Data Wrangling Report - WeRateDogs Twitter Archive

The overall aim of this data wrangling process was to combine data from 3 different sources, all in relation to certain sets of tweets taken from the WeRateDogs Twitter page. After gathering and assessing the data, my goal was to find at least 8 quality issues and 2 tidiness issues. However, to my benefit, I identified 11 quality issues and 4 tidiness issues.

To account for word tokenization and word cloud visualization later on, I installed **wordcloud** and **spacy** through the pip Python package installer. I also installed **scipy** and **statsmodels** to account for regression analysis and visualization.

Gathering Data

Firstly, Udacity provided an exclusive Twitter archive sent by WeRateDogs for Udacity students to use in this project, named ***twitter_archive_enhanced.csv***. This archive contains 5,000+ of their tweets as of August 1, 2017. I stored this CSV file in a Pandas DataFrame named `arc`.

Udacity provided yet a second dataset called ***image_predictions.tsv***. This file was hosted on Udacity's servers and I programmatically downloaded it using the Requests library. This file contains the breed predictions of each image associated with each tweet, all ran through a neural network. I stored this TSV file, with tab delimiters, in a Pandas DataFrame named `image_predictions`.

Lastly, I needed to gather more data about the tweets that were not given in ***twitter_archive_enhanced.csv.*** To do this, I used tweepy to create a Twitter API object. I then scraped the JSON data of each tweet id listed under that CSV file, then stored each tweet's JSON data to its own line in a file named ***tweet_json.txt.*** The fields I picked to be included in the DataFrame were *retweet_count*, *favorite_count*, and *created_at*. After successfully parsing the data, I read the TXT file line-by-line into a Pandas DataFrame named `tweets`.

Assessing Data

Visual Assessment

I looked through the ***twitter_archive_enhanced.csv*** and ***image_predictions.tsv*** files in Excel. I spotted the following 4 quality issues and 1 tidiness issue:

- Quality
    - Inconsistent and inaccurate readings mapped from text to numerator & denominator fields (e.g. 24/7 being mistaken as a rating).
    - There is one entry with no rating.
    - Nonsensical dog names such as *a, an, quite, by, actually, such, not, one.*

    - Unnecessary anchor tags under the *source* field.

- Tidiness
    - *Doggo, floofer, pupper, and puppo* all are not under 1 category variable.

Programmatic Assessment

Between arc and image_predictions, I used a helper function to identify the tweet ids they had in common and which ids were missing. I stored these ids in a separate DataFrame. Afterward, I identified the tweet ids in common between arc and tweets. Again, I stored these ids in separate DataFrames, then I checked which missing tweet ids that image_predictions and tweets had in common. Afterwards, I discovered these 2 quality issues.

- Quality
    - arc contains 281 tweets that are missing from image_predictions.
    - arc contains 22 tweets that are missing from tweets. After dropping previous entries, there should be 13 tweets to be dropped from arc.

After dealing with missing data, I invoked the .info() function on each DataFrame to check for inconsistent or wrong data types. I found the following 1 quality issue:

- Erroneous data types under *in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, timestamp,* and *retweeted_status_timestamp* in arc.

I invoked .value_counts() on some of the fields in all of the DataFrames. Additionally, I looked at the text content of the entries which are retweets, which indicated that there may be duplicate tweets. I found the following 3 quality issues and 3 tidiness issues:

- Quality:
    - Retweeted data indicates duplicated tweets.
    - *rating_denominator* has values besides 10.
    - Inconsistent timestamp format between arc and tweets.
- Tidiness:
    - *favorite_count* and *retweet_count* fields under tweets need to be merged with arc.
    - *p1, p1_conf, and p1_dog* under image_predictions need to be merged with arc.
    - dog breeds under the *p1* column have untidy formatting.

I also looked at each unique *rating_numerator* individually under arc and spotted the following 1 quality issue:

- Some numerator values contain a decimal point (e.g. 9.75/10).

Cleaning Data

Before I dealt with the aforementioned issues, I created a clean copy of arc into a new DataFrame named arc_clean. I cleaned the data in 3 stages for each issue: Define, Code, and Test. For each of these stages, I defined an action I needed to take, implemented that action in code, and tested if it was done correctly.

Storing Data

After I was satisfied with the cleaned data, I stored arc_clean into a file named ***twitter_archive_master.csv.***