WRANGLE REPORT

By: Jorge Muñoz Rama

Introduction

- The objective of this report is to describe the effort made to gather, access and store data from the Twitter account called WeRateDogs. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog.

Gathering Data

The data collected in this project comes from three files:

- The twitter-archive_enhance.csv and tweet-json.txt files; which have been provided by Udacity and we have opened them programmatically using pandas.read_csv method.
- The image-predictions.tsv file, which we have downloaded programmatically from the link https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv', to do this we have used the request package and pandas.read_csv.

Assessing Data

We used different methods like info(); isna().sum(), Values-count(), duplicated(), groupBy, query, str.extractall, head() and tail(). The principal issues that we found was:

Quality

- timestamp column is a string, it should be datatime.
- The source colum has 4 diferent type of source but the information is among html tags.
- The column text are retweet and all of them star with (RT @) we are going to drop them
- The column name didn't have NAN, We can see that there are 745 None values in this column. That amount of NAN (None) is huge and at the same times there are value like a, an, the, which are not names, all of them start with lowercase, 109 names that star with lowercase where none of them are dog names. They extracted the names of the dog from the column text after the phrases "this is" or "here is", but some names , appear right after 'named.
- In expanded_urls, there are some repeated url in the same field or doesn't star with https://twitter.com/. We are going to clean this column extracting the correct patron like https://twitter.com/dog_rates/status/889531135344209921/photo/1 or video.
- We find values in the rating_numerator less than 10 and some values so high like 1776, 960 or 666.
- The rating-denominator had some values in the denominator different than 10.

- We found that in the column text there is a patron where the rating is at the end of the tweet and also there are some values in the numerator in the text column with decimal places that is causing problems as well as we found in the same tweet more the one rating patron. It seem that they extracted the rating from the first one.
- We saw that some of the row in the column text are retweet and all of them star with (RT @) we are going to drop them
- image_prediction has 66 duplicated jpg_url (imges).
- We got 324 rows where all prediction(p1_dog, p2_dog and p3_dog) are false, some prediction are not dog.

Tidiness

- doggo, floofer, pupper and puppo columns, we created a new column stage and traid to get more data from de text column. We saw that some tweets have more the one dog in the picture and more the one stage, so we trie to get that form the column text.
- We used rating_numerator and rating_denominator to create a new column called rating.
- merging the three data frame in only one dataframe by tweet_id

Cleaning Data

- Make a copy of the three dataframe
- Get the information among html tags in the source column.
- Convert timestamp column to datatime
- We extracted from tex column the stage of the dog. In our case we got those stage: pupper,doggo, puppo, floofer, puppers,doggos, puppos, puppo-doggo, pupper-doggo, doggo-floofer. We made that clasification because there are more than one dog in the tweet with more of one stage.
- We created a new column called stage.
- We drop the columns pupper,doggo, puppo,floofer.convert None value by nan in name column.
- convert None value by nan in name column.
- Change the format of the name column to lowrcase
- We got more names frome the text column after the word named.
- We extrated from expanded_urls just valid link like this format https://twitter.com/dog_rates/status/670444955656130560/photo/1.
- We cleaned some row in the column expanded_urls where there were a repeated valid url.
- We droped rows where expanded_urls have nan's (not images).
- we got all the possible rating in the column text. The last at the final of the tweet is the valid rating, to do that we used str.findall().
- We split into two new column numerator and denominator and made equal to the rating:numerator and rating_denominator and after that we drop numerator and denominator.
- We saw that the row 516 doesn't have rating, 24/7 is a date and the are a rating_numerator

value equal to 1776 which is a outlier we droped those rows.

- We droped all the rows in the column text where RT @, because are retweets.
- We change to lowercase in the image_clean dataframe the column p1,p2 and p3.
- We droped 66 duplicated values rows in jpg_url column.
- There are 318 rows where p1_dog, p2_dog and p3_dog are False, they are not dogs or he algorithm is no able to recognize as dog the image. we droped them.
- We used wide_to_long panda function, but firstly we have changed the name of the column. The idea is to have just 3 columns (p, p_conf and p_dog).
- We grouped by tweet_id and chose the last one of each tweet_id group, using last().
- We changed in witter_count_clean the tweet_id column using astype(int).
- Merge archive_clean. with twitter_count_clean, on='tweet_id'. and created archive_master dataFrame.
- we chcked there were not 'favorite_count == 0'.
- We merged archive_master and image_clean.
- we renamed the columns in archive_master before to store the dataFrame
- And finally we stored the archive_master as csv file named 'twitter_archive_master.csv'.