

# Data Wrangling Report

## 1.1 Gathering Data

### The Dataset(s)

The dataset to be wrangle is the tweet archive of Twitter user [@dog\\_rates](#), also known as WeRateDogs. This archive/dataset consists of 2356 tweet data observation from November, 2015 to August, 2017. WeRateDogs is a Twitter account that introduces and rates people's dogs with funny, humorous comments about dogs.

From the images in the dataset above (i.e. WeRateDogs Twitter archive), another dataset is created which entails of image predictions (three predictions) alongside each tweet ID, image URL, and the image number that corresponded to the most confident prediction (numbered 1 to 4 since tweets can have up to four images). The image prediction dataset as well as the API dataset won't be cleaned directly. Some cleaning will be computed after merging them to our tweet archive dataset.

### Gather Twitter archive from a CSV file

Using the link given by Udacity, the WeRateDogs Twitter archive dataset is manually downloaded as [twitter\\_archive\\_enhanced.csv](#) file and imported this file into a dataframe (df\_twitter).

### Gather tweet image predictions

The tweet image predictions file hosted on Udacity's servers is downloaded programmatically using Python's Requests library and it is saved locally to image\_predictions.tsv file. Afterwards, this file has been imported into a Python Pandas dataframe (df\_image).

### Gather data from the Twitter API

Using the tweet IDs in the Twitter archive, the entire dataset is being checked for every tweet from Twitter API and stored every tweet's entire set of JSON data in a file called tweet\_json.txt file. The df\_api dataframe is imported from this JSON including only tweet\_id, retweet\_count & favorite\_count columns.

## 1.2 Assessing Data

### Visual Assessment

By opening the twitter\_archive\_enhanced.csv and image\_predictions.tsv and scrolling through them with pandas library, 1 quality and 2 tidiness issues were spotted:

**Quality:** unnecessary html tags in source column of twitter archive in place of utility name e.g. Twitter for iPhone.

**Tidiness:** doggo, floofer, pupper and puppo columns in df\_twitter table should be merged into one column named "stage" Tidiness: Twitter archive data without any duplicates (i.e. retweets) will have empty retweeted\_status\_id, retweeted\_status\_user\_id and retweeted\_status\_timestamp columns, which can be dropped.

### Programmatic Assessment

Pandas' info method has been used on df\_twitter to spot erroneous datatypes and other quality

issues. Right after, the `value_counts` method was used on various variables such as on `rating_numerator`, `rating_denominator` columns to look up outliers as well as the range of their values and its distribution. Also the columns stage and name have been checked through various method to see if the names were clean and if no observation had more than one dog stage.

Therefore, 7 quality issues have been identified after scrutiny:

- contains retweets and therefore, duplicates
- many `tweet_id(s)` of `df_twitter` table are missing in `df_image` (image predictions) table
- erroneous datatypes (timestamp columns)
- `rating_numerator` column has values less than 10 as well as some very large numbers (e.g. 1176)
- `rating_denominator` column has values other than 10
- erroneous dog names starting with lowercase characters (e.g. a, an, actually, by)
- some records have more than one dog stage

The `info` method on the other 2 dataframes (`df_image` and `df_api`) didn't reveal any major quality issues that needed to be fixed before merging.

Although, 2 tidiness issues will have to be fixed:

- "breed" column should be added in `df_twitter` table; some computation should be carried out to make the breed and confidence information more readable and easier to use
- `retweet_count` and `favorite_count` columns from `df_api` (tweet status) table should be joined with `df_twitter` table.

### 1.3 Cleaning Data

For each quality/tidiness issue, the programmatic data cleaning process took place in 3 stages Define, Code & Test. During the cleaning process, we moved from three datasets to one: `merged_df`.

### 1.4 Storing Data

After the completion of the cleaning process, the `merged_df` DataFrame has been stored and saved in `twitter_archive_master.csv` file.