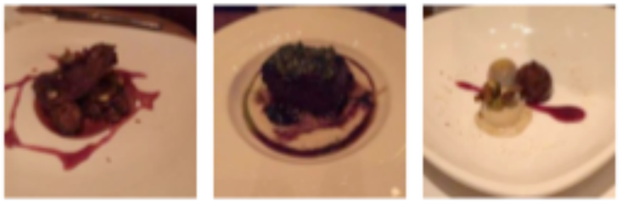

Figure 8: Top-3 has_table_service photos

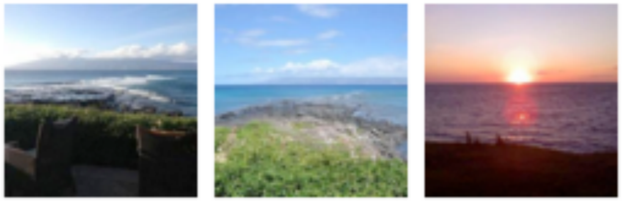

Figure 9: Top-3 ambience_is_classy photos

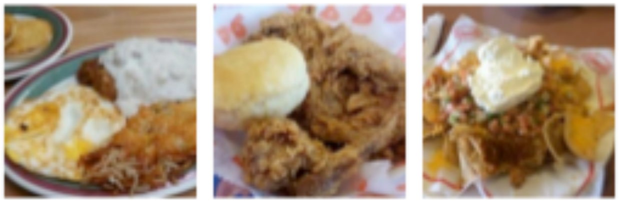

Figure 10: Top-3 good_for_kids photos

The highest scoring images reveal qualitatively that all the classifiers were trained correctly. The expected results were that good_for_lunch photos show lunch food like burgers, sandwiches and burritos and has_alcohol photos show cocktail glasses. The interesting results are for has_outdoor_seating and ambience_is_classy photos. The highest scoring has_outdoor_seating photos are of tropical island like settings possibly from beach restaurants. And the highest ambience_is_classy photos reveal waterfront views from restaurant rather than posh interiors.

To test the restaurant label classification, predictions were performed on the test set and the challenge server was used to score the predictions.

The evaluation metric for this competition is Mean F1-Score also known as example-based F-measure in the multi-label learning literature. The F1 score, commonly used in information retrieval, measures accuracy using the statistics precision p and recall r. Precision is the ratio of true positives (tp) to all predicted positives (tp + fp). Recall is the ratio of true positives to all actual positives (tp + fn). The F1 score is given by:

$$F1 = 2\frac{p \cdot r}{p + r} \quad \text{where} \quad p = \frac{tp}{tp + fp}, \quad r = \frac{tp}{tp + fn}$$

The F1 metric weights recall and precision equally, and a good retrieval algorithm will maximize both precision and recall simultaneously. Thus, moderately good performance on both will be favored over extremely good performance on one and poor performance on the other.

From the photo label classifier scores, the first approach that was taken to predict restaurant labels was to the take the mean value of photo label scores. This method yielded a test score of 0.74306.

The analysis of the restaurant predictions revealed that the number of restaurants predicted to have certain labels like good_for_lunch and has_outdoor_seating were exceeding low. Less than 1% of test restaurants were tagged with good_for_lunch whereas 30% of training restaurants were tagged with good_for_lunch. A closer analysis of some restaurants not tagged good_for_lunch that had photos with high good_for_lunch scores revealed that even though some photos showed that the restaurant was good_for_lunch, other unrelated photos such as pets, logos, dark images caused the mean score of "applies" to be lower than "does_not_apply". The next heuristic base don this intuition was to take the max score for "applies" and the continue taking the mean score for "does not apply". This method yielded a test score of 0.76758.

Further analysis showed that the partial max method was predicting 62% of the test restaurants to be good_for_lunch. Even though the statistics of the training set may not indicate anything about the statistics of the test set, the max scoring was modified to be a weight sum of max and mean scores to match the statistics of the training set. This further improved the test score to 0.77328.

This same method weighted sum of max and mean scores for "applies" and just mean scores for "does_not apply" to match training statistics was implemented for all labels. The final score obtained from this method was **0.7797.**

5. Conclusions

With transfer learning from pre-trained AlexNet to predict photo label scores and then an ensemble heuristic to predict restaurant labels from photo label scores, a final F1 score of 0.7797 was achieved for the challenge.

A lot of methods can be explored to try and improve this score. Firstly deeper net architectures and Leaky ReLu activations can be explored. Also, newer net architectures like ResNets can be tried.

Finally, the restaurant label ensemble weights can be learned from the training dataset such that:

label_applies_restaurant =
A1*max_applies + A2*mean_applies +Abias

label_doesnot_apply_restaurant =
B1*max_doesnotapply + B2*mean_doesnotapply + Bbias

References

[1] https://www.kaggle.com/c/yelp-restaurant-photo-classification/

[2] http://www.image-net.org/

[3] http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[4] https://www.cs.nyu.edu/~fergus/papers/zeilerECCV2014.pdf

[5] http://arxiv.org/pdf/1409.1556.pdf

[6] http://arxiv.org/abs/1409.4842

[7] http://arxiv.org/abs/1512.03385

[8] http://arxiv.org/abs/1403.6382

[9] http://arxiv.org/abs/1411.1792

[10] http://caffe.berkeleyvision.org/gathered/examples/finetune_flickr_style.html

[11] RNN

[12] MULTI-OUTPUT LOSS