**Data Wrangling Report for Twitter-@WeRateDogs**

1. **Gathering the data**

**About the datasets**

The dataset I'll be wrangling is the tweet archive of Twitter user @dog\_rates (<https://twitter.com/dog_rates>) , also known as WeRateDogs. This archive/dataset consists of 2356 basic tweet data from November, 2015 to August, 2017.

WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. Based on the images in the above dataset (i.e. WeRateDogs Twitter archive), another dataset is created which consists of image predictions (the top three only) 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). Though no wrangling will be done directly on this image predictions dataset, it will definitely provide some additional data for our main tweet archive dataset.

**Gather Twitter archive CSV file**

Using the link provided by Udacity, I downloaded the WeRateDogs Twitter archive manually as twitter\_archive\_enhanced.csv (<https://d17h27t6h515a5.cloudfront.net/topher/2017/August/59a4e958_twitter-archiveenhanced/twitter-archive-enhanced.csv>) and imported this file into a dataframe (twitter\_arc\_df).

**Gather tweet image predictions**

I downloaded the tweet image predictions file hosted on Udacity's servers programmatically using Python's Requests library and saved it locally to image\_predictions.tsv file. Then, I imported this file into a Python Pandas dataframe (twitter\_img\_df).

**Gather Status data from tweets**

I downloaded every tweet's entire set of JSON data in a file called tweet\_json.txt file manually. Created a dataframe twitter\_api\_df from this JSON including only tweet\_id, retweet\_count, favorite\_count and display\_text\_range data.

1. **Assessing the data**

**Visual Assessment**

I used Microsoft Excel to explore the twitter\_archive\_enhanced.csv and image\_predictions.tsv and I was able to spot the followng 2 quality and 2 tidyness issues:-

* Quality- unnecessary html tags in source column in place of utility name e.g. <a href=""[http://twitter.com/download/iphone""rel=""nofollow"">Twitter](http://twitter.com/download/iphone%22%22rel=%22%22nofollow%22%22%3ETwitter) for iPhone
* Quality - text column contains unneccessary text
* Tidyness - Reshape twitter\_arc\_df- doggo, floofer, pupper and puppo columns should be merged into one column named stage
* Tidyness - twitter\_arc\_df - Original tweets will have empty retweeted\_status\_id, retweeted\_status\_user\_id and retweeted\_status\_timestampcolumns, which can be dropped

**Programmatic Assessment**

I used pandas' info method on twitter\_arc\_df to spot erroneous datatypes and other quality issues, if any. Then I used value\_counts method on rating\_numerator, rating\_denominator and name columns to look up the range of their values and its distribution. Also to verify 1 tidiness issue that I found during the visual assessment, I queried the archive dataframe to see if any of its tweets has more than one dog-stage mentioned. This entire activity helped me to identify the following **7 quality issues.**

* also contains retweets other than original tweets
* many tweet\_id(s) of twitter\_arc\_df table are missing in twitter\_img\_df (image predictions) table
* erroneous datatypes in in\_reply\_to\_status\_id, in\_reply\_to\_user\_id and timestamp columns
* rating\_denominator column has values other than 10
* incorrect dog names starting with lowercase characters (e.g. a, an, actually, by)
* some records have more than one dog stage

After taking a look at the sample of each of these dataframes, I was able to identify the following 2 tidiness issues: "breed" column should be added in twitter\_arc\_df table; its values based on p1\_conf and p1\_dog columns of twitter\_img\_df (image predictions) table retweet\_count and favorite\_count columns from twitter\_api\_df (tweet status) table should be joined with twitter\_arc\_df table.

1. **Cleaning the data**

As all the quality and tidiness issues were related to twitter\_arc\_df table, I created a copy of only this table and named it twitter\_arc\_df\_clean. For each quality/tidiness issue, I performed the programmatic data cleaning process in 3 stages - Define, Code & Test. During the cleaning process, I converted the datatypes of source and newly created stage columns of twitter\_arc\_df\_clean to category datatype.

1. **Storing the data**

After the completion of the cleaning process, I stored the archive\_clean DataFrame in twitter\_archive\_master.csv file.