Machine Learning Engineer Nanodegree

Capstone Project

Benjamin Rouillé d'Orfeuil

May 19, 2018

I. Definition

Project Overview

Yelp is a social networking site that publishes crowd-sourced reviews about local businesses. About two years ago, Yelp challenged Machine Learning practitioners to build a model that automatically tags restaurants with multiple labels using a dataset of user-submitted photographs. The goal of this project is to develop such a model.

The competition was hosted by Kaggle, a platform where data scientists use their skills to produce the best models for predicting and describing datasets uploaded by companies and users. The various datasets and inputs are accessible via the Yelp Restaurant Photo Classification competition webpage[1]. Yelp provides a training dataset (234,842 photographs) and a test dataset (237,152 photographs) for this competition[2]. Each photograph belongs to a business and the task is to predict the business attributes purely from the business photographs. There are 9 different attributes in this problem:

- good for lunch;
- good for dinner;
- takes reservations;
- outdoor seating;
- restaurant is expensive;
- has alcohol;
- has table service;
- ambience is classy;
- good for kids;

and a total of 2,000 (10,000) businesses in the training (test) dataset. Note that this is a multi-instance multi-label classification problem. Each business has multiple photographs and predictions need to be done at the business level. Likewise, multiple labels can be assigned to the same business and hence the classifier needs to account for label dependencies.

Problem Statement

Neural networks have proven to be incredibly efficient at classifying images and often outperform other machine learning algorithms at this task. It comes then as no surprise that deep learning models are used extensively in this project. One now faces two options: i) build and train a deep neural network

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

[2] The datasets are quite large. Both the training and test archive files have a size of about 7 GB.

from scratch or ii) use transfer learning[3]. The properties of the dataset such as its size and nature usually dictate the type of approach to adopt. The Yelp dataset being both large and complex, it would be unrealistic to train a deep neural network model from scratch given this task would require fine expertise and enormous resources. Also, deep neural networks that have been pre-trained on large and diverse dataset like ImageNet[4] capture universal features in its early layers that are relevant and useful for most computer vision problems. Thus, leveraging such features allows to reach a better accuracy than any method that would rely only on the available data. For those reasons, transfer learning is a better approach for this project and a pre-trained deep learning model is used as a fixed feature extractor.

The next step entails the selection of the most relevant pre-trained model for the problem domain. Four state-of-the-art deep learning models whose weights have been pre-trained on the ImageNet database are considered here. For each model, the bottleneck features[5] are extracted and used as inputs of a very simple classifier. Each classifier is independently trained and their performance is then evaluated on an unseen set of features. The model whose bottleneck features lead to the most accurate predictions is chosen and the bottleneck features are computed for the entire training dataset.

Before feeding the bottleneck features to a classifier, one needs to address the multi-instance aspect of this project. There are essentially two options: i) derive a feature vector for each instance and combine them accordingly to get one feature vector per restaurant or ii) assign to each instance the label of its corresponding restaurant, proceed to classification and average the output probabilities for each label. Both scenarios are investigated in this project.

Finally, a classifier is trained and predictions are made. Two models are considered for the classification task: i) a multi-layer neural network with a final layer containing one node for each label and ii) gradient boosted trees. It is worth noting that a neural network automatically accounts for eventual dependencies among labels because it shares weights for the different label learning tasks. For the other model, label dependencies are handled through classifier chains.

Metrics

The harmonic mean between precision (p) and recall (r), the F_1 score, is used to evaluate the performance of the algorithm:

$$F_1 = 2\frac{p \cdot r}{p + r} \text{ where } p = \frac{tp}{tp + fp} \text{ and } r = \frac{tp}{tp + fn}$$

In the above formula, tp, fp and fn denote the true positive, false positive and false negative counts, respectively. In a classification task, $p = 1$ for class i means that every item labeled as belonging to class i does indeed belong to class i whereas $r = 1$ for class i means that every item from class i is labeled as belonging to class i. Though, precision says nothing about the number of items from class i that are mislabeled (fn) and recall says nothing about the number of items that are incorrectly labeled as belonging to class i (fp).

[3]Machine learning technique where a model trained on one task is re-purposed on a second related task.
[4]Large visual database designed for use in visual object recognition software research.
[5]Last activation map before the fully connected layer.

II. Analysis

III. Methodology

IV. Results

V. Conclusion