WeRateDogs Project: Wrangle Report

Data Analyst Nanodegree Program

Mário Damhur

March 31, 2020

1. Introduction

I will discuss what I did to wrangle this project, starting with the gathering of the dataset, followed by the assessment and finish with the clean up process. Let's dive in!

2. Wrangle Process

2.1 Gather

Gathering is the first step in the data wrangling process. There are three different datasets for this project and all of them have a different method to obtain.

- **twitter_archive_enhanced.csv**: This dataset was provide by Udacity, so I just download the dataset and open with pandas with the name local_df
- **image_predictions.tsv**: This dataset was in the server of Udacity so I need to use request library to download programatically and open use pandas with the name image_predictions_df
- **tweet_json.txt**: To get this dataset we need to use Twitter API and use the id column from twitter_archive_enhanced.csv dataset to extract the json. But, since I had problems with Twitter API, I just downloaded the dataset provided by Udacity for those who have problems with twitter.

2.2 Assess

Assessing is the second step in the data wrangling process. We use this step to identify quality issues and tidiness issues in our dataset. I use the following types of assess.

- **Visual assessment**: Here I just opened the dataframes with pandas and visualized all the tables to identify issues just scrolling through the data.
- **Programmatic assessment**: Here I use the methods of pandas to observe more closely the issues that we can’t see just visualizing the dataset.

2.3 Clean

Cleaning is the third step in the data wrangling process. After identifying the issues on the dataset in the second step of the wrangling process, I start cleaning all dataset to combine them into one single dataset in the final. This is the process I followed:

Quality

- **First of all, replace "None" values to np.nan**

Duplicated quality issues

- Rename tweet_id to id for standardize (local_df, image_predictions_df)
- Erroneous datatype assigned to tweet_id/id column (int -> str)
- Cleaning the values of source column (local_df, tweets_api_df)
- Removing retweets and replies (local_df, tweets_api_df)
- The information of text column is truncated to 50 characters, we could lost information to extract from the text (local_df, tweets_api_df)

local_df table

- Erroneous datatypes to timestamp
- Rename timestamp to date

tweets_api_df table

- Extracting and cleaning ratings properly from the text column

image_predictions_df

- Adjusting the letter case on each value in the prediction columns to have a consistent format
- Providing more descriptive name for the columns about the model predictions

Tidiness

- Remove unnecessary columns
    - local_df
        - source (Duplicate with tweets_api_df)
        - in_reply_to_status_id
        - in_reply_to_user_id
        - text
        - retweeted_status_id
        - retweeted_status_user_id
        - retweeted_status_timestamp
        - rating_numerator (Since we extract on the tweets_api_df)
        - rating_denominator (Since we extract on the tweets_api_df)
        - expanded_urls

    - tweets_api_df
        - text
        - retweeted
    - image_predictions_df
        - img_num
- The last four columns in local_df table (doggo, floofer, pupper, puppo) should be one column contain these values
    - Remove original columns
    - Replace " " to np.nan on the new column
- Combine all tables