Tweepy Data Wrangling Wrangle Report

For this project, I was tasked with gathering data on a series of tweets from the twitter account WeRateDogs using a variety of sources, and then cleaning and analyzing them. The details of this project were designed to simulate a real world example of how combining multiple gatherings of information might be necessary to complete your data analysis.

The document 'twitter_archive_enhanced.csv' was provided by Udacity, and included variables for the text of each tweet, the rating for the dog in each tweet, the timestamp for each tweet, and other features. I used the pandas to_read function to read this into the notebook. Udacity also provided a table from their neural network, with three dog predictions per tweet image, which I downloaded programatically in the notebook. The third source, Twitter's API, was unavailable to me, as I never heard back from them about receiving access to a developer account, but I used the provided .txt file in JSON format and extracted out the retweet and favorite count per tweet.

In assessing the data for this project, I mostly used a programmatic approach to see what could be cleaned. The .info function helped me to find that many columns, such as timestamp, retweet_count, conf, and others needed to be different data types in order to effectively reference them. I also found through visual assessment that the rating for each tweet was in two columns, which violates the requirements for tidy data. I made a number of other observations about the data, which I will address in the cleaning section of this report in more detail. Also, I've included a full list inside the Jupyter Notebook wrangle_act.ipynb.

One issue I faced when cleaning the data was figuring out how to convert the values in the 'doggo', 'puppo', 'pupper', and 'floofer' columns to Booleans. I ended up defining a dictionary with the specific keyword corresponding to True, and ‘None’ corresponding to False, then using the map function in a for loop. In another instance of cleaning, I created a function to distill each of the three image predictions down to one, given that it was True that it was a dog, which I then merged with the twitter archive DataFrame. I also merged the retweet_count and favorite_count DataFrame on the id columns using the merge function. The most difficult part of the data cleaning was using str.extract and regex to try to more correctly extract the rating from each text string, which took several tries, but turned out to be essential as I used it in my analysis. The rest of my data cleaning is fully documented in wrangle_act.ipynb

The problems and challenges posed by this data wrangling project were unique and required an array of well- designed solutions. Overall, I’d say this portion of the project made me more confident about gathering, cleaning, and assessing data from API’s and multiple formats in general