‘WeRateDogs’ Twitter archive Report

Jessica Sarah Stow | stowjess@gmail.com
ALX-T Udacity Data Analytics 2022 student

Wrangling process:

Step 1: Gathering data

Using pandas’ ‘.read’ function, I gathered the data from three different sources (`twitter-archive-enhanced.csv`, `image-predictions.tsv` and `tweet-json.txt`).

Step 2: Assessing data

I visually assessed each data frame by using Pandas’ ‘. head’ and ‘.tail’ functions to check for quality issues. Since visual assessment is not sufficient to pick up all errors (due to the large size of the data frames), I programmatically assessed each data frame by using Pandas’ ‘.info’ and ‘.describe’ for quality as well as tidiness issues.

The following three tidiness issues were noted:

1. A non-descriptive column header for `df_api` dataframe (column 'id' / 'tweet_id'). This will make it difficult to merge all data frames together
2. Three data frames (`df_archive`, `df_pred` and `df_api`), were used instead of one data frame, however, all three share common variable 'tweet_id' ('id' for `df_api` data frame)
3. The dog stage variable was separated in four separate columns: 'doggo', 'floofer', 'pupper', 'puppo', instead of a single categorical variable.

The following eight quality issues were noted:

For the `df_archive` dataframe:

1. The dataframe contains retweets (181 rows with values for 'retweeted_status_id')
2. The 'name' column has values 'None' and contains names that are words, e.g. "a", "the" and "an".

3. The 'rating_numerator' column has a very large maximum value of 1776, as well as other large values
4. The 'rating_denominator' has values that are not equal to 10 (all We Rate Dogs ratings are a score out of 10)
5. The data frame contained erroneous datatypes ('tweet_id', 'timestamp', 'retweeted_status_timestamp')

For the `df_pred` data frame:

6. The data frame contained tweets without images (2075 tweets with image predictions, 281 without)
7. The dataframe contained erroneous datatypes ('tweet_id')

For the `df_api` data frame:

8. The dataframe contained erroneous datatypes ('id')

Step 3: Cleaning data

In this step I made use of the ‘Define, Code, Test’ method. I explained how I intended on fixing the issue (define), the functions and codes I used to clean the issue (code) and the code I used to determine whether the issue was resolved (test).

Each of the issues addressed in ‘Step 2: Assessing data’ were corrected.

Question for reviewer: ¶

I wanted to change 'in_reply_to_status_id', 'retweeted_status_user_id', 'retweeted_status_id' and 'retweeted_status_user_id' from data type 'float' to data type 'object', but this converted NaN values to string values 'nan' so they were no longer read as null entries.

How do I correct for this? (How can I read 'NaN's as null values when changing to datatype 'str'?)

See the code I used below:

```
# Change to 'str' datatype:
df['in_reply_to_status_id'] = df['in_reply_to_status_id'].astype(str)
df['in_reply_to_user_id'] = df['in_reply_to_user_id'].astype(str)
df['retweeted_status_id'] = df['retweeted_status_id'].astype(str)
df['retweeted_status_user_id'] = df['retweeted_status_user_id'].astype(str)

# View info
df.info() # This showed that all 4 variables that I changed above do not contain any null/NaN values.
```