Data Wrangling Report

Objectives

The objectives of this case study are:

- **Data wrangling**: In this objective data was gathered from three different sources using manual and programmatic methods, which were then assessed (for quality and tidiness) and cleaned for storing in preparation for exploratory data analysis.
- **Storing**: After data-wrangling, storing the gathered, assessed, and cleaned data into a `'twitter_archive_master.csv'` file, made sure our data was safe and can always be referenced from the desired checkpoint.

Gathering data

In the phase, three datasets were gathered from different sources and stored in a pandas data frame for assessment.

- The WeRateDogs Twitter archive dataset was downloaded manually from a resource page in udacity as 'twitter-archived-enhanced.csv'.
- The tweet image prediction dataset was downloaded programmatically as 'image-predictions.tsv' using the Requests library and the URL hosted on udacity server.
- Additional data from Twitter API were downloaded from querying the Twitter API for each tweet's JSON data using Python's Tweepy library and then each tweet's entire set of JSON data was stored in a file called 'tweet_json.txt' file. Each tweet's JSON data was written to its own line. Then read the .txt file line by line into a pandas DataFrame with tweet ID, retweet count, and favorite count.

Assessing and Cleaning Data

In these phases, data was assessed visually and programmatically, a process of defining the issue to be assessed, coding the issue, and then testing the code to verify it is clean. In the table below are the observation being made and the actions taken to solve them.

Quality issues

Dataset	Observation	Action
twitter_archived	1. Column named name: inaccurate names (a, an, the,...) were seen in column 2. Column named source: html anchor tag should not be included in the observations 3. There are 181 rows non-null values in retweeted_status_id, retweeted_status_user_id and retweeted_status_timestamp. Therefore there are 181 rows of retweets in the dataset which would need to be dropped 4. Missing records: in_reply_to_status_id (78 instead of 2356), in_reply_to_user_id (78 instead of 2356), retweeted_status_id (181 instead of 2356), retweeted_status_user_id (181 instead of 2356), retweeted_status_timestamp (181 instead of 2356), expanded_urls (2297 instead of 2356) 5. Erroneous datatypes: tweet_id, timestamp	1. Due to the fact that the name of the dogs wasn't going to be used in the analysis, no value was changed and therefore it was not cleaned. 2. Pandas splitting function was used to separate the anchor tags from the links. 3. All retweets were removed by filtering out all 181 non-null values from the table 4. Due to the fact these columns weren't going to be used in the analysis, they were dropped from this analysis. . 5. tweet_id was changed from int to object data type, and timestamp was changed from object to DateTime data type
image_predictions	1. Erroneous datatype: tweet_id	1. tweet_id was changed from int to object data type 2. All names were capitalized

Dataset	Observation	Action
	2. p1, p2, and p3 have an inconsistent structure in the capitalization of names	
twitter_data	1. Erroneous datatype: tweet_id	1. tweet_id was changed from int to object data type

Tidiness issues

Dataset	Observation	Action
twitter_archived	1. doggo, floofer, pupper, and puppo mean the same thing which is 'stage' so they should be in rows instead of columns	1. After cleaning each column for inaccurate data, each of the stage column values were concatenated to form one, so that any tweet with more than one stage will be a list
image_predictions	1. All columns were observational units of twitter_archived table	1. Pandas merging was used to image_predictions dataframe with twitter_archived dataframe
twitter_data	1. Columns retweet_count and favorite_count: This observational units should be in the twitter_archived table	1. Pandas merging was used to twitter_data dataframe with twitter_archived dataframe

Results

Merged table of twitter_archived, image_predictions and twitter_data dataframes

```
1]: twitter_archived_clean.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 2174 entries, 0 to 2173
Data columns (total 21 columns):
tweet_id              2174 non-null object
timestamp             2174 non-null datetime64[ns]
source                2174 non-null object
text                  2174 non-null object
rating_numerator      2174 non-null int64
rating_denominator    2174 non-null int64
name                  2174 non-null object
stage                 2174 non-null object
jpg_url               1993 non-null object
img_num               2174 non-null int64
p1                    1993 non-null object
p1_conf               1993 non-null float64
p1_dog                1993 non-null object
p2                    1993 non-null object
p2_conf               1993 non-null float64
p2_dog                1993 non-null object
p3                    1993 non-null object
p3_conf               1993 non-null float64
p3_dog                1993 non-null object
retweet_count         2174 non-null int64
favorite_count        2174 non-null int64
dtypes: datetime64[ns](1), float64(3), int64(5), object(12)
memory usage: 373.7+ KB
```

Storing

Finally, after the process of gathering, assessing, and cleaning the dirty data, the merged table of twitter_archived, image_predictions and twitter_data dataframes was stored in a '`twitter_archive_master.csv`' file, to make sure our data was safe and can always be referenced from the desired checkpoint.

```
twitter_archived_clean.to_csv('twitter_archive_master.csv',index=False)
```