Wrangle Report

Steve Choi
April 14th, 2019

Introduction

In this project, whole data wrangling process was explored on a tweet archive of Twitter user @dog_rates, also known as WeRateDogs. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog.

The entire project was performed on a Jupyter Notebook using the following python packages, or libraries. You can install these packages via conda or pip.

- pandas
- numPy
- matplotlib
- seaborn
- requests
- tweepy
- json
- datetime

Gathering Data

I gathered data from three different sources. To start with, WeRateDogs downloaded their Twitter archive and sent it to Udacity via email exclusively for you to use in this project. This archive contains basic tweet data (tweet ID, timestamp, text, etc.) for all 5000+ of their tweets as they stood on August 1, 2017.

1. **Twitter-archive-enhanced.csv**:
   With the tweet archive provided by WeRateDogs, Udacity "enhanced" Twitter archive and provided a link so that students can download it manually.

2. **image_predictions.tsv**:
   Udacity also provided students with a tsv file which they ran every image in the WeRateDogs Twitter archive through a neural network that can classify breeds of dogs.
3. **tweet_json.txt**:

Tweepy was used to query Twitter's API for additional data beyond the data included in the WeRateDogs Twitter archive. This additional data includes retweet count and favorite count. Each tweet's entire JSON data was written to .txt file on its own line.

Assessing Data

Once the gathering is done, each 3 DataFrame was assessed both visually and programmatically. Through the assessment process, few numbers of data quality and tidiness issues were reported as below:

Quality Issues

- Remove retweets and replies (keep only the original tweets)
- Missing image URLs in `expanded_urls` column
- Columns with mostly null values
- `timestamp` values in the wrong format
- Errors in both `rating_numerator` and `rating_denominator` columns
- Values in `rating_denominator` not equal to 10
- Values in `rating_numerator` that are less than 10
- Values in `rating_numerator` that are off the chart
- HTML tags in `source` column
- Duplicated URLs in `expanded_urls` for URLs with more than 2 images
- Dogs with more than 1 dogs stages
- Duplicated URLs in `jpg_url` (These might come from retweets & replies)

Tidiness Issuses

- `expanded_urls` from archive data and `jpg_url` from image prediction overlaps
- Collapse dog stages into one column
- Merge DataFrames into one master DataFrame

There are more issues on top of what I have reported. eg) Upper and lower cases in dog breed predictions, tidiness issues with dog prediction table, and etc. However, due to the purpose of this project, I will be moving onto data cleaning.

Cleaning Data

All the cleaning activities were performed in 3 steps, Define, Code, and Test.
In Define, reported quality or tidiness issues in assessment were defined with more details.
In Code, the actual code was written to resolve the issue with appropriate methodologies.
In Test, the result of codes was checked if the issues were completely fixed. If not, I iterated the same process starting from Code until it worked.