Twitter Data Wrangling Report

The goal was to wrangle *WeRateDogs* Twitter data and analyze it to create interesting analyses and visualizations.

Gathering Data

Three pieces of data needed to be acquired from different sources as described below. This involved dealing with different data acquisition methods and different file formats.

Data description	Source	Gathering Process
The WeRateDogs Twitter archive in a *csv file*.	File provided by Udacity	▪ Manual Download ▪ Loaded into a dataframe in the workspace
The tweet image predictions in a *tsv file*. It contained the dog breed predictions from a neural network classifier	File hosted on Udacity's servers (URL provided)	▪ Downloaded programmatically using the Requests library ▪ Loaded into a dataframe in the workspace
Additional data (retweet count and favorite count)	Twitter API	▪ Queried the Twitter API for each tweet's JSON data using Python's Tweepy library ▪ Stored each tweet's entire set of JSON data in a text file. ▪ The text file was read line by line into a Pandas DataFrame

Assessing Data

The data was visually assessed by scrolling through the csv file.

Programmatic assessment was done using Pandas functions on the three tables/dataframes:

- head()
- info()
- describe()
- value_counts()
- sample()

Multiple data quality and data tidiness issues were identified and resolved. Some issues identified have been marked as scope for further work.

The specifications of the project to include only original ratings, exclude tweets after August 2017 and include only tweets with images were followed to identify issues and clean the data.

Cleaning Data

The tidiness issues were dealt with first by making structural / organizational changes to the data. The data quality issues were then handled on a properly structured dataset using Pandas functions.

The issues identified and how they were handled is outlined in the following table:

Data Tidiness Issues

Issue	Cleaning Process
The dog stages (doggo, pupper, floofer and puppo) in the twitter_archive table are supposed to be a single variable with possible values	▪ Extracted the dog stage from the tweet text using Regular expressions ▪ Checked that it matches with cases where the stage is already specified ▪ Assigned NaN for cases where the stage could not be extracted from tweet text ▪ Removed the four separate columns
The additional data gathered from the API and image predictions can be merged with the twitter_archive table as they contain information about the same tweets.	▪ Used Pandas merge function to join the tables on tweet_id ▪ Result is a single dataframe with information from all three tables

Data Quality Issues

Issue	Cleaning Process
Erroneous datatypes are present in the twitter archive table: tweet_id, in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id as they appear like numerical columns	Converted the datatype to string using astype()
Fix datatype for tweet_id in image_pred table and data_from_api table	Since the datasets were merged, tweet_id was handled in the previous issue itself

Datatype for the timestamp in twitter_archive is erroneous. Once the datetime datatype is fixed it needs to be checked that there are no tweets after August 2017	Converted the datatype to datetime() using Pandas to_datetime() and checked that there were no timestamps later than Aug 2017
Nulls in the dog stages column need to be handled as they are represented as None	This was dealt with while addressing tidiness issues relating to the dog_stage variable
Datatype of p1, p2, p3 is better represented as category than object in image predictions	Converted the datatype to 'category' using astype()
We want only original ratings. Retweet records are present in the dataset.	Dropped the rows where the status_id corresponding to retweets was not null
Replies also do not qualify as original tweets.	Dropped the rows where the status_id corresponding to replies was not null
Unwanted columns relating to retweets and replies are present	Dropped columns that had variables denoting retweet or reply details
Some records from the twitter_archive table do not have image_urls and hence no predictions. We want only tweets with images	Dropped the rows from the dataset where jpg url was null, which meant there were no images for those tweets
Validity of jpg_urls	A crude check was carried out by checking if the url contained http. This is not foolproof and can be checked more intensively

Issues identified for future work:

- The 'name' column has a lot of erroneous entries like a, an, the , etc. and a lot of 'None' values. The extraction can be done more accurately from the tweet text
- The rating_numerator and rating_denominator have values other that are unlikely in some cases and can be extracted from the tweet text
- The data on breed prediction can be organized better to reduce redundant variables
- The expanded_urls column has some missing data

Results

The twitter_clean dataframe was saved to a master csv file to be used for analysis and visualization.