Wrangling and Cleaning

Datasets

Nowadays, data are everywhere, with various formats and big amounts. Most parameters related to any references can be categorized or tabulated, so forming the data. Recently, it is a trend to analyze these to get valuable insights.
They can be used in different subjects. Data driven from the sensors around a production line and the warehouses can give new indications or form innovative ideas. Also, Internet now is a rich source of data in different fields, whether for scientific reasons or in social media platforms. And the latter is our study here.

Dogs' rates on Twitter

One of the applications is studying the tweets generated from the WeRateDogs page on Twitter. It shares their followers' dogs' pictures and rate them on a scale. People react to their postings, like and retweet the original tweets.

Gathering

To answer any questions or explore findings, a data shall be present to work on. This was the first step; to gather different data from multiple sources. We wanted to get tweets which we have their IDs. This was the first table containing tweets IDs along with some other info as the posting date, name of the dog and the ratings given from the user. It is an available file with a format of Comma Separated Values (.csv) that then read as Pandas date frame. Each tweet nearly has at least one image which is analyzed using a neural network software and a data file was ready in a Tab Separated Values (.tsv) format, which is only then have been read in a Pandas data frame also. This one was downloaded to the script from a URL programmatically using the Requests library. This file involved predictions of the dogs breeds present in the images along with the reference tweet ID.

As these data was driven basically from Twitter, the Twitter API specifies each tweet as tweet object which is a dictionary of other objects like user, place, creation date, favorite counts and retweet counts. And to make our data complete, these rest of attributes had to be got. So the Twitter API is queried with the help of a Twitter developer account using its authentication and the Tweepy library to get its JSON data as a text file, gathering them by the tweets' IDs in hand, each one in a line. This process took about thirty minutes to retrieve about two thousand tweets. This text file is then read to a data frame.

Assessing

Then I found some quality and tidiness issues with the whole set of data that needed cleaning.

Quality issues:

- Creation time was not in the right format for easier analysis
- Tweets IDs had a wrong data type of integers
- One dog image had multiple dogs' specifications
- Some ratings were read wrongly as the ½ is a half and 24/7 is a time duration

- Wrong naming of dogs not matching with the tweet
- Some tweets IDs (25) did not found their match in the API, so their reference data are without favorite and retweet counts
- No standard for naming the property of the (tweet ID) along the data frames
- Much data has not available values in its properties such as the place of tweeting and geographical info.

In addition to that some tidiness rules have not been met such as the following points.

Tidiness issues:

- Dog stages including (Doggo, Floofer, Pupper and Puppo) based on the page definition of these were not organized in a specific column
- The whole set of data was distributed individually to different tables although their relation to each other.

Cleaning

We now have our data as Pandas data frames need cleaning based on the assessed finding issues. So first the data is copied to new data frames as a backup for the previous ones. Then some missing values columns were dropped. Then figuring out the misrepresentation of dog's breeds and fixing them relative to the actual tweet text. And then comes unifying the four columns for each dog breed to one column holding them for each tweet id row. After that, data types for each property is fixed.

Many trials are made with multiple methods until reaching the most efficient one to detect or fix issues. The following fix was to modify the proper ratings and id property naming. Then the last messy in multiple data divisions; to merge all data frames in one data frame under the same id row. By this point. Data is ready for analysis study.