wrangle_report

August 8, 2021

0.1 Wrangle Report

The dataset wrangled in the project 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.

The wrangle and analyze data project tasks include:

- Wrangling the twitter data through the following processes:
  - Gathering data
  - Assessing data
  - Cleaning data
  - Storing, analyzing, and visualizing your wrangled data
  - Reporting on the data wrangling efforts and data analysis & visualizations efforts

0.1.1 Gathering Data

My wrangling efforts for the wrangle and analyze data project included gathering data from the following sources:

- The WeRateDogs Twitter archive. The *twitter_archive_enhanced.csv* file was provided to Udacity students. WeRateDogs downloaded their Twitter archive and sent it to Udacity via email exclusively for you to use in this project. This archive contains basic tweet data (tweet ID, timestamp, text, etc.) for all 5000+ of their tweets as they stood on August 1, 2017.
- The tweet image predictions, i.e., what breed of dog (or other object, animal, etc.) is present in each tweet according to a neural network. This file was was downloaded programatically using requests. get() function from requests which was then saved in a tsv file named *image_predictions.tsv*.
- Twitter API by creating a Twitter developer account to generate my consumer key, consumer secret, access token and access secret. Also Python's Tweepy library was used to gather each tweet's retweet count and favorite ("like") count at minimum, and any additional data I find interesting which was extracted into the *tweet_json.txt* file.

0.1.2 Assessing Data

After gathering all the necessary data for analysis, I assessed each table to spot the quality issues and tidiness issues that might be present in the data, in order to make our data clean for better analysis and visualization of the data. Here are the fixes below:

Quality

`archive` **table**

- Missing data in (in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, expanded_url) columns
- Timestamp has +0000 at the end. Not in the right format
- Erroneous datatype (timestamp and retweeted status timestamp should be integer instead of string)
- Tweet id is integer instead of string
- Lowercase dog names such as *a, an, the, quite, my, such etc.* are unusual. all dog names in lowercase
- Text shows there are retweets or replies in the data
- Extraction of numerator and denominator are incorrect

`image_predictions` **table**

- Tweet id is integer instead of string
- Names in Algorithm's predictions p1, p2, and p3 are sentence case sometimes, lowercase other times
- Compound names in p1, p2 and p3 columns have underscore sometimes, hyphen other times

`df3` **table**

- Tweet id is integer instead of string

Tidiness

- One variable in four columns in `archive` table (doggo, floofer, pupper and puppo)
- df3 should be part of the `archive` table

0.1.3 Cleaning Data

After assessing the data, I cleaned the quality and tidiness issues listed above one at a time by defining, coding and testing.

- Change timestamp to datetime format
- Change tweet id datatype to string (all three tables)
- Convert all non-dog names to np.nan. All non-dog names start with lowercase
- Filter the null values for the three columns related to retweets and check to verify them.
- The extraction of the numerator and the denominator didn't seem to work fine, as floats were not extracted correctly. The extraction will be executed again and the results shall be stored in the DataFrame
- Change the breed names to the same format for consistency
- Use pandas apply function to make changes to compound names with hypen and underscore for consistency
- Drop the columns in which missing data are present in the archive table. Since they are not going to be used in my analysis

- Replace None variables in the dog stage columns, then combine all stages to one column, separate the joint multiple stages and then convert the missing values to nan. In the end, drop individual stages from the archive table
- Write a loop to select dog breed into predictions variable
- Merge all the tables as one

```
In [ ]:
```