Wrangle and Analyze Data: `wrangle_report`

Documentation for data wrangling

Introduction

Real-world data rarely comes clean. Using Python and its libraries, the goal of this project is to gather data from a variety of sources and in a variety of formats, assess its quality and tidiness, then clean it. This is called data wrangling. The data wrangling efforts are documented in a Jupyter Notebook, plus showcase them through analyses and visualizations using Python (and its libraries) and/or SQL.

The dataset that to be wrangled (and analyzed and visualized) is the tweet archive of Twitter user @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.

Project Details

The **tasks** in this project are as follows:

- Data wrangling, which consists of:
  - Gathering data
  - Assessing data
  - Cleaning data
- Storing, analyzing, and visualizing your wrangled data
- Reporting on 1) your data wrangling efforts and 2) your data analyses and visualizations

1. Gathering data

The data was gathered from three different datasets described below:

1. **Twitter archive file:** The WeRateDogs Twitter archive: `twitter_archive_enhanced.csv`
2. **Tweet image predictions:** The tweet image predictions (`image_predictions.tsv`) i.e., what breed of dog (or other object, animal, etc.) is present in each tweet according to a neural network.
3. **Twitter API data:** Each tweet's retweet count and favorite ("like") count was collected using the tweet IDs in the WeRateDogs Twitter archive by querying the Twitter API for each tweet's JSON data using Python's Tweepy library and storing each tweet's entire set of JSON data in a file called `tweet_json.txt` file.

2. Assessing Data

After gathering each of the above pieces of data, the data from the three datasets was assessed visually (using `Excel`) and programmatically (using `pandas`) for quality and tidiness issues.

The following issues were found:

Quality issues

Twitter archive file

- archive data contains retweets.
- `tweet_id` is an integer
- `timestamp` is of 'object' datatype.
- name has values that are string 'None' instead of `NaN` and some values have unusual names of less than 3 characters such as 'a'.
- Some of the ratings are wrongly mentioned e.g. in one case, the rating should've been 13/10, not 960/00, while some tweets are not about dogs, so doesn't contain rating (tweets not containing dog images can be discarded), and some of the ratings contain decimal in the numerator.
- `NaNs` represented as 'None' (str) for name, doggo, floofer, pupper, and puppo columns.

Tweet image predictions

- There are some missing rows in images dataset (2075 rows instead of 2356): either the rows are missing or some tweets didn't have dog images.
- There are some duplicate `jpg_urls`.
- p1, p2, and p3 contains underscores instead of spaces in the string.

Twitter API data

- There are some missing tweets compared to the data of archive.

Tidiness issues

- There are 4 different columns (doggo, floofer, pupper, and puppo) for dog stages.
- The different dataframes should be merged into a single one.

3. Cleaning Data

The various quality and tidiness issues found during assessing data were solved in this cleaning step using `pandas`.

First, all three datasets were merged into a single dataset (dataframe). Then, the various tidiness and quality issues (e.g. related to `name` and `rating`) were solved.

Finally, the cleaned data was exported to a csv file, `twitter_archive_master.csv`.

4. Visualization

The master dataset was then analyzed and visualized to drive insights.