Wrangle Report
Cyril Ocloo

July 7, 2022

1 Data Wrangling Report

1.1 Introduction

The objective of this project was to wrangle the datasets (twitter archive, tweet_json file, image prediction file) by gathering, assessing and cleaning cleaning them for analyzing and visualizing.

The dataset is the tweet archive of Twitter user @dog_rates, also known as WeRateDogs. WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. These ratings almost always have a denominator of 10. The numerators, though? Almost always greater than 10. 11/10, 12/10, 13/10, etc. Why? Because "they're good dogs Brent." WeRateDogs has over 4 million followers and has received international media coverage.

The goal of this report is to provide a briefly describe the data wrangling techniques that I used to gather, assess and clean the dog twitter archive.

1.2 Gathering Data

I gathered the following files for the analysis and visualization.

- **The twitter archive (WeRateDogs)** This archive.csv file was manually downloaded from the WeRateDogs Twitter Archive. The downloaded dataset which consist 2356 tweets was read using pandas command for reading `.csv` files and assigned to `df_1`.

- **The tweet image predictions** The image prediction file was programmatically downloaded by using the Requests library(`requests.get`) to retrieve the tweet image prediction (image_predictions.tsv) from the URL `'https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-` The retrieved data was saved and created a dataframe for the saved data. Since the file is a `tsv` file which is separated with tabs, I added `sep='\t'` to the pandas command for reading `.csv` files and assigned to `df_img`.

-

Each tweet's retweet count and favorite ("like") count My twitter developer account application was not approved so I downloaded the file (`tweet_json`) which contains the JSON data for each tweet and read it to extract the needed features ie (retweet and like counts).

1.3 Assessing Data

After gathering all three pieces of data(`df_1 , df_img, tweet_json`), they were then assessed visually and programmatically for quality and tidiness issues. The following are some of the quality and tidiness issues with the dataset gathered.

1.3.1 Quality Issues

Twitter archive (`df_1` table)

- 2278 missing values in in_reply_to_status_id, in_reply_to_user_id column
- 2175 missing values in retweeted_status_id, retweeted_status_user_id,retweeted_status_timestamp
- Remove non-empty row from the column above to avoid double counting since retweets are essentially duplicates of the actual tweets
- 59 missing values in expanded_urls.
- some names such as 'a', 'an','the','not','such','None' etc which starts with lowercases are not dog names.
- timestamp and retweeted_status_timestamp has datatype object instead of date_timestamp.
- rating_numerator is stored as int instead of float.
- indices 313,1068,1012,1165,1662,2335, 45,340,695,763,1689,1712 were incorrectly extracted
- indices 516, 342 and 1663 has no rating (drop)
- tweet_id is stored as int instead of float.
- name column can renamed as dog_name
- dog_rating column can be created and normalized out of 10 using rating_numerator and rating_denominator columns in df_1 dataset and dropping rating_numerator and rating_denominator columns after.

The tweet image predictions (`df_img` table)

- p1,p2,p3 column name I believe is dog breed need to be renamed.
- tweet_id datatype is saved as int instead of object datatype.
- some of the images are not for dogs.
- Attributes p1,p2,p3, p1_conf, p2_conf, p3_conf can be divided into two separate columns with p1,p2,p3 column represented by dog_breed and p1_conf, p2_conf, p3_conf represented by predicted confidence

Each tweet's retweet count and favorite ("like") count (`tweet_json` table)

- id datatype is saved as int instead of object datatype.
- id column rename to tweet_id for consistency.

1.3.2 Tidiness issues

`df_1` table

- Columns doggo,floofer,pupper,puppo can be categorized in to one column.

- doggo, floofer, pupper, puppo column has none values which I think should be Null/NaN.

`df_1`, `df_img` , `tweet_json` table

- All three dataset can be merged into one master dataset

1.4 Cleaning Data

Firstly, a copy of all the 3 dataset were made and cleaned programmatically by tackling the quality and tidiness issues stated in the assessing data stage. During the cleaning process, use the define-code-test framework and clearly document it. In the end, all the 3 dataset were merged into one master pandas dataframe. These are some of the programatic cleaning techniques used:

Twitter archive (`df_1` table)

- Remove non-empty row from the column below to avoid double counting since retweets are essentially duplicates of the actual tweets.
- These following columns has over 2100 NaNs and are irrelevant to my analysis and had to be dropped
  - in_reply_to_status_id
  - in_reply_to_user_id
  - retweeted_status_id
  - retweeted_status_user_id
  - retweeted_status_timestamp
- Change timestamp datatype from object instead of date_timestamp by using pandas command `to_datetime`.
- Convert rating_numerator and rating_denominator to float from int by using the astype command.
- Create a dog_rating column
- Normalized the dog_rating column out of 10 using rating_numerator and rating_denominator columns in df_1 dataset by dividing the rating_numerator / rating_denominator and then mulitiply by 10.
- drop rating_numerator and rating_denominator columns .
- Change tweet_id datatype from int to object.
- Correct some indices whose denominator and numerator ratings were wrongly entered with the right ratings.
- Drop indices with no rating.
- Created a dog_cycle column for doggo, floofer, pupper, puppo column which has some none values and replace it with Null/NaN.
- Convert dog_cycle column datatype to a categorical datatype.
- Drop columns doggo, floofer, pupper, puppo.

The tweet image predictions (`df_img` table)

- Write a function extract the dog_breed and predicted confidence from p1,p2, p3 and p1_conf,p2_conf and p3_conf respectively.
- assign a new column name dog_breed and predicted confidence
- drop needed columns p1,p1_conf,p1_dog ,p2,p2_conf ,p2_dog ,p3 ,p3_conf, p3_dog

Each tweet's retweet count and favorite ("like") count (`tweet_json` table)

- Rename id column as tweet_id using column rename command and then change tweet_id datatype from int to object

Merging all 3 dataset to form 1 master dataset

- Merge df_1_clean and tweet_json_clean dataset on tweet_id column (left join) using pd.merge function.
- Named the merged dataset as df_1_clean
- Now df_1_clean and df_img_clean on tweet_id column (inner join) using pd.merge function.
- Name the new merged dataset as the twitter_archive_master

1.5 Storing Data

The cleaned master dataset is saved as a `.cvs` file as `twitter_archive_master`