Yelp Rating Prediction

Rongguang ou (219817313)
Jorge Tadeo (220000847)
Manuel Herrera (219721880)

CSC 180 Intelligent Systems
Mini-Project 1
Friday, September 27, 2019

Problem Statement

Using Yelp's dataset of 6 million reviews of 192,000 businesses, are we able to predict a restaurant's Yelp rating from the text of user reviews alone? The goal is to learn which neural network implementation in Tensorflow would predict a business's star rating based on all the reviews compared to its given averaged star rating.

Methodology

Data collection was provided by Yelp offered as their public dataset for educational purposes with each file as a JSON type.

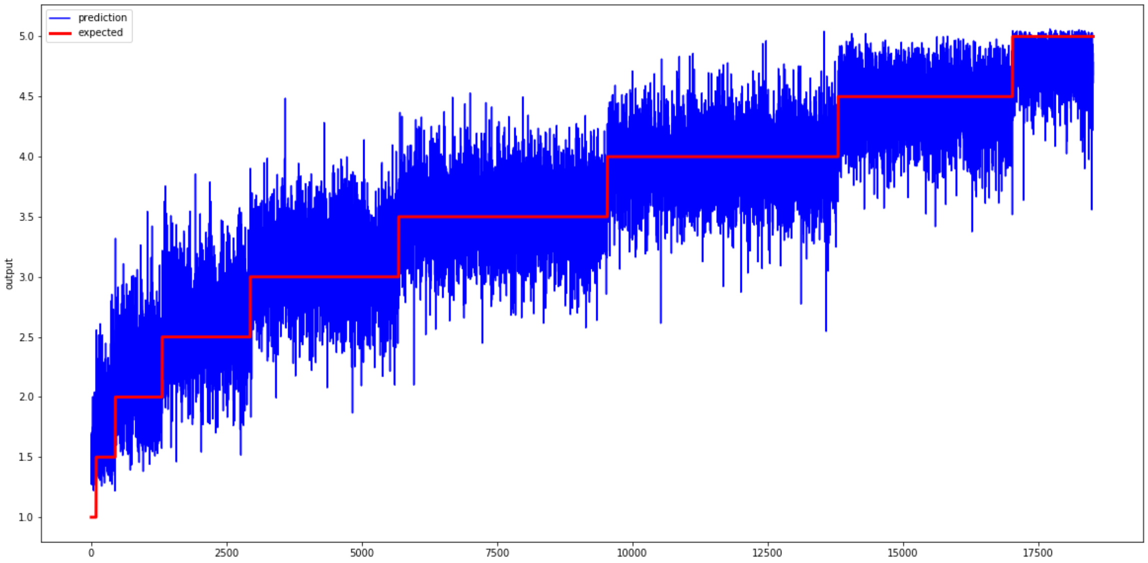
Essential data from both business and review JSON files were extracted to a tab-separated file (tsv) in order to view them as Pandas DataFrames. One containing users' text review, star rating, and business id. The second containing a business's: business id, stars rating, review count, latitude, and longitude. We aggregated the reviews by business id to a numpy array and converted it to another dataframe sorted by business id with its series of reviews. This new dataframe and the business dataframe were merged to form our final dataframe which would be our input table for future calculations.

In addition, we preprocessed our dataframe for proper calculations. We removed business with less than ten reviews associated with them to ensure our data frame contained business with reviews with enough data to predict a star rating. Furthermore, we removed unnecessary columns with no prediction power such as the business id, name, or review count. Missing values in the remaining columns were added with the median value of the column data. Z-scores were generated for the numerical values in longitude and latitude. We transformed our all review columns by vectorizing the natural language data by using scikit-learn to convert the series to a term frequency - inverse document frequency (TF - IDF) matrix in which our rows are text and the columns are words, and values are its values. On our final dataframe, we removed the columns in order to insert our TF-IDF data to a dataframe that would next be split for training and testing..

The neural network is composed of 6 dense layers. The input layer is made up with number of neurons equal to the input feature. We used mean_square_error as error checking function and for gradient descent we used Adam. Model training was repeated for 50 times to increase the chance of obtaining the global maximum. Root mean square function was used to evaluate prediction scores.

Experimental Results and Analysis

We increased predictability by setting the max feature to a higher number graduadually. We observed that the more features we created the lower the RMSE. We found that using a batch size of 32 was optimal for our model. We used sigmoid, relu as the activation function and optimizer adam, sgd to test performance and found both created about the same performance, however relu created slightly better RMSE score. The training for our model was repeated 50 times so that we could avoid local extrema and landing global extrema. As a result, our model generated a RMSE score of 0.3047.



Task Division and Project Reflection

The division of tasks were decided upon skill and understanding of the project. Jason handled fetching and reshaping data into the correct format for each usage, Jorge worked on preprocessing of the data, and Manuel focused model training and parameter tuning. The team worked project design and the written report.

A few challenges we came across where based on both hardware and user. Inputting the aggregated reviews through the tfidf vectorizer function would take a long time to process without shrinking our batch size. We also made the mistake of not inserting the array matrix back into our dataframe to split our train and test function. Hardware also played a role with larger batch sizes needing more RAM. Therefore, our solution was to choose a proper max feature for the tfidf vectorizer that our computer could handle. Overall, we learned about the possibility of building an AI model with the power to predict the star review of a business based on its text-formed reviews.