Wrangle report

The purpose of this document is to report step by step things done for data wrangling in the Jupyter notebook.

1. Gathering data

For data gathering we imported three pandas data frames:

- **Twitter_archive** that was provided by Udacity as a .csv file called “Twitter-archive-enhanced.csv”.
- **Image_predictions** that provided from a tsv file downloaded using requests module (file name was “image-predictions.tsv”).
- **Tweets_df** built using Twitter API via Tweepy library. First we scraped tweepy data and stored in a json, then converted to a txt and imported txt for then converting to a pandas data frame. We only picked from the Json tweet_id, retweets and favorites.

2. Assessing data

First of all we inspected data using head(), shape and info() method. Some things called our attention, so we discovered some problems about Quality and Tidiness of the data.

These are the problems we found.

About Quality.

Twitter_archive table

1- Dog names in twitter_archive weren’t accurate, some names are a and None. There were no null values if you inspected with info() method but then you detected there were a few mistakes.

2- Some of the dog’s names that appeared as "None" had computation errors

3- Source variable included the html tags.

4- There were retweeted rows since retweets are essentially duplicated of the actual rows.

5 - We had incorrect values in rating numerators. Some have not been properly cleaned and did not include decimals

6 - We had incorrect values in rating denominators. 23 of them weren't 10, even 1 is 0.

Image_predictions table

7- There were some predictions such as web_site. Surely they were not dog breeds.

8- p1 variable, p2 variable and p3 variable did not have a standardized use of spaces. Some of them use "_" and some "-". They should always use the same standard.

9- They had not applied a specific criterion about Capital letters in p1, p2 and p3 variables. Some had Capital letter, others no.

tweets_df table was OK, no null values and no weird things about that data.

About Tidiness:

1 – Data frames were not joined. The variable for joining them was tweet_id and did not have the same format. After cleaning we had to change that for joining data

2 – Dog stages needed to be combined into one column (doggo, floffer, etc)

3 – Dropped columns we were not going to use.

3. Cleaning data

In cleaning data we proposed a solution for every problem we identified. Some of the problems were solved together.

Problems 1 and 2 about Quality were solver droping Name column.

Problem 3 was solved using a regex to extract the text inside the html format.

Problem 4 was solved deleting retweeted rows (those that were not null in retweeted_status_id_column)

Problem 5 was solved using regex for extracting from text column the decimal and the integer. Then turning the column into a float

Problem 6 was solved droping rating_denominators that were different to 10. Then converting rating_denominator into float.

Problem 7 was solved discarding the rows without dog breed predictions.

Problems 8 and 9 were solved standardizing strings with pandas.str.replace method and pandas.str.capitalize() method.

Tidiness issues.

Problems about tidiness were solved first changing to str in every dataframe the tweet_id column, then merging them into two steps.

Then we melted dog stages concatenating variables and using str.replace method.

After that we removed variables we were not going to use after. The removed variables were: 'in_reply_to_status_id', 'in_reply_to_user_id', 'retweeted_status_id', 'retweeted_status_user_id', 'retweeted_status_timestamp', 'jpg_url', 'img_num', 'expanded_urls', 'doggo', 'floofer', 'pupper' and 'puppo'.

Finally we saved the cleaned and tidy data frame with the twitter_archive_master and stored as a csv file.