Data Analyst Nanodegree

WRANGLE REPORT

TWITTER DATA WERATEDOG PROJECT

WeRateDogs®

13K Tweets

WeRateDogs®

@dog_rates

Your Only Source For Professional Dog Ratings Instagram and Facebook ➪ WeRateDogs partnerships@weratedogs.com

Adoptable Dogs ➡ campsite.bio/weratedogs Joined November 2015

19 Following 8.8M Followers

Not followed by anyone you're following

Prepared by : Lina Montrimaite

October 30, 2020

Project Details

In this project I wrangled and analyzed the tweet archive of Twitter user @dog_rates, also image prediction file and additional data via Twitter API. The goal was to wrangle Twitter data to create interesting and trustworthy analyses and visualizations.

Project tasks of this are as follows:

- Data wrangling, which consists of:
  - Gathering data
  - Assessing data
  - Cleaning data
- Storing, analyzing, and visualizing your wrangled data
- Reporting on

  1) your data wrangling efforts and

  2) your data analyses and visualizations

Gathering data

The data for the project was gathered from three different datasets described below:

1. **Twitter archive**: It 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. **(file name: twitter_archive_enhanced.csv)**
2. **Tweet image predictions**: Provided three predictions of the tweet image according to a neural network. **(file name: image_predictions.tsv)**
3. **Twitter API data**: This data included retweet count and favorite count was collected using the tweet IDs in the WeRateDogs Twitter archive by querying the Twitter API for each tweet's JSON data using Python's Tweepy library and

storing each tweet's entire set of JSON data in a file called **(file name: tweet_json.txt)**

2. Assessing Data

After gathering data from the three datasets, the data was assessed visually and programmatically for quality and tidiness issues.

The following issues were found:

Quality issues

Twitter archive

- Dataset contains retweets entries and reply tweets entries columns.
- Timestamp column is 'object' data type not datetime.
- Dog name column contains not the dog names.
- Dog stages have four separated columns and repeated stages.
- Source column have ‘object’ data type instead of category.

Tweet image predictions

- Prediction contains not the dog breed.
- Predictions have underscores instead of spaces.

Twitter API

- Tweet id column data type is an object not the sting.

Tidiness issues

- Column’s names p1, p2 and p3 in Tweet image prediction dataset don’t identify the columns content.
- The **Twitter archive, Tweet image predictions** and **Twitter API** datasets should be merged into a single one.

- The sources names in **Twitter archive** are not clear.

3. Cleaning Data

The various quality and tidiness issues found during assessing data were solved in this cleaning step using pandas.

The following cleaning steps was been used:

1. Removed retweets entries and reply tweets columns. (used .isna() and drop() functions)
2. Converted timestamp column data type from 'object' to 'datetime64'. (used pd.to_datetime() function)
3. Combined four columns in one dog_stage column. (Created the "dog_stage" column by extracting stages from text columns using .str.extract() function and creating "stage" columns by combining columns "doggo", "floofer", "pupper" and "puppo" . After comparing and cleaning the columns get one column with the dog stage.)
4. Converted source column data type to category. (used .astype('category'))
5. Eliminated not dog breed predictions. (First I filtered predictions by False on p1_dog, p2_dog and p3_dog which showed that prediction is not dog breed. After I created the list of the not dog breeds and replaced with empty cells used replace function)
6. Replaced "_" from predictions. (used replace() function)
7. Converted tweet_id column to string data type. (used .astype() function)
8. Changed image prediction 'p1', 'p2' and 'p3' columns names.
9. Merged archive, imagine prediction and tweets tables in one dataframe. (First merged 2 datasets and later merged third used pd.merge() function)
10. Made a source name clear. (replaced full source path with source name by using .replace() function.)

Finally, the cleaned data was saved to a csv file, twitter_archive_master.csv.

4. Visualization

The master dataset was then analyzed and visualized to drive insights.