WeRateDogs Twitter Data- Wrangle Report

By Vedavyas Kamath

October 2019

This report is to outline the approach taken and efforts made to wrangle the WeRateDogs twitter data to firstly gather, then assess and finally clean up the data in order to make it ready for analysis.

Data Gathering:

The data for WeRateDogs was gathered and collected from 3 different sources:

1. The WeRateDogs enhanced archive was made available at the Udacity servers which I had manually downloaded.
2. A URL was given for image predictions file which I have downloaded programmatically using Python.
3. Finally downloaded the entire data for tweets present in the enhanced archive from Twitter by querying the Twitter API and by using the Tweepy library in Python.

After gathering the files mentioned above from various sources, loaded them into different Pandas data frames so as to be assessed!

Assessment:

I performed 2 types of assessments on all 3 data frames that were created while gathering. Since all 3 datasets were not so huge and contained about 2,500 rows only, opened the files in MS-Excel and visually assessed the data to find out any obvious errors or mistakes.

Was able to identify an issue with the twitter archive dataset where could see that some dog names were incorrect. There were some dogs with names like ‘a’, ‘an’, ‘unacceptable’ etc. which made no sense. The main goal of the visual assessment was to see what all information is present in each of the datasets and in what format.

Then went on to do the programmatic assessment of all 3 datasets to find out the 8 quality issues and 2 messy data issues.

Cleaning:

To make tracking of files and datasets easy, first combined data from all 3 datasets into one data frame by matching the ***tweet_id*** which is common across all 3 datasets.

Converted the ***tweet_id*** from integer to string as this is just a unique identified for each tweet and is not required to be analyzed or used for numeric calculations.

The data frame contained entries for re-tweets in ***retweeted_status_id*** and ***retweeted_status_user_id*** which was something we did not want. So filtered out the rows in these columns which were non-null and kept only records with null values. Further also dropped these columns from the data.

Similarly, we also wanted only tweets and not replies from user to those tweets. Replies from users were stored in the ***in_reply_to_status_id*** and ***in_reply_to_user_id*** columns. Again filtered out entries that were non-null to keep only null values in these 2 columns.

We needed only those tweets that had an image also associated with it and so dropped all entries with null value in ***jpg_url*** column which holds the URL of the image.

Identified all the incorrect dog ***name***s listed in the data set and replaces all such entries as **'None'**.

There were dogs that were having multiple dog types. Merged these multiple values together so that it could be further analyzed or removed easily from the data.

Created a new ***dog_type*** variable by melting the data in 4 separate columns ***pupper***, ***doggo***, ***fluffer*** and ***puppo***, so that the data complies with tidy data standards.

Converted the ***timestamp*** column to actual datetime object by removing some extra unwanted characters that were present.

The ***source*** column had only 3 possible values but these values were too long and complicated as they contained some HTML tags. Simplified the values by removing unwanted text by using Regular Expressions in Python.

Finally chose the ***breed*** along with ***confidence*** value that is most likely to be for each of the dogs.

Had observed during visual assessment that the values in ***rating_numerator*** were not correct in the dataset and were not matching the actual ratings given by users in their tweets. So extracted the numerator and denominator of ratings from the tweets and added the correct ratings ***new_numerator*** and ***new_denominator*** to the dataset.

Finally removed all the unwanted columns and exported the data frame to "**twitter_archive_master.csv**" that contains all vital columns from all 3 datasets required for analysis.