wrangle_report

March 8, 2022

0.1 Introduction :

Real world data rarely comes clean. Using Python and its libraries, we will collect data from a variety of sources and in a variety of formats, evaluate its quality and accuracy, and then clean it up. This is called a **data wrangling.** In this file, we will provide a full explanation of the data wrangling process, which goes through three important stages: **1. Gathering data 2. Assessing data 3. Cleaning data**

The dataset that we will be wrangling (and analyzing and visualizing) is the tweet archive of Twitter user [@dog_rates](https://twitter.com/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.

0.2 1. Gathering data

In this step, I gathered all three pieces of data as described below in the wrangle_act.ipynb notebook. #### 1- The WeRateDogs Twitter archive: I Downloaded this file manually by clicking the following link: twitter_archive_enhanced.csv. Once it is downloaded, I uploaded it and read the data into a pandas DataFrame.

2- The tweet image predictions This file (image_predictions.tsv) is present in each tweet according to a neural network. It is hosted on Udacity's servers and I downloaded it programmatically using the Requests library and the following URL: here

3- Data from the Twitter API Gather each tweet's retweet count and favorite ("like") count at the minimum and any additional data you find interesting. Using the tweet IDs in the WeRateDogs Twitter archive, query the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data in a file called tweet_json.txt file. > **Note:** I used tweet_json.txt provided by udacity since Tweeter refuse my API access

1 2- Assessing Data

In this step, I assess them visually and programmatically for quality and tidiness issues
Quality Issue

1.0.1 From df_twt_arch :

1 - In columns: ('doggo', 'floofer', 'pupper', 'puppo', 'name') 'None' assigned instead of 'NaN' for empty missing data **{visual assessment} 2** - columns not needed: ('in_reply_to_status_id', 'in_reply_to_user_id','retweeted_status_id', 'retweeted_status_user_id','retweeted_status_timestamp') - columns ('source' ,'text','name) need to rename to be familliar with users **{visual assessment} 3** - column timestamp dtype should be datetime and split into two columns date and time for better visualisation**{programmatic assessment} 4** - 'tweet_id' must be a string.**{programmatic assessment} 5** - 'source' column contains tag html. **{visual assessment} 6** - column 'name' has values: 'None', 'a', 'O', 'Devón'. **{programmatic assessment} 7** - expanded_urls has missing value and inccorrect urls **{programmatic assessment}** and **{visual assessment} 8** - Rating dinominator must be equal to 10 there are other values: (0, 15, 70, 7, 11, 150, 170, 20, 50, 90, 80, 40, 130, 110, 16, 120, 2) **{programmatic assessment**

1.0.2 From df_img:

9- The predictions ('P1', 'P2', 'P3') columns are not clear and familiar to the reader and have strange predictions (spatula, barrow, minibus, etc) **{programmatic assessment} 10** - Some "tweet_ids" have same "jpg_url", after checking using the urls: (https://twitter.com/dog_rates/status/803692223237865472) (https://twitter.com/dog_rates/status/691416866452082688) and changing the ids they were the same tweet **{programmatic assessment}** - ids img does not exist "Hmm…this page doesn't exist. Try searching for something else": **{visual assessment}** - 759566828574212096 - 802247111496568832 - 851953902622658560 - 842892208864923648 - 861769973181624320 - 873697596434513921 - 888202515573088257

1.0.3 From df_json:

11 - Ivalid urls: (https://…)(https:/…) (https:/t.c…) - 175 duplicated url **{programmatic assessment} 12**- retweet_status has one value 'Original tweet', no need it **13** - Tweets missing retweet count and favorite count **{programmatic assessment}**

1.1 Tidiness Issue

1- doggo, floofer, pupper, puppo these 4 variables shoule be combined into one categorical variable 'dogtionary'. **{visual assessment} 2**- rating nominator, rating dinominator should be one column since rating dinominator always be 10 **{visual assessment} 3**- Dataframes: twitter_archive, image_predictions, and tweet_json, Should be one df (twitter_master_df) **{visual assessment} 4**- in twitter_master_df: expanded_urls and url have same values **{visual assessment}**

2 3- Cleaning Data

Clean all of the issues I documented while assessing.I Performed this cleaning in the "Cleaning Data" section in the wrangle_act.ipynb.

- [x] I have made a copy of the original data before cleaning.
- [x] I have used the Define-Code-Test framework.
- [x] I have documented the Define-Code-Test framework.
- [x] I have documented each issue in a few sentences.
- [x] I have successfully cleaned **all** issues identified in the assessing phase.
- [x] I have created a tidy master dataset with all pieces of gathered, cleaned data.