Wrangle Report

Gathering data

The first dataset called ‘twitter-archive-enhanced’ was provided by Udacity in a csv format. This file has 2356 tweets and the name was changed to df_archive.

The second dataset called ‘image_predictions.tsv’ in a tsv format, is provided by Udacity's server and using the Request library, was downloaded programmatically. I changed the name to df_images_predictions.

The last dataset was obtained from the Twitter API using Tweepy. With this, we created a developer portal and elevated account to access the API and access token & secret. With this last dataset, I created a dataframe called json_df using only the tweet_id, favorite_count and retweet_count columns.

Assessing data

After gathering the necessary data from the different resources, I assess them visually and programmatically for quality and tidiness issues.

I transferred each data frame into google spreadsheets to complete my visual assessment. And for my programmatic assessment, I used several functions to assess the data such as:

head(): to get the first few rows in the dataframe in order to know test if my dataframe has the right data
tail(): to get the last few rows
info(): to get a concise summary of the columns, range index, data types to explore the data
sample(): to get a random sample of items
isna().sum(): used to look for the missing values
value_counts(): count unique values

Quality issues:

- Some of the dog name columns have other values like ‘none’, ‘a’, ‘such’
- Incorrect timestamp datatype
- Null values were recorded as ‘Nan’ or ‘none’ creating inconsistency
- The ‘text’ column have other values like url’s
- Expanded Url column had duplicated data or other links unrelated to twitter
- The df_archive had retweets
- Some of the columns needed to be removed as there was no analysis performed on them

- Remove all the html tags from values in the source column. Keep only the text between the tags.
- Remove unnecessary columns and keep algorithm's p1 prediction for the image in the tweet

Tidiness issues

- There were 4 different dog stages instead of having 1
- Merge tables: archive with image prediction and json_df_clean

During the assessment, I was able to acknowledge some issues like inconsistent data, incorrect data, invalid data and missing data.

Cleaning data

- I created a back-up for the three dataframes (df_archive, df_images_predictions and json_df).
- I used the template given to us for the wrangling process to define, code and test.
- For each code, I created a copy with the previous issue fixed in order to avoid having to create a new one from scratch if something didn't worked while cleaning the data.
- Used replace function to replace each of the none and other values to group them into one as ‘None’
- I used to_datetime() to separate the timestamp
- Used isna().sum() to verify the null value recorded as Nan and none and replaced them with none
- In the ‘text’ column, split the text with the url into two different columns and called the latter ‘url_short’
- Merged the Floofer, pupper, puppo and doggo into one column called ‘dog_type’ and drop all 4 tables
- Expanded URL column have rows with duplicated data or other website sources so I created a new one called ‘new_expanded_url’ and dropped the original column.
- Remove replies and retweets from archive table and non-matching IDs from image prediction table
- Remove columns that are not necessary: in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, retweeted_status_timestame
- Remove all the html tags from values in the source column. Keep only the text between the tags. Used beautifulsoup library to get the source distribution
- Remove unnecessary columns and keep algorithm's p1 prediction for the image in the tweet
- Merge archive_clean_source with image_clean_drop & json_tab.tweet_id

Finally, the data was stored into a new csv file called ‘twitter_archive_master.csv’.