Data Wrangling of WeRateDogs

Before starting on this project, I did my best on understanding the motivation of this project and descriptions of given datasets.

Gathering Data

I gathered data from "twitter-archive-enhanced.csv" using Pandas library. Similarly, to gather data from image_prediction.tsv, I used Request library.

For the third dataset, since I was able to get all data through tweepy API of twitter I used Udacity provided tweet_jason dataset.

I assessed structure and information about each data frame, and I looked their sample rows, shapes, and descriptions. Then I searched for missing values, duplicated values, unique values of some specific columns such as name, source, and text.

Assesssing Data

Archived_tweet

1. timestamp, retweet_status_timestmp are type objects, need to change into datetime type.

2. missing data: in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, expanded_urls

3. "source" column has HTML tag.

4. retweets indicate that there are duplicate tweets

5. name column has "None" for name of dog and some names are lower case

6. rating_numerator and rating_denominaotr have incorrect entries and they are in object type

image_predictions df

1. tweet_id is integer type

2. jpg_url which means there is duplicate link

3. column names are not meaningful

4. breed names in p1, p2, p3 have underscore, and in lowercase and prediction confidence level is given as decimals

df_tweet_json

1. No duplicated entries

2. column id has integer type

Tidiness

1. Dog stages of dog have four seperate columns, these columns can be formed into one column

2. archive_tweet df, image_predictions and df_tweet_json can be formed into one datagrame by joining betweet tweet_id and id

Cleaning Data

I changed tweet_id column of archived and image dataset to object type. I used str.extract method to combine four columns of dog stage: doggo, floofer, pupper, puppo into one column, "dog_stage" column. This way dog_stage would be a single column. Then those four columns are dropped from the dataframe.

Since data from the two data frames have common tweet_id, I merged image_predictions and df_tweet_json to archive_tweet. Then I merged archive_tweet and tweet_jason to combined_df. Before combining these two files I change column name “id” of tweet_jason df to

“tweet_id”, so they have common column name.

I changed column "timestamp" as datetime type. To do this, I used pd.to_datetime() method.

Since the source link column contained html tag, I replaced the full source link with the main text provided in the source column of the source column. Then I changed source column to “category” type.

In column “name” I replaced all lower-case name with NaN. Then I changed all names into title-case.

Then I remove duplicated entries of column “jpg_url”.

Since there are incorrect rating_numerator and rating_denominator, I extracted correct rating_numerator and rating_denominator. Then I assigned these columns to combined_df, so that previously incorrect columns would be replaced by these columns.

I removed rows that have retweeted_status_id, retweeted_status_user_id, and retweeted_status_timestamp, so that have blank and null retweeted_status_id, retweeted_status_user_id, and retweeted_status_timestamp will stay.

In column p1, p2, and p3 I replace underscore (_) with space. Then I changed the dog_breed of p1, p2, p3 to title-case. Then I changed confidence interval in columns p1_conf, p2_conf, p3_conf to percentage. Then I change column names into meaningful names.

Finally, I saved the cleaned data frame to “twitter_archive_master.csv”.

Now, this data set is ready for analyzing and visualization.