The directory contents of this distribution are as follow:

Code\_submission.jpynd - A python based jpynd file for building the best model for sentiment classification on the Yelp dataset.

README - This document

Predict\_label.csv - A csv file contains the predicted result of the best model

Train\_data.csv - training data set from Yelp customer reviews with no labels (650000 reviews are included)

Train\_label.csv - labels for reviews from train\_data.csv

Test\_data.csv - testing data with no labels

This is python based code for building a sentiment classification model. Python 3.0 or higher version is required. For building and training the model, tensorflow backend is required.

Building Instructions

Make sure tensorflow backend has been installed in your anaconda application. For installing the tensorflow in anaconda, you can simply use following command in jupyter notebook :

!pip install tensorflow

Running Instructions

All the preprocessing and feature extraction processes can be run cell by cell in local computer. However, the model building and training requires a GPU. If you do not have GPU, it is suggested to implement the model training steps in Google’s colabratory which provides a free GPU. For training the model in colab, you simply need to mount your colab on your Google drive and upload all required input data files generated from preprocessing and feature extraction processes. If you decide to train the model in colab, there will be 6 .txt files from your local computer, including a train\_x, a train\_tfidf, an embedding, a train\_y, a test\_x and a test\_tfidf file. The format of those files will be described in next section.

Input Data Format

The initial input data contains:

1. train\_data.csv - The training data is a Yelp dataset with customer reviews. It contains 650000 reviews and each review has unique id.
2. train\_label.csv - This file contains the labels corresponding to the reviews in train\_data.csv file.
3. test\_data.csv - Unlabeled dataset with 50000 reviews and used for testing the performance of the model.

Input for training:

If you choose to train the model on Google’s colab, there are another 6 input files you need to generate from preprocessing and feature extraction processes.

1. train\_x - which can be read by np.loadtxt( ) into an 650000\*280 format array
2. train\_y - has the same format as train\_label.csv
3. embedding - which is a 100000 \* 1200 matrix and can be read by np.loadtxt( )
4. train\_tfidf - extracted feature for training the model, it is a 650000 \* 280 matrix and can be read by np.loadtxt( )
5. test\_x - the format is similar to train\_x but only have 50000 reviews
6. test\_tfidf - extracted feature and it is required when we make the final predictions on test data.

Output file

1) predicted\_label.csv - It contains 50000 rows and two columns. The first column is test\_id for each predicted label and it is unique. The second column is label predicted by our model. The submission file must satisfy this format.