Wrangle Report

Fatema Buhuligah

The dataset that I will be wrangling (and analyzing and visualizing) is the tweet archive of Twitter user @dog_rates, also known as WeRateDogs. WeRateDogs is a Twitter account that rates people dogs with a humorous comment about the dog.

I followed the wrangling data process Gather, Assess and Clean as I learned in the curriculum of Wrangling Data.

1. Gathering Data

I gathered data from various sources and a variety of file formats using Python.

- The first source is File on Hand, given by Udacity The WeRateDogs Twitter archive as **twitter_archive_enhanced.csv**.
- The second source is Downloading Files from the Internet **image_predictions.tsv** is hosted on Udacity's servers I download it programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv.
- The Third source is Twitter APIs. I query the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data in a file called **tweet_json.txt** file.

2. Assessing Data

Assess data visually, through looking at each table of the data frames, and assess data programmatically, for quality and tidiness issues using pandas. Since Pandas collapse rows, it has best done using programmatic assessment. After I assess the three data frames (df1, df2 & df3) here are the issues I found:

Quality

- **In df1,** we focus on original ratings tweet (no retweets) and there are records with retweets.
- **In df1,** no need for these columns (retweeted_status_id, retweeted_status_user_id & retweeted_status_timestamp) since we focus on original ratings tweet (no retweets).
- **In df1,** we focus on original ratings tweet that have images and there are tweets without image.
- **In df1,** no need for (in_reply_to_status_id & in_reply_to_user_id) columns since we focus on original ratings tweet.
- **In df1,** tweet_id & timestamp columns have datatype of int64 & object instead of object & datetime respectively.

- **In df1,** source column contains html tag <a.>
- **In df1,** there are more than one issue in rating_numerator & rating_denominator columns:
    - some rating values are not extracted properly
    - some tweets have multiple rating values
    - some rating values contain decimal
    - some ratings have scales different than 10
- **In df1,** name column has different entries with lowercase and with non-name entries like ('a', 'an' and 'such').
- **In df2 ,** the tweet_id column has different Data type int64 .
- **In df3,** the Data type of retweet_count and favorite_count columns should/better be int64 instead of object.

Tidiness

- **In df1,** timestamp contains two variables, whereas each variable should form a column.
- **In df1,** these columns (doggo, floofer, pupper and puppo) should be in one column.
- **df2 and df3,** should be part of df1.

3. Cleaning Data

Using pandas, clean the quality and tidiness issues you identified in the "Assessing Data" part.

- Drop records with retweets.
- Drop retweeted_status_id, retweeted_status_user_id and retweeted_status_timestamp columns.
- Remove tweets without image by dropping records with null expanded_urls.
- Drop in_reply_to_status_id & in_reply_to_user_id columns.
- Change data type of tweet_id from integer to object.
- Change data type of timestamp from object to datetime. (I will do it later after split the column, in issue 12)
- Remove the html tag & keep the text.
- Re-extract the ratings from the tweet text column.
- Drop the tweets that have multiple rating values.
- Convert data type from string to float.
- Drop the rating that has scales different than 10
- Replace each entry with lower case in name column with "None"
- Change data type of tweet_id in df2_clean from integer to object.
- Change data type of retweet_count & Favorite_count from object to integer.
- Split timestamp column into two columns based on the variables.
- Drop timestamp column after splitting.
- Combine the four stages columns in one new column called stage.
- Merge df3_clean & df2_clean with the df1_clean.