Wrangle_Report

April 4, 2021

1 Wrangle report

1.1 Introduction

The purpose of this project is to asses, clean, and analyze Twitter data. We will be analyzing data from a channel called WeRateDogs (@dog_rates). WeRateDogs is a channel that rates dogs - the comments are funny and ratings are almost always above 10 which is the denominator.

This report will describe what steps we took to wrangle the data

1.2 Project details

We will divide our data wrangling into three parts:

- Gathering data
- Assessing data
- Cleaning data

1.2.1 Gathering data

We gathered our data from three separate places:

- Twitter archive

The Twitter archive was a file one hand which we downloaded manually

- Tweet image predictions

What breed of dog is present in each tweet according to a neural network. This was a .tsv file that we needed to download programmatically from a cloud server.

- Twitter API

Using 'tweet ID' and and WeRateDogs Twitter archive I queryed Twitter API for each tweet's JSON data. I used Tweepy library to store the data in 'tweet_json.txt'. Next I read the data line by line ('tweet_id', 'favorite_count', 'retweet_count', 'created_at', 'source', 'retweeted_status', 'url') into pandas.

1.2.2 Assessing data

Once the data was uploaded to pandas / Jupyter Notebook I used two methods to assess the data:

1. Visual assessment - I opened all data frames and checked visually for and inconsistencies.
2. Programmatic assessment - By using many standart methods like

- info(), columns, describe(), value_counts(), count(), sample(5), head()

Based on this info I came up with the problems listed below:

Quality:

- Align data types between data frame (tweet_id)
- Change dates to the same formats
- Drop unnecessary columns
- Keep only original tweets
- Correct or drop rating_denominator
- Correct or drop rating_numerator
- Drop rows without proper dog names
- Only keep predictions with the highest confidence level
- Correct prediction names

Tidiness:

- Merge Three datadrames into one using Tweet_id
- Merge Dog stages into one column

1.2.3 Cleaning data

The first step I took was to change the type format for tweet_id for all data frames this way I was sure the merging the data will not produce any issues.

Next, I merged the three data frames which helped me have a big picture of the whole data frame.

After that, I removed columns which in my opinion were unnecessary. Moreover based on the analysis of 'Tweet image predictions' I removed p2 and p3 as they had the lowest confidence levels.

Next, I cleaned the Name column by finding and dropping incorrect names - dropped all rows with lowercase names.

The next step was to keep only the original tweets - I accomplished that by removing tweets that had 'retweeted_status_id'.

Merging Dog stages into one column was next on my list. First I removed all the instances where there was a 'None'. Next, I merged to columns, here we had a problem as suggested by the reviewer - some records had multiple stages. In that case, we kept both stages and divided them with a comma.

Next, I cleaned the 'rating_denominator' and 'rating_numerator', as suggested by the reviewer we extracted correct values for both columns from the text and I changed them to floats. Then, I checked for more discrepancies. I found that in multiple cases the rating was given for multiple

dogs in one tweet in that case I divided the rating by the number of dogs and corrected it manually. Moreover, I corrected I did this both for 'rating_denominator' and 'rating_numerator'. After that, I dropped one tweet which in my opinion had an incorrect rating and there was no indication of what was the correct score.

Next, I cleaned the 'prediction names' column using the str.replace method.

The final step was to change the 'timestamp' datetime format.

1.3 Final steps

The finished data frame was saved to 'twitter_archive_final.csv'

[ ]: