wrangle_report

August 29, 2022

0.1 Reporting: wragle_report

- Create a **300-600 word written report** called "wrangle_report.pdf" or "wrangle_report.html" that briefly describes your wrangling efforts. This is to be framed as an internal document.

0.2 Data wrangling

0.2.1 Data gathering

Following the imporatation of all necessary libraries for the project, I imported the "twitter-archive-enhanced.csv" file using pandas and assigned it to a dataframe.

Afterwards, I imported the 'image_prediction.tsv' file using requests from the url given. i then read it into a pandas dataframe.

Thereafter, I created a tweepy API object using the key, secrets and token which I obtained from the twitter developer account I had created. I made a list of Ids from the tweet_id column in the dataframe i created from twitter-archive-enhanced.csv. I then queried tweepy.api for the tweet_ids listed and the json files of each of the tweet_ids written into the tweet_json.txt file. Errors files were printed using exception.

After downloading, I read each line of the tweet_json.txt file into a list called data then read data into a dataframe(df) and selected my columns of interest which I stored in tweepy_data dataframe. This aspect posed the greatest challenge during the project and this capped the data gathering phase.

0.2.2 Assessing

I assessed the 3 dataframes visually (using pandas and excel) and also programmatically using a number of python functions such as .info(), .duplicated(), .value_counts() etc. I picked up on a number of structural and quality issues within the dataframe which I documented as follows:

Quality issues

1. There are 181 retweeted tweets of the 2356 tweets in the twitter_archive_clean dataframe

2. Dog stages with null values represented as none

3. rating_denominator not equal to 10

4. extreme rating_numerator values

5. tweets after August 1st, 2017 have no image prediction data

6. poorly written column names (img_num, p1, p1_conf, p1_dog, p2, p2_conf, p2_dog, p3, p3_conf, p3_dog)

7. tweet not about dogs (e.g tweet_id 666104...)

8. NULL dog name represented with none
9. images showing not dog rated in image_prediction table

Structural issues

1. twitter_enhanced_clean and tweepy_data_clean in different tables

2. Timestamp column containing date and time.

3. dog stages split into 4 columns (doggo floofer pupper puppo).

0.2.3 Data Cleaning

I started the cleaning process by first making, of the dataframes, there after i started tackling the structural issues also known as tidiness issues using the define, code, format.
Issue1: I merged the twitter_enhanced_clean and tweepy_data_clean dataframes using the tweet_id and id as keys in the respective columns
Issue2: I converted the timestamp column from obj. to the datetime datatype them extracted the date and time into their respective colums and dropped the timestamp column.
Issue3: I removed tweets that returned a value for retweeted_status_id.
Issue4: While the numerator can be more than 10, I felt it was important for the denominator to be uniform so I made the denominator 10 for ratings rating several puppies then used the .update option to update the dataframe and dropped all the remaining column without 10 as it's denominator
Issue5: I dropped rating_numerators > 15 in order not to skew the data.
Issue6: I filtered off tweet_ids after August 1, 2017
Issue7: I renamed poorly written column names in the image_predictions_clean dataframe
Issue8: I removed tweet_id = 666104133288665000 which I discovered during my visual assessment which was not about dog but chicken
Issue9: Replaced the none values with nan in the name column of twitter_enhanced_clean
Issue10: Used Regex to extract the dog_stage from the text column and dropped the doggo,pupper,floofer and puppo columns
Issue11: Finally, I used image_prediction_clean to remove the rows that were predicted not to be dog in the 3 prediction trials.

In [ ]: