Data Wrangling: We Rate Dogs

Data Gathering

Data for the project came from three sources:

- Original twitter archive data: csv provided in the resource tab, were initialized in tw variable. In this gathering step, I just download the file from the resource tab in the project.
- Predictions data: programmatically downloaded from Udacity , were initialized in img_pred variable. In this gathering step, I downloaded the data programmatically using request library, and then after I created a file called “image_predictions.tsv” and stored the data on it. Then using read_csv function provided from pandas library I got the data and put it in a variable.
- Addition twitter data: obtained from the Twitter API using Tweepy, were initialized in tw_data variable. In this gathering step, I got the API keys from Twitter, then using Tweepy I got the data I want using get_status function and stored the data in tweet_json.txt with each tweet line.

Assessing Data

Quality

Here, I found that the source column has an HTML tag called <a> which illustrate link.

Also, rating_denominator and rating_numerator column had some invalid data.

expand_urls have missing data.

Dogs have names None

timestamp and retweeted_status_timestamp should be datetime not Strings

Some name of dog is less or equal than 2 character

We only want original ratings (no retweets) that have images.

some tweet_ids have the same jpg_url.

Tidiness

column timestamp separate them into two columns Date and Time

last four columns are stages for the dog, better we make it one column called stage using melt function

we do not need the tweet_id and jpg_url columns in img_pread table

tweet_id column not needed in tw_data table

Cleaning

First of all, in order to make the cleaning easy I decided to merge all the tables into one table, and that was the first thing I did.

Source quality issue was easy to solve using apply function and findall form re library.

For the rating_denominator what I did is making all the rating 10 which make sense, because when assessing data, I found out that the rating max was above 100! So, I decided to make them all 10

Also, for the rating_numerator I kept the rating 15 and less, because the max of the rating was above 1000! Which does not make sense.

In the expanded_urls, I just get the index of the missing values and went in a loop with and add the each tweet id in the end of the url.

Any dog name None change to NaN.

I separated the timestamp into two columns Date and Time using split function.

I kept only the tweets with the original ratings.

The last four columns 'doggo', 'floofer', 'pupper', 'puppo' the stages of the dog, i wanted to keep them in one columns for tidiness, so I used melt function to do that.

Finally I stroed the cleaned data in a twitter-archive-master.csv file.