Report: **Data Wrangling Report for WeRateDogs Twitter Data**

By: **Quang Luong**

Date: **12/10/2018**

Introduction:

The purpose of this document is to briefly describe the data wrangling process.

Gathering Data:

1. The first file needed to retrieve was the Twitter archive file (twitter-archive-enhanced.csv) that was already made available to us by Udacity.
2. The second file, the image-predictions.tsv, was made available on the Udacity server and was downloaded programmatically using the Requests library.
3. The third file to be used as input was the Twitter json file, which contains many columns. But for the interest of this project only the tweet id, retweet count and the favorite count was used. This file was to be accessed through the Twitter API using the Tweepy library.
    a. Since I had reservations about setting up a Twitter account to connect to the API to retrieve the file, I again used the json file supplied by Udacity.
    b. Once this file was read in, I converted it to a Panda DataFrame.

Assessing Data:

1. I first performed a visual assessment of all three data sets and noted
    a. The datatypes

b. Any missing values – dog phases had 'None' as the value.
c. The dog phases in the Archive table could be pivoted into one column.
d. Checked the rating score within the Text column (Archive) and saw that it does not always line up with that of the variables numerator & denominator ratings.
e. Further visual assessment was performed by using external tools such as Microsoft Excel.

2. I also performed the assessment <u>programmatically</u>:
    a. Retweets - indicated by what I thought would be a good variable to use: retweeted_status_id.
    b. Dog names (Archive dataset) – saw that quite a lot were misidentified to be names.
        i. Discovered that in the Text column, the name is whatever follows the text string "This is". This led to incorrect names when there is "This is quite…", "This is very…" etc.
            1. I thought of correcting the names by also checking for word following "Named…" or "Name is…" by using some sort of regular/pattern expression. This would still leave the ones that are "This is…" In the end, I ignored the name misspelling altogether.
    c. Rating score loses its decimal precision.
    d. Not all predictions were of dogs. What I find very interesting was that the image predictions algorithm was amazing!! It was able to detect many different objects.

<u>Cleaning Data:</u>

1. During the cleaning process, some quality issues ended up not a big issue after performing tidiness cleaning. For example, during assessing I thought the json tweets file, which contains the "ID" should be named "Tweet_ID" as to be consistent with the other 2 tables. But after joining all the tables, I left the ID as is; although it should be removed from the final dataset but I kept it in there as a check/validation that

the join was working as expected. Also, the in_reply_to_user_id and in_reply_to_status_id rarely have values, but I decided to leave it in there; maybe future analysis could use this.

2. Getting the dog breed: I used whichever the first algorithm (p1, p2, or p3) that determines that it is a dog.
    a. If all algorithm did not determine it was a dog, that record was removed.

3. I converted dog stages to datatype Category but ended up not using it in any analysis.

4. Overall, I cleaned what was discovered during the assessment process by:
    a. First, perform any missing data fixes
    b. Then Tidiness issues
    c. Then Quality issues